Intel® Celeron™ Processor-Based Transaction Terminal Sample Design

Application Note

September, 1999

Order Number: 273281-001

Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The Intel® Celeron™ processor and Intel 440BX AGPset may contain design defects or errors known as errata which may cause the products to deviate from published specifications. Current characterized errata are available on request.

MPEG is an international standard for video compression/decompression promoted by ISO. Implementations of MPEG CODECs, or MPEG enabled platforms may require licenses from various entities, including Intel Corporation.

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

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature may be obtained by calling 1-800-548-4725 or by visiting Intel's website at http://www.intel.com.

Copyright © Intel Corporation, 1999

*Third-party brands and names are the property of their respective owners.

Application Note

Contents

Figures

Tables

1.0 Introduction

Transaction terminals include point-of-sale (POS), ATM, kiosk, and service terminals. This application note provides a sample transaction terminal design using the Intel Celeron™ processor and provides recommendations to enable shorter design cycles. This application note also describes various peripherals commonly used in transaction terminals (TTs) and how they interface to a Celeron processor platform.

Caution: This sample design has not been implemented in hardware. This document is for reference only. Customers are responsible for validating designs created using the information in this document.

1.1 Key Terms and Abbreviations

AGP	Accelerated Graphics Port
ATM	Automatic teller machine
PIIX4E	The Intel 82371EB PCI ISA IDE Xcelerator, which is one of two chips in the Intel 440BX AGPset used in this sample design. Also called the Southbridge.
POS Terminal	Point-of-sale terminal
TT	Transaction terminal

1.2 Revision History

Revision Number	Date	Description
001	09/20/99	Initial Version.

2.0 Embedded Applications Overview

2.1 Influence of PC Technology

Embedded systems come in a variety of forms, including transaction terminals, industrial equipment, and telecommunication equipment.

Increasingly, high-performance embedded processors are being used in embedded applications, and PC technology is quickly filling the performance needs. Using desktop processors in the embedded applications with standard operating systems allows embedded designers to get their products to market quickly.

Transaction terminals represent a broad category of computing devices including electronic cash registers, PC-based point-of-sale (POS) terminals, handheld computers, back-office PCs, servers, automatic teller machines, and point-of-interaction kiosks and service terminals. These terminals are either stand-alone or connected to the corporate enterprise network. With increasing competition in the retail and financial industries, the ability to network these devices is an extremely high priority. Networking these applications provides the infrastructure for end customers, such as retailers, to achieve real-time collection and processing of sales data. Retailers can then use this information to better manage their business and optimize their operating efficiency. In many cases, connectivity expands beyond the store fronts and back-room inventory to the production assembly, financial institutions, credit card processing centers, warehouses, online E-commerce services, and alliance stores.

POS terminals, widely used in supermarkets, department stores, restaurants, convenience stores, and banks, are PC-like platforms with peripherals added to address various needs. The main function of a POS terminal is to process transactions such as receipt generation, scanning, weighing, and inventory management. A POS can be a stand-alone system or can be used in conjunction with back-office systems for inventory management, training, and advertising. POS terminal users may need to look up products, re-index merchandise by product code, description or stock number, or track layaway and backorders. When discounts are offered, the POS terminal can support multiple price levels.

Transaction terminals require high performance to support demanding front-end applications, such as “commercial-ready” multimedia, extensive peripheral connectivity, wireless and mobile devices. They also require the ability to access the enterprise network through an Intranet or the Internet using Windows NT-based servers. Performance upgrades are being driven by the growing requirements of distributed databases, loaded on the front end, and rich data type content such as video and high resolution graphics.

2.2 Celeron™ Processors in Embedded Applications

Intel Pentium®, Pentium II-Low Power and Celeron processors are now being used in embedded systems. The sample design discussed in this document uses the Celeron processor. The Intel family of processors for embedded applications now includes the 300A, 366 and 433 MHz Celeron processors, which include MMX™ technology. With support for embedded product life cycles, the Celeron processor—and supporting Intel chipsets—enable developers to meet the high performance requirements of today's value-driven embedded applications with the same technology used in today’s value desktop PCs.

Using the Celeron processor in embedded designs provides scalability and offers a wide performance range and an easily alterable platform.

3.0 Overview of Transaction Terminals

A typical transaction terminal designed for a point-of-sale (POS) application is shown in Figure 1.

Figure 1. Stand-alone POS Terminal

A6263-01

3.1 Transaction Terminal Implementation

Transaction terminals are PC-like platforms adapted for retail and service environments. They may include liquid-crystal display (LCD) touch screens, which are especially important in many retail and service environments where the operator must select numerous items quickly.

Unlike a desktop system, a transaction terminal may have an expanded number of serial and Universal Serial Bus (USB) ports. These additional serial ports are usually needed for peripherals such as bar code scanners, credit card readers, magnetic stripe readers, and pin pads.

USB keyboards, mice, and receipt printers can be implemented as hot plug-and-play peripherals. Some of the other peripherals common in transaction terminals include cash drawers, digital scales, and pole displays.

In most cases, transaction terminals must be available at all times. For this reason, hardware monitoring and the capability for fault recovery are crucial. Since transaction terminals are typically kept on at all times, the processor's power management features should be used to place the terminals in a low-power mode when not used.

This sample design tested peripherals and Windows CE drivers to provide support for:

- Infrared (IrDA)
- Flash memory
- PCMCIA cards
- AC '97-compliant audio codec
- Modem
- LCD touch screen
- Network interface cards.

This sample design discusses the components used for a PCMCIA card, touch screen controller, LCD/touch screen, additional serial ports, infrared, and the AC '97 audio codec.

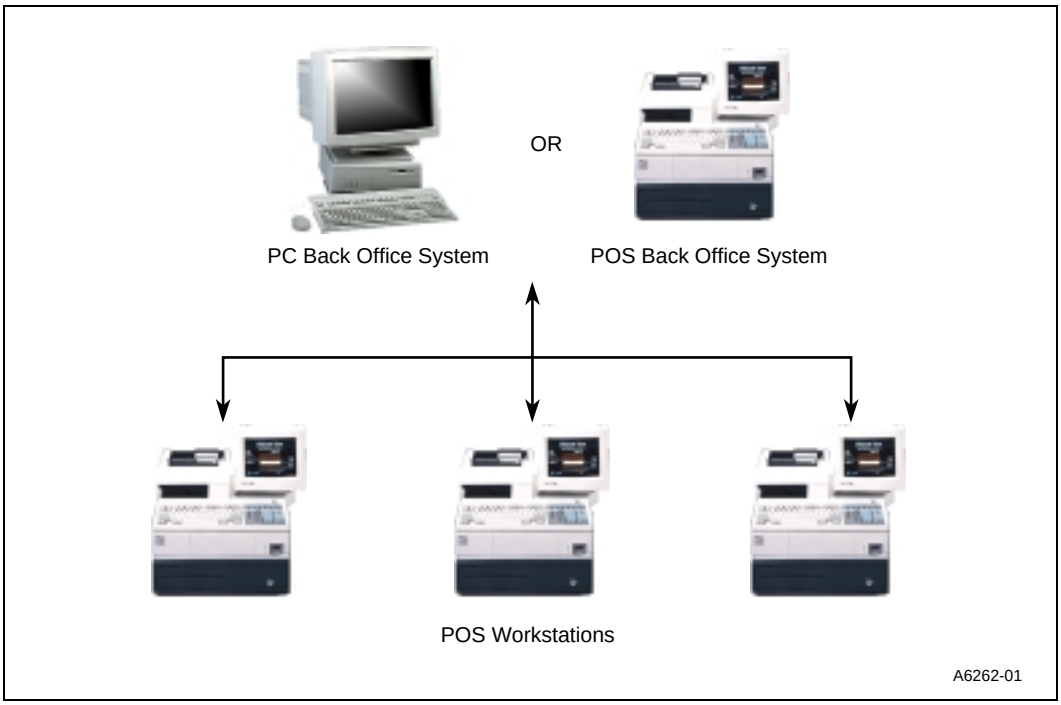
3.2 Transaction Terminal Operating Systems

In transaction terminal applications using the embedded Celeron processor, frequently used operating systems include DOS*, QNX*, Linux*, BeOS*, VxWorks*, pSOS*, and Windows* 3.1. In addition, Windows NT, Windows 95, and Windows 98 are now being used to meet the increasing performance needs of many embedded applications. These Windows operating systems have the advantage of having a large base of readily available applications, and features such as Desktop Management Interface (DMI), and support for interactivity. Another operating system solution that is gaining momentum in transaction terminal applications is Windows CE, which can be used in smaller applications that require a ROM-able operating system.

A typical transaction terminal implementation can include a high performance back-end system and a front-end system with less functionality. The hardware core of a transaction terminal is similar for ATMs, Kiosks, and POS terminals. In a traditional POS implementation, a company may have to develop and train people on two or more systems. At the machine level, or front-end, where functionality is limited, applications may be developed based on proprietary operating systems. At the back-end or supervisory level, applications are developed on open systems based on Windows 95 and Windows NT operating systems. Because these levels are often built on different platforms, the systems may not talk to each other, look the same, or share databases, graphics, and other application files.

An alternative to the configuration described above is to design a transaction terminal that uses an embedded Celeron processor with MMX technology running Windows 95, Windows 98, Windows NT, or Windows CE, and a based back-end system that uses a Pentium II or Pentium III processor with Windows NT. This configuration enables transaction terminal system designers to add functions such as inventory management and disaster recovery. In addition, the front-end transaction terminal can communicate with the back-end or supervisory level systems. This system allows sharing databases, graphics and applications. By using an off-the-shelf operating system, the added benefits are shorter design cycles and quicker time-to-market. Figure 2 illustrates this configuration.

Figure 2. Back-end and Front-end Transaction Terminal



3.2.1 Windows* CE

All software for Windows CE is based on the Win32 application programming interface (API). Developers can write programs for Windows CE using familiar tools such as Visual C/C++*, or Visual Basic*. Windows CE helps in moving some embedded applications away from proprietary systems so that products have a shorter development cycle, fewer costs and less user training. Since Windows CE is ROM-able, a hard disk drive may not be necessary.

Because Windows CE, Windows 95, and Windows NT are based on the same Microsoft technologies, many of the same development tools and utilities can be used on both systems, allowing faster development and a similar look and feel.

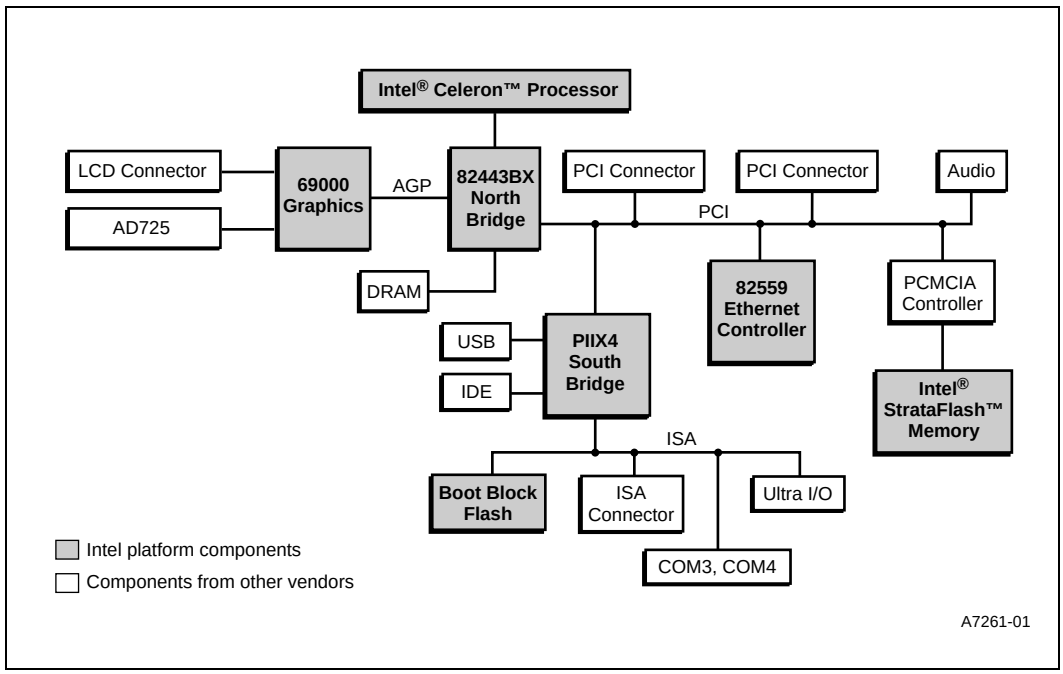
For an application with reduced functionality, Windows CE can be used as the core OS because it requires a small footprint, unlike Windows NT. A Windows CE-based device with the kernel, the file system, and the registry can fit into 512 Kbytes of ROM.

Windows CE must be configured to meet the specific needs of the application. Functionality must be added to it in the form of installable device drivers. Windows CE has numerous components allowing for different configurations. Windows CE also uses Unicode characters to provide multilingual support.

4.0 Transaction Terminal Sample Design Overview

Figure 3 is the block diagram for the transaction terminal sample design. Intel components are shown in gray shading. The design also includes several components from other vendors.

Figure 3. Block Diagram of Transaction Terminal Platform Sample Design



4.1 Core Components

As shown in Figure 3, the core components of this reference design are:

- Intel Celeron processor
- 82443BX “northbridge”
  - — DRAM interface
  - — PCI bus and connectors
  - — AGP port
- 82371EB PCI-to-ISA/IDE Xcelerator (PIIX4E) “southbridge”
  - — IDE connector
  - — ISA bus and connector
  - — Universal Serial Bus host controller
- 82559 Ethernet Controller
- 69000 HiQVideo™ Accelerator with Integrated Memory

- AC ’97 Audio Solution
  - — Ensoniq AudioPCI* ES1371
  - — AD1819 SoundPort* Codec
- SMSC FDC37B78x Ultra I/O*
- SMSC FDC37C669 Super I/O* for additional serial ports
- TriTech Microelectronics TR88L803 Touch Screen Controller
- Texas Instruments PCI1225 PC Card (PCMCIA) Controller
- Intel StrataFlash™ Memory

5.0 The Celeron™ Processor

This reference design supports the Celeron™ processor with MMX™ technology and an integrated 128-Kbyte L2 cache at a core clock rate of 300, 366, or 433 MHz. The Celeron processor has a system bus that runs at 66 MHz. For embedded applications, this processor is available in a 370-pin PPGA package.

The Intel Celeron processor is designed to provide high performance without sacrificing value, and is binary compatible with previous generations of Intel architecture processors. The Celeron processor provides enhanced performance for applications running on advanced operating systems such as Windows* NT, Windows CE, and UNIX*. The Celeron processor brings a balanced level of performance to the embedded market segment, integrating the best attributes of Intel processors—the dynamic execution performance of the P6 microarchitecture and the capabilities of MMX™ technology. Celeron processors also include the latest features to simplify system management and lower the total cost of ownership for small business environments.

The 300A, 366 and 433 MHz chips feature Intel's dynamic execution P6 microarchitecture core with a 66 MHz multi-transaction system bus. The 128-Kbyte on-chip L2 cache operates at the same speed as the processor. MMX technology enhances the performance of embedded products that run software applications for transaction terminals, imaging, and video processing. In addition, applying MMX technology to networking software stacks can help dramatically improve the performance of communications applications. The imaging features, accelerated by MMX technology, enable exciting new capabilities for transaction terminals, such as incorporating JPEG images and video processing.

The Celeron processor is available in a scalable 370-pin plastic pin grid array (PPGA) package (49x9mm) for embedded applications.

While many embedded applications require ever-higher performance, they are also driven by value considerations. The Celeron processor meets emerging product requirements in these important application segments:

- In retail and financial transaction terminals, increased performance is required to support Internet technologies such as Java*, JPEG imaging, video, and audio.
- The Celeron processor and MMX technology single instruction multiple data (SIMD) instructions support the increased processing demands of graphics, imaging, and video.
- The Celeron processor in Socket 370 packaging provides scalability and flexibility to meet a variety of price/performance targets.

For additional information on the Celeron processor, refer to the *Intel® Celeron™ Processor* datasheet. Information on PGA370 can be found in *370-Pin Socket (PGA370) Design Guidelines*. Note that external termination, power distribution, power decoupling, and adherence to PC layout rules is required as provided in the documents listed in Appendix A, “Related Resources”, in Table 1, “General Purpose Input and Output Pins” on page 24, and in the schematics in Appendix B.

5.1 Using the Celeron™ Processor with the Intel® 440BX AGPset

The Celeron processor and the 440BX AGPset provide a performance-optimized solution for embedded product life cycles. The 440BX AGPset consists of two chips, the 82443BX and the 82371EB:

- The 82443BX, also called the northbridge, forms a bridge between the processor, memory, Accelerated Graphics Port (AGP), and PCI bus.
- The 82371EB is also called the PIIX4E (PCI ISA IDE Xcelerator) or southbridge.

The 440BX improves the performance of the Celeron processor with Quad Port Acceleration (QPA). By using four ports, QPA improves the bandwidth between the processor, SDRAM, PCI bus, and Accelerated Graphics Port (AGP).

The 440BX includes a number of other performance-enhancing features, including improved bus arbitration, deeper buffers, and an open-page memory architecture. Designing with the 440BX AGPset provides a cost-effective way to ensure that your embedded designs will be ready for future 100 MHz bus implementations.

5.2 Data Integrity Features

Data integrity is critical in many embedded applications. The Celeron processor supports Built-in Self Test (BIST) to provide single stuck-at fault coverage of the microcode and large logic arrays. In addition, BIST tests the instruction and data caches, Translation Lookaside Buffers, and ROMs. The processor also features an IEEE 1149.1 self-test port. For additional data integrity, the 440BX AGPset includes Error Correction Code (ECC) memory control.

5.3 The Celeron™ Processor Execution Architecture

The P6 family of processors succeeds the Pentium processor line of Intel processors. The P6 processors implement Intel's dynamic execution microarchitecture, which incorporates a unique combination of multiple branch prediction, data flow analysis, and speculative execution. These features enable P6 family processors to deliver higher performance than the Pentium family of processors, while maintaining binary compatibility with all previous Intel architecture (IA) processors.

The Celeron, Pentium II, and Pentium III processors are aggressive P6 microarchitectural implementations of the 32-bit Intel architecture. They are designed with a dynamic execution architecture that provides the following features:

- out-of-order speculative execution to expose parallelism
- superscalar issue to exploit parallelism
- hardware register renaming to avoid register name space limitations
- pipelined execution to enable high clock speeds
- branch prediction to avoid pipeline delays

The microarchitecture is designed to execute legacy 32-bit Intel architecture code quickly, without additional effort from the programmer. The new Celeron processors also contain an integrated 128-Kbyte L2 cache and an Advanced Programmable Interrupt Controller (APIC). The APIC supports I/O and is 8259A compatible.

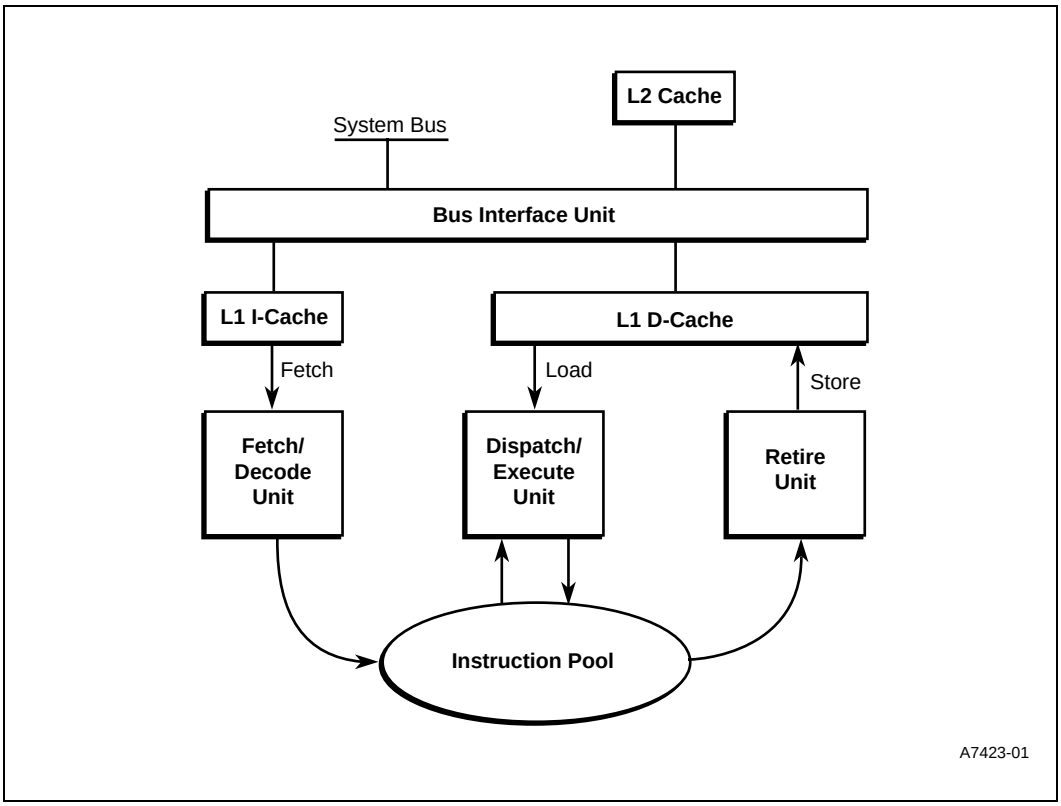
5.3.1 The Celeron™ Processor Pipeline

The Celeron processor pipeline contains three independent engines coupled with an instruction pool. The three engines are:

- Fetch/Decode - The in-order issue front end
- Dispatch/ Execute - The out-of-order core
- Retire - The in-order retirement unit

Figure 4 shows an overview of the Celeron processor's architecture.

Figure 4. Block Diagram of the Celeron™ Processor's P6 Core



5.3.2 The Bus Interface Unit

The bus interface unit is responsible for connecting the three internal units to external hardware. The bus interface unit communicates directly with the L2 (second level) cache supporting up to four concurrent cache accesses. The bus interface unit also controls a transaction bus to system memory, with a MESI snooping protocol for cache coherence.

5.3.3 Fetch/Decode – The In-order Issue Front End Unit

The front end supplies instructions in program order to the out-of-order core. It fetches and decodes Intel architecture-based processor macroinstructions, and breaks them down into simple operations called micro-ops (µops). It can issue multiple µops per cycle, in original program order, to the out-of-order core.

Because the core aggressively reorders and executes instructions out of program order, the most important consideration in performance tuning is to ensure that enough µops are ready for execution. Accurate branch prediction, instruction prefetch, and fast decoding are essential to getting the most performance out of the in-order front end.

5.3.4 Dispatch/Execute – The Out-of-Order Core Unit

The core's ability to execute instructions out of order is a key factor in exploiting parallelism. This feature enables the processor to reorder instructions so that if one µop is delayed while waiting for data or a contended resource, other µops that are later in program order may proceed around it. The processor employs several buffers to smooth the flow of µops. This implies that when one portion of the pipeline experiences a delay, that delay may be covered by other operations executed in parallel or by executing µops which were previously queued up in a buffer.

The out-of-order core buffers µops in a Reservation Station (RS) until their operands are ready and resources are available. The core may dispatch up to five µops each cycle. The core is designed to facilitate parallel execution. Load and store instructions may be issued simultaneously. Most simple operations, such as integer operations, floating-point add, and floating-point multiply, can be pipelined with a throughput of one or two operations per clock cycle. Long latency operations can proceed in parallel with short latency operations.

5.3.5 Retire – The In-Order Retirement Unit

For semantically-correct execution, the results of instructions must be processed in original program order. Likewise, any exceptions that occur must be processed in program order. When a µop completes and writes its result, it is retired. Up to three µops may be retired per cycle. The unit in the processor that buffers completed µops is the reorder buffer (ROB). The ROB updates the architectural state in order; that is, updates the state of instructions and registers in the program semantics order. The ROB also manages the ordering of exceptions.

6.0 Intel® 440BX AGPset

The Intel 440BX AGPset (440BX) optimizes Celeron processor performance for 3-D and video applications. As Intel's second-generation AGPset with Intel Quad Port Acceleration (QPA), the Intel 440BX AGPset improves the performance of the system bus by increasing the width and depth of buffers to the system bus, Accelerated Graphics Port, SDRAM, and PCI bus. The 440BX AGPset is compatible with ATA/66 HDD. This makes the 440BX AGPset an excellent design solution for next-generation performance embedded systems. The new 440BX AGPset supports the emerging set of visual computing applications, including 3-D and video applications for design, data visualization, multimedia, and education. The 440BX integrates many power management features that enable power savings when the system resources are idle.

The 440BX consists of two chips: the 82443BX System Controller and the 82371EB PCI ISA IDE Xcelerator (PIIX4E). These components are described in the following sections.

6.1 82443BX System Controller

The 82443BX is the Host-to-PCI northbridge. It connects the Celeron processor to memory, the Accelerated Graphics Port (AGP), and the PCI bus. It also contains an optimized DRAM controller and data path and power management functions.

The 82443BX functions and capabilities include:

- 64-bit GTL+ based system data bus interface
- 32-bit system address bus support
- 64/72-bit main memory interface with optimized support for SDRAM
- 32-bit PCI bus interface with integrated PCI arbiter
- AGP interface with up to 133 MHz data transfer capability
- Extensive data buffering between all interfaces for high throughput and concurrent operations

The physical interface design is based on the Gunning Transceiver Logic (GTL+) specification and is compatible with the Intel 440BX AGPset solution. The 82443BX provides an optimized 72-bit DRAM interface (64-bit data plus ECC). This interface supports 3.3 V DRAM technologies.

Special termination and careful routing is required on the high-speed GTL+ technology data bus signal lines. GTL+ is a low output swing, incident wave switching, open-drain bus with external pull-up resistors that provide both the high logic level and termination at the end of the bus. This termination is included in this design. Additional details can be found in the *Pentium® II Processor AGTL+ Guidelines* (see Appendix A for document locations). Analog signal simulation of the Intel Celeron processor system bus, including trace lengths, is highly recommended when designing a system. Information on thermal and EMI requirements can be found in *Pentium® II Processor Thermal Design Guidelines* and *Pentium® II Processor Electro-Magnetic Interference Guidelines*. Power requirements are described in the *Pentium® II Processor Power Distribution Guidelines*. Detailed layout rules can also be found in the design schematics in Appendix B.

6.2 Intel® 82371EB PCI ISA IDE Xcelerator (PIIX4E)

The 82371EB PCI ISA IDE Xcelerator (PIIX4E), or southbridge, is a multi-function PCI device implementing a PCI-to-ISA bridge function, a PCI IDE function, a Universal Serial Bus host/hub function, and an enhanced power management function.

This device is compatible with the Intel 82443BX. The PIIX4E also allows complete plug-and-play compatibility. It supports two IDE connectors (Ultra DMA/33) to be used with up to four IDE devices in bus master mode.

As a PCI-to-ISA bridge, the PIIX4E integrates many common I/O functions found in ISA-based PC systems: two 82C37 DMA Controllers, two 82C59 Interrupt Controllers, an 82C54 Timer/ Counter, and a Real Time Clock. In addition to compatible transfers, each DMA channel supports Type F transfers. The PIIX4E also contains full support for both PC/PCI and Distributed DMA protocols implementing PCI-based DMA. The Interrupt Controller has edge- or level-sensitive programmable inputs and fully supports the use of an external I/O Advanced Programmable Interrupt Controller (APIC) and serial interrupts. Chip select decoding is provided for the BIOS, a real time clock, a keyboard controller, a second external microcontroller, and two programmable chip selects. The PIIX4E provides full plug-and-play compatibility.

The PIIX4E supports Enhanced Power Management, including full Clock Control, Device Management for up to 14 devices, and Suspend and Resume logic with Power On Suspend, Suspend to RAM, or Suspend to Disk. It fully supports operating system directed power management via the Advanced Configuration and Power Interface (ACPI) specification. PIIX4E integrates both a System Management Bus (SMBus) host and slave interface for serial communication with other devices.

For PCI devices, the PIIX4E is PCI device #7 (IDSEL connected to pin AD18 through a 110 Ω resistor per *PCI Local Bus Specification*, Revision 2.1).

The PIIX4E's interrupt control unit provides interrupt handling to all ISA devices on the board including the SMSC FDC37B78x Ultra I/O* (floppy disk, serial ports, parallel port control), and the SMSC 37C669 Super I/O* (2 serial ports).

The following sections describe these features in greater detail. For the complete specification for this product, refer to the *82371EB PCI-TO-ISA/IDE Xcelerator (PIIX4E)* datasheet

6.2.1 IDE / Floppy

The PIIX4E supports two IDE connectors for up to four IDE devices, providing an interface for IDE hard disks and CDROMs. Up to four IDE devices can be supported in bus master mode. The PIIX4E contains support for "Ultra DMA/33" synchronous DMA compatible devices. This design uses one IDE connector with support for up to two devices.

6.2.2 XBUS, Power Management, and USB Host Controller

The PIIX4E contains a Universal Serial Bus (USB) Host Controller that is compatible with the Universal Host Controller Interface (UHCI). The Host Controller's root hub has two programmable USB ports. Follow these guidelines when implementing these functions.

- X-Bus Signals

  XOE# and XDIR# are connected as control signals to the X-Bus transceiver, with XOE# connected to G# and XDIR# connected to DIR of the transceiver. The internal RTC of the PIIX4E is used; therefore RTCALE and RTCCS# become general purpose outputs (GPO25 and GPO24, respectively) by programming the General Configuration Register (GENCFG) in Function 0, Offset B0h–B3h.

- Power Management Signals

  SUSA# of the PIIX4E is connected to the PWR_DWN# signal of the clock generator. SUSA# is asserted during power management suspend states, POS, STR, and STD suspend states. PWR_DWN# is an active low control input to the clock generator to power down the device.

  CPU_STP# and PCI_STP# of the PIIX4E are asserted low to disable the processor and PCI clock outputs respectively. They are connected to CPU_STOP# and PCI_STOP# of the clock generator.

  SUSC# of the PIIX4E is first inverted and then connected directly to PS_ON# of the ATX power supply. Therefore when SUSC# is asserted, during STD suspend state, all power rails including 3.3 V, 5 V, -5 V, 12 V, and -12 V power rails, are turned off. This is used for the remote-off function. The inverter described above *must* be connected to a Standby Power rail.

  RSMRST# connection requires a minimum time delay of one millisecond from the rising edge of the standby power supply voltage. A simple RC circuit is used to provide this time delay (t = RC, in the design this is ~2.7 KΩ * 10 µF ~ 2.7 ms) and a Schmitt trigger circuit is used to drive the signal on the board.

- USB Interface

  The following are general layout guidelines for the USB interface:

  — Any unused USB ports should be terminated with 15 Kbyte pull-down resistors on both P+/P- data lines.

  — 27 Ω series resistors should be placed as close as possible to the PIIX4E (<1 inch). These series resistors are for source termination of the reflected signal.

  — 47 pF caps must be placed as close to the PIIX4E as possible and on the PIIX4E side of the series resistors on the USB data lines (P0+, P1+). These caps aid in signal quality (rise/fall time) and help to minimize electromagnetic interference (EMI).

  — 15 KΩ ±5% pull-down resistors should be placed on the USB side of the series resistors on the USB data lines (P0+ and P1+), and are REQUIRED for signal termination by USB specification. The stub length should be as short as possible.

  — The trace impedance for the P0+, P1+ signals should be 45 Ω (to referenced ground) for each USB signal P+ or P-. The impedance is 90 Ω between the differential signal pairs P+ and P- to match the 90 Ω USB twisted pair cable impedance. Note that the twisted pair characteristic impedance of 90 Ω is the series impedance of both wires, resulting in an individual wire presenting a 45 Ω impedance. The trace impedance can be controlled by carefully selecting the trace width, trace distance from power or ground planes, and physical proximity of nearby traces.

— USB data lines must be routed as “critical signals” (i.e., hand routing is preferred). The P+/P- signal pair must be routed together and not parallel with other signal traces to minimize crosstalk. Doubling the space from the P+/P- signal pair to adjacent signal traces helps prevent crosstalk. Do not worry about crosstalk between the two P+/P- signal traces. The P+/P- signal traces must also be the same length. This will minimize the effect of common mode current on EMI. (Common mode current is caused by differential signals whose currents are not perfectly matched.)

The following are general layout guidelines for the USB power and ground lines:

— Ferrite beads are placed on each power and ground line to minimize EMI. They should be placed as close as possible to the USB connector.

— Voltage divider circuits are used to drive the status of the SUB power line to the OC[1:0]# inputs. OC[1:0]# signals are 3.3 V inputs and have a leakage current of maximum + 1 µA.

— Fuses are used on each power line for overcurrent protection.

Refer to the *Intel® 440BX PCIset Design Guide* for sample layout topologies.

- IDE Interface

  The PIIX4E provides two standard IDE interfaces. One (the primary IDE interface) is included in this design. The primary IDE controller is connected as described in the *Intel® 440BX PCIset Design Guide*. The secondary IDE controller is not used on the board. Hence, all inputs to the secondary IDE controller are pulled to the inactive state with a 10 KΩ resistor. All outputs and bidirectional pins are left floating.

  A hard drive active LED circuit is connected to the HD_ACT# signal of the primary IDE connector. If the secondary IDE controller is used, an OR-gate between HD_ACT1# and the HD_ACT2# should be used to drive the LED circuit.

  Proper operation of the IDE circuit depends on the total length of the IDE bus. The total signal length of the IDE drivers to the end of the IDE cables should not exceed 18". Therefore, the PIIX4E should be located as close as possible to the IDE connectors to allow the IDE cable to be as long as possible. When the distance between the PIIX4E and the ATA connectors exceeds four inches, the series termination resistors should be placed within one inch of the PIIX4E. Designs that place the PIIX4E within four inches of the ATA connectors can place the series resistors anywhere along the trace. The capacitance of each pin of the IDE connector on the host should be below 25 pF when the cables are disconnected from the host.

- Clocks

  Refer to the *Intel® 440BX PCIset Design Guide* for layout topologies.

- General purpose input and output pins

  Fifteen general purpose inputs and fifteen general purpose outputs are provided on the board. Table 1 lists the pins and provides sharing information:

Table 1. General Purpose Input and Output Pins

GPO	Pin	Muxed With	GPI	Pin	Muxed With
GPO0	1	None	GPI1	2	None
GPO8	3	None	GPI2	4	REQA#
GPO9	5	GNTA#	GPI3	6	REQB#
GPO10	7	GNTB#	GPI4	8	REQC#
GPO11	9	GNTC#	GPI5	10	APICREQ#
GPO12	11	APICACK#	GPI7	12	None
GPO15	13	APICCS#	GPI13	14	None
GPO21	15	SUS_STAT2#	GPI14	16	None
GPO24	17	RTCCS#	GPI15	18	None
GPO25	19	RTCALE	GPI16	20	None
GPO26	21	KBCCS#	GPI17	22	None
GPO27	23	None	GPI18	24	None
GPO28	25	None	GPI19	26	None
GPO29	27	IRQ9OUT#	GPI20	28	None
GPO30	29	None	GPI21	30	None

NOTE: All GPIO pins in the sample design are always available. Pins that are multiplexed with other signals are not used for their alternate function and are therefore always available for GPIO.

7.0 Functional Description of Sample Design Hardware

This section describes additional hardware components used in the sample design (the Celeron processor is described in Section 5.0 and the Intel 440BX AGPset is described in Section 6.0). For more detailed information about the transaction terminal design, refer to the schematics located in Appendix B

7.1 In-Target Probe (ITP)

The design includes an in-target probe (ITP), also called a Pentium II processor debug port. This is a 30-pin interface that communicates directly to the processor, allowing access to the processor's registers and signals. Using this debug utility allows easier debugging of low-level code such as BIOS and drivers.

The connectors for the ITP are provided by AMP Incorporated. Debuggers can be purchased through a variety of vendors.

7.2 Power Management and Voltage Regulation

A Semtech SC1185 programable synchronous DC/DC converter, dual low dropout regulator controller is used to provide the 1.5 V, 2.5 V, and core supplies. The VIDx pins on the Celeron processor select the core voltage to be supplied by the SC1185 device. PCB Layout guidelines for the SC1185 are included in the schematics in Appendix B. Voltage and temperature are monitored with sensors and the system shuts down whenever critical problems are detected. Voltage outputs are monitored by the reset and power good circuitry with the Linear Technology LTC1326 triple supply monitor. Processor temperature is monitored with a Maxim MAX1617 device.

As a design alternative, a National Semiconductor LM79 hardware monitor chip can be used instead of the LTC1326 triple supply monitor. In addition to voltage outputs, the LM79 chip can monitor fan RPM and chassis intrusion.

7.3 Cache

The concept of level two (L2) cache developed from the need to have data and instructions available to the processor locally. With the high clock rates of newer generation processors, there is enough time to allow for a code fetch and data to arrive from system DRAM for the processor to use before a stall occurs. With faster processors, the need for L2 cache is even greater; the processor may be so fast that, without cache, the processor core stalls while waiting for code and data to arrive from the system DRAM. While the processor is stalled waiting for instructions, it cannot perform other tasks.

The Celeron processors offer dual 16-Kbyte, 4-way set associated level one (L1) caches, with the addition of an integrated 128-Kbyte L2 cache. This architecture greatly lessens bus utilization and increases the overall throughput of the system.

7.4 Networking

The 82559 Ethernet controller is used for the network interface in this design. The 82559 10/100 Mbps Fast Ethernet controller with an integrated 10/100 Mbps physical layer device is Intel's leading solution for PCI board LAN designs. It is designed for use in Network Interface Cards (NICs), PC LAN On Motherboard (LOM) designs, embedded systems, and networking system products. The 82559 combines a low power and small package design, which is ideal for power and space constrained environments.

The 82559 supports the following:

- Advanced Configuration and Power Interface (ACPI) 1.20 A-based power management
- wake on Magic Packet
- wake on interesting packet
- advanced System Management Bus (SMB)-based manageability
- Wired for Management (WfM) 2.0 compliance
- IP checksum assist
- PCI 2.2 compliance
- PC 98, PC 99, and Server 99 compliance

7.5 AGP Video

The 69000 HiQVideo™ Graphics Accelerator with integrated memory is used in this design.

The 69000 is a portable graphics accelerator that integrates high performance memory technology for the graphics frame buffer. Based on the HiQVideo graphics accelerator core, the 69000 combines flat panel controller capabilities with low-power, high performance integrated memory. The 69000 solution integrates 2 Mbyte SDRAM for use in frame buffering. This memory supports up to an 83 MHz clock, allowing a high memory bandwidth. The 69000 is connected to the AGP port in this design.

7.5.1 Features of the 6900 HiQVideo Graphics Accelerator

- High Performance Integrated Memory

  The 69000 incorporates 2 Mbyte of proprietary integrated SDRAM for the graphics/video frame buffer. The integrated SDRAM memory can support up to 83 MHz operation, increasing the available memory bandwidth for the graphics subsystem.

- HiQColor Technology

  The 69000 integrates HiQColor technology based on the proprietary Temporal Modulated Energy Distribution (TMED) algorithm, HiQColor technology is a process that allows the display of 16.7 million true colors of STN panels without using Frame Rate Control (FRC) or dithering. TMED also reduces the need for the panel tuning associated with current FRC-based algorithms. The TMED algorithm eliminates flaws normally associated with dithering and FRC such as shimmer, Mach banding, and other motion artifacts.

- Versatile Panel Support

  The HiQVideo family of graphics accelerators support a wide variety of monochrome and color single-panel, single-drive (SS) and dual-panel, dual drive (DD), standard and high-resolution, passive STN, and active matrix TFT/MIM LCD, and EL panels. With HiQColor technology, up to 256 gray scales are supported on passive STN LCDs. Up to 16.7 million different colors can be displayed on passive STN LCDs and on 24-bit active matrix LCDs.

  The 69000 offers a variety of programmable features to optimize display quality. Vertical centering and stretching are provided for handling modes with fewer than 480 lines on 480-line panels. Horizontal and vertical stretching capabilities are also available for both text and graphics modes for optimal display of VGA text and graphics modes on 800x600, 1024x768 and 1280x1024 panels.

- Television NTSC/PAL Flicker Free Output

  The 69000 uses a flicker reduction process, which makes text of all fonts and sizes readable by reducing the flicker and jumping lines on the display. The 69000 uses a line buffer and digital filters to average adjacent vertical lines for odd/even display. The chip also uses both horizontal and vertical interpolation to make both graphics and text appear smooth on a standard television. This process reduces the effect of flicker on NTSC displays.

7.5.2 Implementation of the 69000 HiQVideo Graphics Accelerator

The 69000 HiQVideo Accelerator provides VGA and LCD output on the board. Using the Analog Devices AD725 RGB to NTSC/Pal Encoder, the board also provides TV-out capability.

The 69000 is connected to the AGP port (IDSEL connected to AD16 through a 100 Ω resistor) and uses PCI interrupt INTA#.

An RGB monitor is always populated on the board. Therefore, when a TV encoder such as the AD725 is added, the RGB signals must be buffered. This is accomplished using an AD8073 video amplifier with high input impedances on each color signal. They are configured for a gain of two, which is normalized by the divide-by-two termination scheme used for the RGB monitor. However, since the RGB signals shift to ground during the horizontal sync period, the AD8073 requires a -5 V supply.)

The board provides an interface to LCD screens with up to 36-bit color. Panel Connector 1 supports 8-bit monochrome to 24-bit color LCD panels. The board also supports 36-bit LCD panels using both Panel Connector 1 and Panel Connector 2.

All panel interface signals from the graphics controller are 3.3 V. Level-shifting buffers should be added between the outputs from the graphics controller and the inputs of a 5.0 V flat panel if the 69000's V_{OL} and V_{OH} do not meet the requirements for the panel's V_{IL} and V_{IH}. The signals involved include pins P[35:0], SHFCLK, LP, FLM and M. The included schematics do NOT include these level shifters. The design assumes 3.3 V panels.

To ensure full compatibility with older CRT displays, 3.3 V to 5.0 V level-shifting buffers are added between the 69000's HSYNC and VSYNC outputs and the display's HSYNC and VSYNC inputs.

Power and ground signals to the 69000 are as specified in the *69000 HiQVideo Accelerator with Integrated Memory* datasheet (see Appendix A). Table 2 lists all power and ground requirements, and their implementation on the board.

Table 2. Power and Ground Requirements

Pin Name	Description	Requirement	Implementation
DACVCC	Analog power for the internal RAMDAC	DACVCC should be isolated from all other V_{CC} pins and should not be greater than CORVCC	Isolated from all other V_{CC} pins through ferrite beads with filtering caps
DACGND	Analog ground for the internal RAMDAC	DACGND should be common with digital ground but must be tightly coupled to DACVCC	Tied to analog GND
MCKVCC	Analog power for the internal memory clock synthesizer (MCLK)	MCKVCC must be at the same voltage level as CORVCC	Coupled to MCKGND through RC network
MCKGND	Analog ground for the internal memory clock synthesizer (MCLK)	MCKVCC/MCKGND pair must be INDIVIDUALLY decoupled	Tied to digital GND through ferrite bead
DCKVCC	Analog power pins for the internal dot clock synthesizer (DCLK)	DCKVCC must be at the same voltage level as CORVCC	Coupled to DCKGND through RC network
DCKGND	Analog ground pins for the internal dot clock synthesizer (DCLK)	DCKVCC/DCKGND pair must be INDIVIDUALLY decoupled	Tied to digital GND through ferrite bead
GND	Digital ground		Tied to digital GND
CORVCC	Digital power for the graphics controller internal logic (a.k.a. the “core” V_{CC})		
MEMGND	Embedded memory ground		Tied to 3.3 V plane with filtering caps
RGND	Internal reference GND	Should be tied to GND	Tied to digital GND
IOVCC	I/O power		TIed to 3.3 V plane with filtering caps
MEMVCC	Power for embedded memory		Tied to 3.3 V plane with filtering caps

Proper layout of the 69000 is important to getting optimal performance.

The power plane attached to the core power supply should be as wide as practical for high-current carrying capacity and low inductance.

Decoupling capacitors should ideally be placed as close as possible to the 69000. This means that the best decoupling will occur if the capacitors are placed directly underneath the component. If a single-sided board is required and capacitors cannot be placed underneath the component, then decoupling is recommended at the corners of the 69000. Placing the capacitors at the corners minimizes decoupling trace lengths from the BGA package, reducing EMI.

Avoid placing audio components near a switching power supply.

Digital signals should be isolated from analog circuitry as much as possible to keep digital and analog currents from crossing. Ground currents from digital signals are noisy and should be isolated from the audio signals. Do not run dynamic digital signals such as clock, address, or data lines in or close to the analog area.

All analog circuitry should be placed as close to I/O connectors as possible and should be isolated to as small an area as possible. Analog components and circuitry should reside over a separate ground and power plane with a gap between digital and analog planes made as wide as possible (25–50 mils is the acceptable minimum, 100 mils is ideal). The two plane pairs should be separated by a 2–3 mm gap. This means using at least a four-layer board with ground and power planes forming an internal high capacitive sandwich. This creates an effective series resistance (ESR) and effective series inductance (ESL) bypass capacitor.

Analog signals should reside over, or be referenced to, the analog ground plane as much as possible. Vias should be kept to a minimum. All analog signal traces should be as wide as possible for lower impedance.

Internal power and ground planes should be solid with no traces routed on them. The analog power plane containing the voltage regulator should coincide with the analog ground plane as much as possible. Analog ground shielding should be used on the external layers, especially near a low level input (such as a microphone).

The analog and digital planes should be connected at one point only through a 0 Ω resistor or a ferrite bead. This maintains the proper reference potential for each ground plane. On the board this should be placed as close as possible to the 69000. This allows analog and digital ground return currents from the 69000 to flow through the analog and digital ground plane, respectively.

There should not be any digital or analog traces crossing the gap between the analog and digital planes.

IC leads should have pads and vias that go directly to the appropriate plane for power and ground.

A separate small digital partition should be used for the crystal oscillator. This partition serves to keep noise from coupling onto the analog signals.

For information on how to interface to LCD Panels for the 69000 HiQVideo accelerator series, refer to *69000 HiQVideo Accelerator with Integrated Memory* datasheet (see Appendix A).

A jumper on VEESAFE has been added to allow VEESAFE to be driven by a scaled version of VDDSAFE (scaled by a 1 KΩ potentiometer) for newer panels that have a low voltage requirement for VEESAFE. It is important that the potentiometer be fed from VDDSAFE rather than some other voltage source so that the resulting VEESAFE will meet the same power sequencing requirements as the panel. The sequencing of VEESAFE is the same for VDDSAFE. The current requirement for a low-voltage VEESAFE is intended to be 1 mA maximum.

7.5.3 Analog Devices AD725 RGB to NTSC/PAL Encoder with Luma Trap Port

The Analog Devices AD725 is an RGB to NTSC/PAL encoder that may be populated on the board to add TV OUT capability. Follow these guidelines for implementation on the board.

- RIN, BIN, and GIN are analog inputs that should be terminated by 75 Ω resistors to ground in close proximity to the IC.
- The AD725 is a mixed signal part. It has separate pins for analog and digital +5 V and ground power supplies. Both the analog and digital ground pins should be tied to the ground plane by a short, low inductance path. Each power pin should have a bypass capacitor of 0.1 µF and a larger tantalum capacitor of 10 µF.
- The outputs have a DC bias that must be AC-coupled for proper operation. This is done on the board by placing a 220 µF tantalum capacitor and 75 Ω resistor in a series on each output.
- On the board, an RGB monitor is always populated. Therefore, when a TV encoder such as the AD725 is added, the RGB signals must be buffered. This is accomplished by an AD8073 video amplifier with high input impedances on each RGB signal. The signals are configured for a gain of two, which is normalized by the divide by two termination scheme used for the RGB monitor. However, since the RGB signals go to ground during horizontal synchronization, the AD8073 requires a -5 V supply.
- A jumper is placed on pin 5 of the AD725. Placing a jumper on pins 1–2 enables the AD725. A jumper on pins 2–3 disables the AD725.
- A jumper is also placed on pins 1 and 12. Placing a jumper on pins 1–2 enables NTSC, placing a jumper on 2–3 enables PAL. It is important that the oscillator circuit, placed on pin 3 of the AD725, also be changed when switching from NTSC to PAL (a 14.318180 MHz crystal is already placed for NTSC). The crystal must be changed from 14.318180 MHz to 17.734475 MHz for PAL.
- Proper layout of the AD725 circuitry is required for optimal performance.
- All passive components should be placed as close as possible to the AD725.
- A “fence” should be formed around the analog signals. *No* digital signals should be routed under or above the analog power and ground planes.
- The filter circuits for the video output signals (CMPS, LUMA and CRMA) should be placed as close as possible to the connector. Long lengths of closely-spaced parallel analog signals should be avoided. Wherever analog signals run in parallel, separated by less than 15 mils for longer than 250 mils, run a ground line between the video input traces of approximately 12 mils in width.
- The three analog inputs (RIN, GIN, BIN) should be terminated to ground by a 75 Ω resistor close to the respective pins.
- Layout guidelines should be followed as in the *69000 HiQVideo Accelerator with Integrated Memory* datasheet (see Appendix A).

7.6 Audio

The audio solution on the board is compliant with the AC ’97 Component Specification, v1.0 . It features an Ensoniq AudioPCI* 97 ES1371 digital controller and an Analog Devices AC ’97 SoundPort* Codec AD1819 connected through the 5-wire AC ’97 link.

The AC ’97 audio architecture provides legacy support as well as a migration path to digital audio. Sound Blaster* emulation is provided, as well as a MIDI data interface. The AD1819 provides an analog front end for high performance audio, modem, and DSP applications.

AC ’97 defines a high quality two-chip audio architecture (an AC ’97 digital controller and AC ’97 codec), advancing the migration to digital audio, while maintaining support for analog audio sources and analog interconnect for backwards compatibility. The two-chip audio solution, comprises a digital audio controller and a high quality analog component that includes digital-to-analog converters (DACs), analog-to-digital converters (ADCs), a mixer, and I/O. The architecture supports a wide range of high quality audio solutions, from a 2-channel mix of digital and analog audio, to multi-channel digital audio. The system is capable of achieving greater than 90 dB SNR performance.

7.6.1 Ensoniq AudioPCI* 97 ES1371 Digital Controller

The Ensoniq AudioPCI 97 ES1371 digital controller supports simultaneous stereo audio, and host-based wave table music synthesis through the integration of an AC ’97 2.0-compatible link and a PCI interface, in addition to audio legacy compatibility. The controller includes digital AC-link converters, WDM digital mixing acceleration, and variable sample rate conversion.

The PCI interface handles up to two digital audio streams, a modem, and handset streams. The digital audio streams are sample-rate converted and mixed to a common rate before being transmitted over the AC ’97 2.2-compatible link to an AC ’97 codec.

The sample rates are completely independent from the incoming and outgoing streams. The controller includes a variable sample rate converter that allows instantaneous support for sample rates ranging from 7 KHz to 48 KHz, with a resolution of 1 Hz.

SoundBlaster* emulation is provided through the combination of hardware digital mixing, software SoundBlaster trapping, and host-generated wave table music synthesis.

The primary interface for communicating MIDI data to and from the host is the hardware MPU-401 interface. The MPU-401 interface includes a built-in 64 byte FIFO for communication to the host bus.

The ES1371 digital controller is PCI device number 16 (IDSEL connector to AD27 through a 220 Ω resistor). It is also PCI bus master #4 (connected to PCI bus master pins REQ4# and GNT4#) and uses PCI interrupt INTC# on the board.

Note: The MIDI/JOYSTICK interface is not implemented on the board.

7.6.2 Analog Devices AC ’97 AD1819 SoundPort* Codec

The AD1819 SoundPort codec is designed to meet all requirements of the Audio Codec Component Specification, v1.03. The AD1819 supports multiple codec configurations (up to three per AC link), a DSP serial mode, variable sample rates, modem sample rates and filtering, and built-in Phat Stereo 3-D enhancement.

The AD1819 is an analog front end for high performance audio, modem, or DSP applications.

The main architectural features of the AD1819 are the high quality analog mixer section, two channels of sigma-delta ADC conversion, two channels of sigma-delta DAC conversion, and data direct scrambling (D2S) rate generators. The AD1819’s left channel ADC and DAC are compatible for modem applications supporting irrational sample rates and modem filtering requirements.

The AD1819 codec is connected to the ES1371 through the 5-wire AC ’97 link. The RESET# signal of the 5- wire AC ’97 link is connected to PCI RESET (RST#) on the board.

The board implements microphone, video, line, and CDROM inputs as well as line, mono and line level outputs. The AD1819 provides interfaces for PC beep, phone input, and auxiliary inputs that are not implemented on the board.

There are various filter pins on the AD1819. These pins are listed below with their implementation on the board.

Table 3. AD1819 Codec Filter Pins and Implementations

Pin Name	TQFP	I/O	Description	Board Implementation
AFILT1	29	O	Antialiasing Filter Capacitor - ADC Right Channel	270 pF ceramic capacitor-to-analog ground
AFILT2	30	O	Antialiasing Filter Capacitor - ADC Left Channel	270 pF ceramic capacitor-to-analog ground
FILT_R	31	O	AC-Coupling Filter Capacitor - ADC Right Channel	1 µF tantalum capacitor-to-analog ground
FILT_L	32	O	AC-Coupling Filter Capacitor - ADC Left Channel	1 µF tantalum capacitor-to-analog ground
RX3D	33	O	3D Phat Stereo Enhancement - Resistor	47 nF capacitor-to-analog ground
CX3D	34	I	3d Phat Stereo Enhancement -	100 nF capacitor connected to RX3D

Note: Proper board layout of the AC ’97 audio components is important to achieving optimal performance. Avoid placing the audio components near a switching power supply.

To keep digital and analog signal currents from crossing, digital signals should be isolated from analog circuitry as much as possible. Ground currents from digital signals are noisy and should be isolated from the audio signals. Do not run dynamic digital signals such as clock, address or data lines in or around the analog area.

All analog circuitry should be placed as close to I/O connectors as possible and should be isolated to as small an area as possible. Analog components and circuitry should reside over a separate ground and power plane with as wide a gap as possible between digital and analog planes (25–50 mil is acceptable and 100 mil is ideal). The two plane pairs should be separated by a 2–3 mm gap. This means using at least a four-layer board with ground and power planes forming an internal high capacitive sandwich. This gives an effective low ESR and ESL bypass capacitor.

Analog signals should reside over, or be referenced to, the analog ground plane as much as possible. Vias should be kept to a minimum. All analog signal traces, especially VREFOUT and analog power lines on the AD1819, should be as wide as possible for lower impedance connections.

Internal power and ground planes should be solid with no traces routed on them. The analog power plane containing the voltage regulator should coincide with the analog ground plane as much as possible. Analog ground shielding should be used on the external layers, especially near the microphone input.

The analog and digital planes should be connected at one point only, through a 0-Ω resistor or a ferrite bead. This maintains the proper reference potential for each ground plane. On the board, this should be placed as close as possible to the AD1819 and its voltage regulator. This allows the AD1819's analog and digital ground return currents to flow through the analog and digital ground plane respectively.

Note: No digital or analog traces should cross the gap between the analog and digital planes.

IC leads should have pads and vias that go directly to the appropriate plane for power and ground. A separate small digital partition should be used for the crystal oscillator. This partition serves to keep noise from coupling onto the analog signals.

7.7 SMSC FDC37B78x Ultra I/O*

The Standard Microsystems Corporation (SMSC) FDC37B78x Ultra I/O provides keyboard, mouse, and floppy disk port control on the board. It also provides two serial ports: COM1 is shared with the touch screen interface, and COM2 is shared with the infrared interface. The Ultra I/O resides on the ISA bus and provides twelve IRQ options with full 16-bit address decode.

The FDC37B78x Ultra I/O provides support for the ISA plug-and-play Standard, v1.0a. The I/O address, DMA channel, and IRQ channel of each FDC37B78x logical device can be programmed through internal configuration registers. There are 480 I/O address location options, 12 IRQ options or serial IRQ options, and four DMA channel options for each logical device.

7.7.1 Floppy Disk Controller (FDC)

The FDC37B78x provides the interface between a host microprocessor and the floppy disk drives. The FDC integrates the functions of the formatter/controller, digital data separator, write precompensation, and data rate selection logic for an IBM PC XT/AT-compatible FDC. The true CMOS 765B core is 100% compatible with the IBM PC XT/AT in addition to providing data overflow and underflow protection. The FDC is also compatible with the 82077AA using SMSC's proprietary floppy disk controller core.

7.7.2 Serial Port Controller

The FDC37B78x incorporates two full function UARTs. They are compatible to the NS16450, the 16450 ACE registers, and the NS16C550A. The UARTs perform the necessary serial-to-parallel conversion on received characters and the parallel-to-serial conversion on transmitted characters. The data rates are programmable from 460.8 Kbaud to 50 baud. The character options are programmable for 1 start; 1, 1.5 or 2 stop bits; even, odd, sticky or no parity; and prioritized

interrupts. Each UART contains a programmable baud rate generator that is capable of dividing the input clock or crystal by any number in the range of 1 to 65535. The second UART supports IrDA 1.0, HP-SIR, ASK-IR, and Consumer IR infrared modes of operation.

7.7.3 Infrared Interface

The SMSC FDC37B78x provides infrared support, including support for Consumer IR and IrDA, v1.0. The interface provides a two-way wireless communications port. Possible IR implementations for the second UART in this chip (logical device 5) include IrDA, Consumer Remote Control, and Amplitude Shift Keyed IR. The IR transmission can use the standard UART2 pins TXD2 and RXD2, or can optional use pins IRTX and IRRX.

Serial communication baud rates up to 115.2 Kbps are provided for through IrDA, v1.0. The Amplitude Shift Keyed IR allows an asynchronous baud rate of up to 19.2 Kbaud.

COM1 is shared with the touch screen. Three 3-pin jumpers are provided to switch from COM1 to touch screen and vice versa. Placing a jumper on pins 1–2 on all three jumpers enables the touch screen. Placing a jumper on pins 2–3 enables COM1.

COM2 is shared with the infrared interface. IRTX2 and IRRX2 of the Ultra I/O directly connect TXD and RXD of the TEMIC TFDU4100 infrared transceiver respectively. The functionality of this COM port must be switched in the BIOS. The board uses the FDC37B78x's alternate IRTX2 and IRRX2 pins, allowing BIOS to switch from COM2 to the infrared interface.

7.7.4 Parallel Port Controller

The parallel port controller is compatible with the IBM PC XT/AT. It supports the optional PS/2-type bi-directional parallel port (SPP) mode, the enhanced parallel port (EPP) mode, and the extended capabilities port (ECP) mode.

7.7.5 Keyboard and Mouse

The universal keyboard controller uses an 8042 microcontroller processor core.

7.7.6 Design Note for Ultra I/O*

Each power pin must be individually decoupled with a 0.1 µF capacitor. Since the real time clock on the Ultra I/O* is not used, VBAT must be tied to ground. If not grounded, this pin may float and confuse the internal circuitry that is trying to detect whether power is valid.

The Ultra I/O provides a BIOS interface. This is not implemented on the board; therefore ROMCS# must be pulled high to disable the ROM buffers and avoid interference with the boot ROM.

The SYSOPT pin is latched on the falling edge of RESET_DRV or on V_{CC} Power On Reset to determine the configuration register's base address. The SYSOPT pin is used to select the CONFIG PORT's I/O address at power-up. Once powered up, the configuration power base address can be changed through configuration registers. See Table 4 for SYSOPT settings.

Table 4. SYSOPT Pin Settings

Port Name	SYSOPT = 0 (1 KΩ pull-down resistor)	SYSOPT = 1 (10 KΩ pull-up resistor) Board Setting
CONFIG PORT	0x03F0h	0x0370h
INDEX PORT	0x03F0h	0x0370h
DATA PORT	INDEX PORT + 1	INDEX PORT + 1

Note: If using TTL RS232 drivers, use a 1 KΩ pull-down resistor. If using CMOS RS232 drivers, use 10 KΩ pull-down resistor. The board uses TTL drivers, therefore SYSOPT should be pulled down with a 10 KΩ pull-up resistor. Therefore, the CONFIG PORT and INDEX PORT for the FDC37B78x are located at 0x0370h after reset.

The setting on SYSOPT *must* be different than SYSOPT on the SMSC FDC37C669 Super I/O in order for the two chips to configure at different addresses. On the board, the FDC37C669 SYSOPT is pulled low with a 1 KΩ resistor. Therefore, the FDC37C669 is located at 0x0370h in memory after reset. The two options are shown in Table 5.

Table 5. SYSOPT Setting Options

Option	FDC37B78x SYSOPT Pin	FDC37C669 SYSOPT Pin
A	SYSOPT = 1 (pulled HIGH with 10 KΩ resistor)	SYSOPT = 0 (pulled LOW with 1 KΩ resistor)
B	SYSOPT = 0 (pulled LOW with 1 KΩ resistor)	SYSOPT = 1 (pulled HIGH with 10 KΩ resistor)

7.7.7 Additional Serial Ports with SMSC 37C669 Super I/O*

An additional Super I/O, SMSC 37C669 is used on the board to add two extra serial ports, COM3 and COM4. The parallel, IDE, and floppy interfaces are not used on this design. This configuration was chosen over separate 16550's to limit cost and conserve board space.

Note: Each power pin must be individually decoupled with 0.1 µF capacitors.

All inputs to unused devices (i.e., parallel, IDE and floppy) are pulled to their inactive state. The value of the pull-up/pull-down resistor is chosen to keep the inputs within 0.5 V of the supply rails in order to minimize current consumption. Each pin has 10 µA of current leakage.

The FDC37C669 provides 11 decoded address lines, AEN, plus one chip-select input. Since the FDC37C669 can only decode the lower 11 address bits and AEN, an address decoder is needed for A[15:11]. An external gate such as an OR gate could be used to decode addresses A[15:11]. The output of the OR-gate is low when A[15:11] are low, thereby asserting the FDC37C669 chip select. Instead of the external address decoder, this design uses one of the two programmable chip select lines on the PIIX4E. The registers that control PCS0#, the first programmable chip select output of the PIIX4E, must be properly initialized by BIOS. The PCS0# output is tied to the Super I/O's chip select. This chip select, AEN, and 11-bit address decode provides the full 16-bit ISA decode necessary.

At the trailing edge of a hardware reset, the SYSOPT input is latched to determine the configuration base address. Refer to Table 6 for SYSOPT settings.

Table 6. SYSOPT Settings

Port Name	SYSOPT = 0 (1 KΩ pull-down resistor)	SYSOPT = 1 (10 KΩ pull-up resistor)
INDEX Base I/O Address	0x03F0	0x0370

Note: If you are using TTL RS232 drivers, use a 1 KΩ pull-down resistor. If using CMOS RS232 drivers, use a 10 KΩ pull-down resistor. The board uses TTL drivers, therefore SYSOPT should be pulled down with a 1 KΩ resistor or pulled up with a 10 KΩ resistor. SYSOPT is pulled low on the board; therefore the FD37C669 is located at 0x03F0h in memory after Reset.

The setting on SYSOPT *must* be different than SYSOPT on the FDC37B78x Ultra I/O in order for the two chips to configure at different addresses. The two options are listed in Table 7.

Table 7. SYSOPT Setting Options

Option	FDC37B78x SYSOPT Pin	FDC37C669 SYSOPT Pin
A	SYSOPT = 1 (pulled HIGH with 10 KΩ resistor)	SYSOPT = 0 (pulled LOW with 1 KΩ resistor)
B	SYSOPT = 0 (pulled LOW with 1 KΩ resistor)	SYSOPT = 1 (pulled HIGH with 10 KΩ resistor)

NOTE: On the board, SYSOPT on the FDC37B78x is pulled HIGH and SYSOPT on the FDC37C669 is pulled LOW.

7.8 Tritech Microelectronics TR88L803 Touch Screen Controller

TriTech Microelectronics TR88L03 Touch Screen Controller is used to implement a touch screen solution. The TR88L03 is a fully integrated pen input processor with two multiplexed A/D input channels. The chip uses a serial interface, and there is no user configuration required.

The low-power TR88L803 contains all the circuitry required to interface the low cost 4- and 5-wire resistive digitizers to applications and provide pen input capability. The TR88L803 uses 10-bit ADCs to resolve 1024 levels. When clocked at 4 MHz, the TR88L803 is designed to simplify pen interfacing by integrating all pen input tasks required to present X, Y position data to the main application at 200 coordinate pairs per second. The TR88L803 provides standard asynchronous serial output or synchronous serial output. The TR88L803, with enhanced noise filtering, is ideally suited for operation in electrically noisy environments.

7.8.1 Implementation of the TR88L803

The touch screen controller shares resources with COM1 on the board. To use either COM1 or the touch screen, the board must be jumpered correctly.

The TR88L803's serial data output is connected to the serial port TX pin through two general-purpose transistors. The transistors are properly biased to provide the necessary signal level swing for the TX pin. The negative signal level is derived with the RX pin of the serial port. RX is not used since data is unidirectional for the touch screen controller.

The touch screen controller provides two independent ADC input channels under the ADC multiplex mode. Control pin MUX_SEL multiplexes the internal ADC between serving the digitizer inputs (X+, X-, Y+, Y-) and the two independent ADC input channels. These channels are not implemented on the board; therefore ADC_1 and ADC_2 are tied to the analog ground through a 10-KΩ resistor and MUX_SEL is tied to ground.

The touch screen controller provides a 4-wire interface for the Dynapro* touch screen.

Mixed signal layout guidelines should be followed as discussed in the LCD/Video and Audio sections.

7.8.2 LCD with Integrated Touch Screen

The touch screen controller provides a four-wire interface for the Dynapro touch screen. The LCD interface consists of two connectors: Flat Panel Connector 1 and Flat Panel Connector 2. The Flat Panel Connector 1 provides an interface to LCD screens with up to 24-bit color. For 36-bit color, both Flat Panel Connector 1 and Flat Panel Connector 2 must be used.

7.9 PCMCIA (Intel® Strataflash™ Memory and PC Card) Socket

This reference design uses the Texas Instruments PCI1225 PC Card Controller, supporting both an external PCMCIA card socket as well as on-board Intel® StrataFlash™ memory. The controller operates on a 3.3 V core and is PCI interface compatible with 3.3 V and 5.0 V PCI signaling environments. Supporting up to five general purpose I/Os, the controller is compliant with both the *PCI Local Bus Specification*, Revision 2.1 and 1995 PC Card Standards. The Fujitsu Takamisawa Americas 565P068-G/J-4V socket is compatible with PCMCIA card type I, II, and III.

7.9.1 Texas Instruments PCI1225 PC Card (PCMCIA) Controller

The Texas Instruments PCI1225 is a PCI-to-PC card controller that supports two independent PC card sockets, compliant with the 1995 PC Card Standards. The 1995 PC Card Standards retain the 16-bit PC card specification defined in PCMCIA, v2.1, and defines the new 32-bit PC card, called CardBus*, capable of full 32-bit data transfers at 33 MHz. The PCI1225 supports any combination of 16-bit and CardBus PC cards in the two sockets, powered at 5 V or 3.3 V.

The PCI1225 is compliant with the *PCI Local Bus Specification*, Revision 2.1 and with the PCI Bus Power Management Interface Specification. The PCI1225 can act as either a PCI master or slave device with PCI bus mastering initiated during CardBus PC card bridging transactions.

7.9.1.1 Slot A: PCMCIA Socket with Texas Instruments TPS2206 PC Card Power-Interface Switch with Reset

Slot A of the PCI1225 is connected to a PCMCIA socket for PCMCIA cards on the board. The Texas Instruments TPS2206 provides voltage regulation, over-current and over-temperature protection for the PCMCIA socket. The TPS2206 also accommodates 3.3 V/5 V systems by first powering-up the PCMCIA card with 5 V, then polling it to determine its 3.3 V compatibility.

The TPS2206 also provides a reset to the PCMCIA card. This reset is triggered by a PCI RESET signal (RST#) on the board.

7.9.1.2 Slot B: Intel® StrataFlash™ Memory

Capitalizing on two-bit-per-cell technology, Intel StrataFlash memory products provide 2x the bits in 1x the space. Offered in 64-Mbit (8-Mbyte) and 32-Mbit (4-Mbyte) densities, Intel StrataFlash memory devices are the first to bring reliable, two-bit-per-cell storage technology to the flash market.

Intel StrataFlash memory benefits include more density in less space, the lowest cost-per-bit NOR devices, support for code and data storage, and easy migration to future devices.

All card signals are internally buffered to allow hot insertion and removal without external buffering. The PCI1225 is register compatible with the Intel 82365SL-DF ExCA controller. The PCI1225 internal data path logic allows the host to access 8-, 16-, and 32-bit cards using full 32-bit PCI cycles for maximum performance. Independent buffering and a pipeline architecture provide high performance with sustained bursting. The PCI1225 can also be programmed to accept fast posted writes to improve system bus utilization.

General purpose inputs and outputs are provided to implement sideband functions. Many other features are designed into the PCI1225, such as socket activity LEDs.

A CMOS process is used to achieve low system power consumption while operating at PCI clock rates up to 33 MHz. Power consumption is further reduced by several low-power modes.

7.9.2 Design Notes for the PCI1225

The Texas Instruments PCI1225 is a PCI-to-PC card controller. The controller provides two PC card slots. Slot A is connected to a PCMCIA socket for PCMCIA cards on the board. Slot B of the PC card controller provides the interface to Intel StrataFlash memory.

The PCI1225 PC card controller is PCI bus master #3 (connected to PCI bus master pins REQ3# and GNT3#) on the board. It is also PCI device number 5 (IDSEL connected to AD16).

The PCI1225 provides the user with flexibility in determining the interrupt implementation. The interrupt mode is selected via bits [2:1] of the Device Control Register at PCI offset 92h. Other registers that must also be configured are described below. The interrupt implementations are listed in Table 8.

The board implements parallel IRQ and PCI interrupts. In this interrupt mode, the PCI1225 routes the legacy interrupts, IRQ[15:2], via the seven multifunction terminals MFUNC[6:0]. The parallel IRQ and PCI interrupt modes are selected by programming bits [2:1] of the Device Control Register at PCI offset 92h to a value of 01b. When this mode is selected, the multifunction terminals must be configured through the Multifunction Routing Register at PCI offset 8Ch.

Table 8. IRQ and PCI Interrupt Implementations

Interrupt Mode	PCI Offset 92h Bits 2:1
Parallel PCI Interrupts Only	00
Parallel IRQ and PCI Interrupts	01
Serial IRQ and Parallel PCI Interrupts	10
Serial IRQ and PCI Interrupts	11

The PCI interrupts INTA# and INTB# are available only on terminals MFUNC0 and MFUNC1, respectively. A maximum of five IRQ interrupts can be implemented through multifunction terminals MFUNC[6:2].

The multifunction terminals are connected on the board as listed in Table 9.

Table 9. Multifunction (MFUNC[6:0]) Board Connections

Terminal Pin	Connection on Board
MFUNC0	INTC#
MFUNC1	INTD#
MFUNC2	IRQ4
MFUNC3	IRQ5
MFUNC4	IRQ9
MFUNC5	IRQ10
MFUNC6	IRQ15

7.10 ISA PCI Expansion Cards

There are two PCI expansion slots on this sample design, allowing for add-in card peripherals to enhance any design. There is also one ISA expansion slot in the design. With the variety of peripherals in this design that would ordinarily be implemented through an add-in card, two extra PCI slots and one extra ISA slot allows for functionality that has not been added.

- ISA Bus/Expansion Slots

  As described in the *Intel® 440BX PCIset Design Guide* (order number 290613), pull-up and pull-down resistors should be placed as follows:

  — 10 KΩ pull-ups on SD[15:0] and SA[19:0]

  — 1 KΩ pull-up on IOCHRDY and REFRESH#

  — 10 KΩ pull-up on IRQx (10 KΩ pull-up on IRQ8, MEMR#, MEMW#, IOR#, IOW#, LA[23:17], SMEMR#, SMEMW#, SBHE#, and BALE

  — 330 Ω on ZEROWS#, MASTER#, MEMCS16#, and IOCS16#

  — 4.7 KΩ on IOCHK#

  — 5.6 KΩ pull-down resistors on DRQx

- PCI Bus/Expansion Slots

  As described in the *Intel® 440BX PCIset Design Guide*, place pull-up signal resistors on the following:

  — 2.7 KΩ pull-up resistors to 5 V on PIRQ[D:A]#, SDONE, SBO#, FRAME#, TRDY#, STOP#, IRDY#, DEVSEL#, PLOCK#, PERR#, SERR#, REQ64# and ACK64# and PAR

  — 10 KΩ pull-ups to 3.3 V on GNT[3:0]#, PHLD# and PHLDA#

  There are two PCI expansion slots on the board: Slot 0 is PCI device #17; Slot 1 is PCI device #18 (IDSEL connected to AD28 and AD29 through a 220 Ω resistor respectively). Each slot is connected to PCI bus master arbitration pins REQ# and GNT#. Slot 0 is connected to GNT0# and REQ0#, and Slot 1 is connected to GNT1# and REQ1#.

Note: For layout considerations when placing series resistors, refer to the *Intel® 440BX PCIset Design Guide.*

Interrupts on the board are configured as indicated in Table 10 to minimize interrupt latency.

Table 10. Board Interrupt Configurations

Device Interrupt Pin	LCD / Video Controller	AC '97 Audio	PCMCIA†	PCI Slot 0	PCI Slot 1
0	INTA#	INTB#	INTC#	INTD#	INTA#
1	X	X	INTD#	INTA#	INTB#
2	X	X	X	INTB#	INTC#
3	X	X	X	INTC#	INTD#

† The PCMCIA controller also uses legacy ISA IRQs. Refer to Section 7.9, "PCMCIA (Intel® Strataflash™ Memory and PC Card) Socket" on page 38 for more information.

Note: Refer to the *PCI Local Bus Specification*, Revision 2.1, for routing guidelines (see Appendix A).

7.11 Clocking

Improper layout of the clock lines on the board can cause cross-coupling with other signal lines. It is recommended that the clock line spacing (air gap) be at least two times the trace width of any surrounding traces. It is also strongly recommended that the clock spacing be at least four times the trace width of any strobe line.

7.11.1 Cypress Semiconductor CY2280

The CY2280 is a clock synthesizer/driver that outputs four processor clocks at 2.5 V. There are eight PCI clocks that run at one-half or one-third the processor clock frequency of 66.6 MHz and 100 MHz respectively. One of the PCI clocks is free-running. The CY2280 possesses power-down, processor stop, and PCI stop pins for power management control. The signals are synchronized on-chip and ensure glitch-free transitions on the outputs.

The CY2280 outputs are designed for low EMI emissions. For further information, please refer to the following URL:

http://www.cypress.com/cypress/prodgate/timi/cy2280.htm.

7.11.2 Cypress Semiconductor CY2318

The CY2318 is a 3.3 V SDRAM buffer designed to distribute high speed clocks in PC systems and SDRAM modules. There are on-chip PLLs that lock to an input clock. It can be used to drive up to four SDRAM DIMMs. An I^2C interface can be used to disable unused output clocks. For more information please refer to the following URL:

http://www.cypress.com/cypress/prodgate/modu/cy2318nz.htm.

7.11.3 Clocking for 440BX Chipset

The Intel 440BX chipset uses several techniques from the mobile and desktop markets to reduce power consumption — hence power dissipation — in a Celeron processor-based embedded system.

In the 440BX there are three main states used in clocking:

- Clock Stopped: CLKRUN# is being monitored for a restart, the clock is stopped.
- Imminent Clock Stop: The system has indicated the clock is about to be halted utilizing the CLKRUN# line.
- Clock Running: The clock is running and the PCI system bus is operational.

The 440BX is a CLKRUN# master unit and behaves according to the protocols specified for a master control device. The PIIX4E companion device to the chipset controls system clocks and is the CLKRUN# central resource. In designing a system, the main system clock should originate from a master clock oscillator, and for the processor and 82443BX, be from the same buffer driver output.

The signal paths must be setup to minimize clock skew and the resulting system instabilities. Trace length matching, along with proper terminations to minimize reflected energy, will result in better system performance and less overall noise that could otherwise affect system reliability.

When the Intel 440BX chipset is used in a system application, the designer may elect to take advantage of the different clocking schemes and modes available to enhance system power and performance. There are five low power clocking modes available:

- Chip Standby: The 440BX, processor, and PCI buses are idle.
- Dynamic Stop Clock: Assists the processor in transitioning in and out of clock stop from system run.
- Powered On Suspend: All system phase locked loops (PLL) are shut down. The real time clock (RTC) and suspend clock (SUSCLK) are running. DRAM refresh is accomplished by using SUSCLK.
- Suspend to Ram (STR): The processor and L2 cache are clocked off. RTC and SUSCLK are running. The DRAM refreshes using SUSCLK.
- Suspend to Disk (STD): The processor and L2 cache, DRAM, and PCI interface are clocked off.

In clocking the main system, the PCI bus clock is derived from the main processor clock and is one half its frequency. For SDRAM, the clock is synchronized with BXDCLKO and BXDCLKRD.

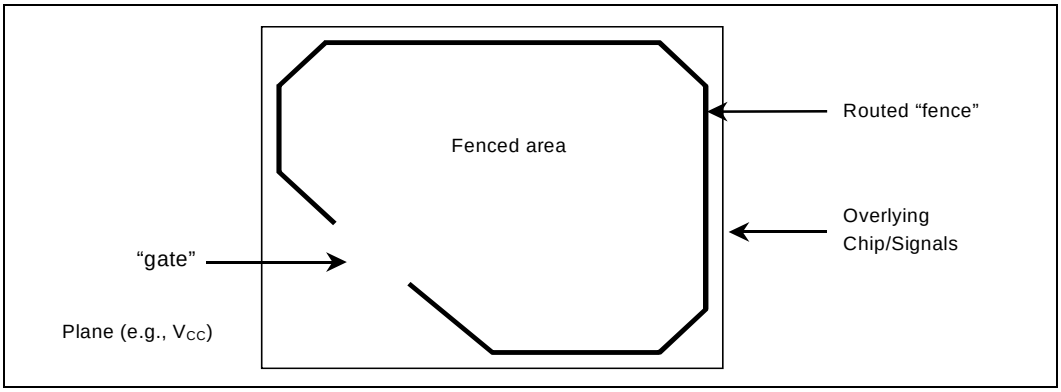
8.0 Design Considerations

High-speed designs such as the design in this application note require careful PCB layout and signal routing for trouble-free operation. Detailed routing notes and guidelines can be found in the schematics in Appendix B. Decoupling capacitors are shown where needed in the schematics. Decoupling capacitors provide a short between power and ground for high frequency noise signals.

- If a part is removed from the design, the outputs can be left unconnected, but the inputs should be pulled either high or low through an appropriate resistor.
- When changing jumper settings, first power down the board.
- Trace widths on all power and ground signals should be ~10 mil.

A fence is a line routed out of a plane such that a given area is isolated from the rest of the plane except at a single point of contact, conceptually the “gate” in the fence. A fence minimizes noise originating from the digital signaling onto the analog signals. This provides higher quality video or audio. An example is shown in Figure 5. The heavy black line is the routed area. The width of the gate or opening can be up to 75% of the length of the integrated circuit or signals in question. The width of the routed fence should conform to the separation routing between power planes (i.e., 25 mils minimum).

Figure 5. Fence Example



9.0 Windows* CE

Windows CE can have a similar look and feel as Microsoft Windows 98 or Windows NT. Experienced Windows programmers can write CE applications with little additional training, since Windows CE applications use Win32 Application Programming Interface (API) calls that are almost identical to those used in applications written for Windows 98/NT. A variety of interfaces are supported for embedded applications. The hardware configuration for an embedded system can include peripherals such as a keyboard, mouse, touch screen, and others. It is often easier for users to use commands and keys that are similar to those on their desktop PC in their office or home. A similar environment in the embedded product segment eliminates the need for retraining personnel, since they are already familiar with the look and feel of the system.

With Windows CE, the PC is used to prototype peripherals and develop device drivers; all development is done on the PC. If the target platform is also Intel architecture, the large knowledge base for Intel architecture can be utilized for hardware expertise, programmers, training materials, and sample code. Intel architecture is very efficient for all Windows operating systems. Since a lot of optimization effort has already been spent on the PC, reuse of PC developments can result in faster time-to-market. This is possible if the architecture of the Windows CE target is the same as the PC. The Windows CE Embedded Toolkit includes PC chipset and peripheral support, including basic VGA drivers and S3 accelerated drivers.

There are some differences between programming for the desktop and programming in Windows CE. Since Windows CE programs are required to have a small memory footprint, an important aspect of its programming is the need to manage with limited memory. Since memory usage must be minimized, having knowledge of memory allocations with Windows CE is important.

Embedded systems may stay powered-on for long periods of time. Programmers need to keep this in mind to ensure that the programs run flawlessly for long durations. Modifications must be made to the OEM Adaptation Layer. Since different solutions are possible for different applications, the applications' varying needs can be met.

9.1 Windows CE for Embedded Applications

Windows CE is designed to be a multi-threaded, preemptive, multitasking operating system for platforms with limited resources. This is to say that multiple processes can be running simultaneously, each assigned a specific priority. Each process can have multiple threads. This allows a process to have more than one flow of execution running concurrently. Windows CE is designed as a modular operating system that can be customized to contain only the modules needed by the application. Refer to Appendix A for the URL for Microsoft's Windows CE Toolkit (ETK) for Visual C++* and the newer CE development tool, Platform Builder. Platform Builder for Windows CE includes a version of the C++ compiler optimized for embedded systems. It also has a more graphical user interface and wizards that automate many of the CE build steps.

9.2 Windows CE Environment

9.2.1 Kernel

The Windows CE kernel contains the core operating system functionality that must be present on all Windows CE-based platforms. It includes support for memory management, process management, exception handling, multitasking and multithreading.

The kernel is provided as the file Nk.lib. The OEM Adaptation Layer must be developed to create the platform-specific Nk.exe module. For an overview of the OEM Adaptation Layer, refer to Section 9.2.3.

The kernel is designed specifically for small, fast, embedded devices. The kernel supports only a single 4 Gbyte address space (a 2 Gbyte virtual address and a 2 Gbyte physical address range). Of the 2 Gbyte virtual address space, 1 Gbyte is divided into 33 slots, each of which is 32 Mbytes. The kernel protects each process by assigning it to a unique open slot in memory. Within each slot, memory regions are allocated for the following:

- 64 Kbytes of reserved space
- Executable files, code, and data sections
- The primary thread's stack, and the process' default heap
- Dynamic-link libraries (DLLs) loaded from RAM
- DLLs loaded from ROM

The kernel protects applications from accessing memory outside of their allocated slot by generating an exception. Applications can check for and handle such exceptions by using the "try and except" Windows CE functions. The system is limited to 32 processes, but the number of threads running in a process is limited only by the amount of available memory.

9.2.2 Modularity

Windows CE is built from a number of discrete modules; therefore the size (footprint) of the operating system software can be controlled by selecting only the applicable modules. Several of these modules are also further divided into components. By selecting a minimum set of modules and components, the ROM and RAM requirements needed to support the end product can be minimized.

Refer to *Understanding Modularity in Microsoft Windows CE* (see Appendix A).

9.2.3 OEM Adaptation Layer (OAL)

A developer can adapt Windows CE for a specific target platform by creating a thin layer of code that resides between the kernel and the target platform. To avoid confusion with the Windows NT Hardware Abstraction Layer (HAL), Windows CE refers to this interface as the OEM Adaptation Layer, or OAL.

In addition to managing functions such as timing and power, the primary purpose of the OAL is to expose the target platform's hardware to the kernel. That is, each hardware interrupt request line (IRQ) is associated with one interrupt service routine (ISR). When interrupts are enabled and an interrupt occurs, the kernel calls the registered ISR for that interrupt. The ISR, the kernel mode portion of interrupt processing, is kept as short as possible. Its responsibility is primarily to direct the kernel to schedule and launch the appropriate interrupt service thread (IST). The IST, implemented in the device driver software module, gets or sends data and control codes to the hardware and acknowledges the device interrupt.

9.3 Windows CE OS Build Information

9.3.1 Getting Started

When installing all the necessary software (either Platform Builder or Visual C++, SDK, and ETK), it is advisable to install support for only x86-based processors to save disk space. Find the correct build environment for the project you are going to develop. In Platform Builder this will be the CEPC platform. Wizards are used to automate many steps in building a CE platform. Several examples are provided in the Visual C++ and the CE toolkits (e.g., Maxall, Minshell). Newer versions of CE support faster downloading of the target system via an ethernet cable. A specially configured low-cost PC called a PC reference platform can be used as a CE target system for initial software development before final hardware becomes available. Details can be found in the CEPC reference platform document listed in Appendix A. PC based CE target system emulation is also possible with the Windows CE DDK.

9.3.2 Building a Windows CE OS with Platform Builder

After installing Platform Builder and X86 support, start Platform Builder, then follow these steps:

1. Choose File - New to create a new platform workspace.

   Each workspace uses several hundred megabytes of disk space, so make sure there is plenty of space available on the destination drive.

2. Give the workspace a name, and click OK.
3. Choose the CEPC board support package. The closest choice for this design's platform is the Maxall platform.
4. Click Finish.

   Make sure that the Win32 (X86) Debug or Release option is selected in the command bar.

There are several build options and drivers that are selected with environment variables. These dependencies are included in the *.bib files associated with the platform. To view the *.bib files click on the Parameter view tab at the bottom of the left window. Click on the Component tab to return to the component view.

The generic vga8 driver works with a wider variety of VGA controller chips. The environment variable, CEPC_DDI_VGA8BPP, controls VGA device driver selection and should be added and set to 1 to use the basic VGA driver instead of the default S3 trio VGA driver. To change or add environment variables, select Platform, Settings, then click the Environment tab.

You must also copy the VGA driver file ddi_vga8 to the workspace. You can copy by dragging the ddi_vga8 file from the right catalog Windows display folder to the workspace window on the left.

To build the platform, select Build and then Build Platform. It takes several minutes to compile all of the programs and build the new CE OS image. Monitor the build process by watching the output window that appears in the center of the screen and verify that there are no errors.

If no errors occur, the new OS image is contained in the file NK.BIN. This is the file that is downloaded to the target system. Prior to downloading the board, the programs ESHELL, CETERM, and CESH must be running with the appropriate cable and network download options and the cables must be properly connected. A floppy disk drive can be used to load the CE bootstrap loader on the target system. Applications for the new platform can be developed using the project feature of Platform Builder. In addition to the project build step, new applications will require an entry in the platform's *.bib file to include the application's module(s) in the OS build

process. An SDK for applications programming can also be generated using Platform Builder. Additional details can be found in the *Windows CE Platform Builder 2.12 Getting Started Guide* listed in Appendix A.

9.3.3 Building a Windows CE OS with the CE Toolkit

In Windows CE 2.1 and higher there are two preconfigured platforms called Maxall or Minshell. In Windows CE Toolkit 2.0 the various platform demos are named DEMO*X*, where *X* is an integer. After accessing the desired build environment for your system, type “blddemo” to build the image for that project. This places the image in the release directory. After building the image, load it into the target system using the Parallel Port Transfer Tool (PPSH) or in newer versions the CE Shell Utility (CESH) and ESHELL. A floppy disk drive can be used to load the CE bootstrap loader on the target system. CETERM is used to monitor debug messages from the target system. The configurations included with the Windows CE Embedded Toolkit for Visual C++ contain all the modules and components needed to build different versions of a Windows CE operating system.

Blddemo.bat, a build batch file included in the directory %_PUBLICROOT%\Common\Oak\Misc, automates the Build Tools required to generate the platform for your Windows CE operating system. For each configuration, a build batch file %_TGTPROJ%.bat (for example Minkern.bat) specifies the environment variables required for your Windows CE project. The following procedure describes how to configure your build environment and use Blddemo.bat to create the Windows CE operating system image for the operating system configuration.

To build a Windows CE operating system configuration:

1. If the configuration includes the Windows CE debug shell utility (CESH), invoke the Microsoft WinDbg debugging program shortcut for your platform (e.g., x86 Debugger).
2. Invoke the new command prompt build window shortcut that you created in Preparing to Build a Windows CE Operating System configuration.
3. As an optional step, modify your command prompt build window to add scroll bars. Scroll bars let you scroll back through the commands that the various batch files and tools execute. Click on the MS DOS icon in the top-left corner of the window, then click on Properties. Click on Layout and change the screen buffer size and height setting to 1000. This modification sets a 1000-line screen buffer so you can see up to 1000 of the most recent lines in that command window. After clicking OK, select the Modify shortcut that started the window to make this change permanent.
4. Set any additional environment variables for the configuration build in your command prompt build window. For example, if you want to build the Windows CE kernel with remote debugging enabled, use the following command:

   ```
   set IMGNODEBUGGER=
   ```

   To compile a configuration so that debugging messages are enabled, use the following command to set the environment variable WINCEDEBUG to *debug*:

   ```
   set WINCEDEBUG=debug
   ```

   For information on configuration-specific environment variables, see the corresponding description of the Windows CE operating system configuration in Section 9.3.3.
5. Enter the build batch file command to build the configuration *blddemo*.

   After the build batch file is done processing, the current directory should still be the %_WINCEROOT% directory.

6. Verify that the build batch file command completed successfully by checking whether the Nk.bin file exists in the directory %_FLATRELEASEDIR% (by default, \Wince211\Release). Nk.bin contains the binary image of the operating system for your selected platform.

9.3.4 Building a New Project

This section provides a tutorial that describes how to build the Windows CE operating system image based on the Maxall configuration. In this section, you will:

- Create a new platform directory
- Create a new project directory
- Create a new command prompt build window
- Build the Windows CE operating system image for your new project

This tutorial assumes that you have installed the Embedded Development Kit in the default directory established by the setup program.

9.3.4.1 Create a New Platform Directory

To create a new platform directory, copy the Wince211\Platform\Cepc directory and all of its contents, including the subdirectories and their contents, into a new platform directory. The platform name should be no more than eight characters in length.

From a Command Prompt window, type the following commands to copy the \Cepc directory from its default location and create \MyPlat, your new platform directory:

```
cd \Wince211\Platform

xcopy Cepc MyPlat /E /V /I
```

9.3.4.2 Create a New Project Directory

The Windows CE Embedded Toolkit for Visual C++ 5.0 (Embedded Toolkit) contains typical configurations of the Windows CE operating system. Use the Maxall configuration as the template for your MyProj build.

1. From a Command Prompt window, type the following commands to copy the \Maxall directory (including all subdirectories and contents) into a new project directory named \MyProj. The project name should be no more than eight characters in length.

   ```
   cd \Wince211\Public

   xcopy Maxall MyProj /E /V /I
   ```

2. Rename the \MyProj\Maxall.bat file. The name of this batch file should always match the name of your project directory. Type the following commands to rename the project batch file:

   ```
   cd \Wince211\Public\MyProj

   rename MAXALL.BAT MYPROJ.BAT
   ```

9.3.4.3 Create a New Command Prompt Build Window

The Embedded Toolkit includes a collection of shortcuts for the Minshell configuration. Each shortcut opens a command prompt build window specific to a supported processor. Use the x86 Minshell command prompt build window shortcut as a template for your shortcut. Your project and hardware development platform directories must first be created before you launch this shortcut.

1. In the Windows CE Embedded Development Kit program group, select the x86 Minshell shortcut.
2. Make a copy of the shortcut and rename it “x86 MyProj”.
3. In the tab shortcut for the icon's properties, modify the target command line parameters that follow the Wince.bat command. These modifications are case sensitive.

   ```
   Wince.bat <cputype> <cpu> <os> <project> <platform>
   ```

   For your build, the last part of the command line (starting with Wince.bat) must read as follows:

   ```
   Wince.bat x86 I486 CE MYPROJ MYPLAT
   ```

4. Click OK to save the modified shortcut.

9.3.4.4 Build the Windows CE Operating System Image for a New Project

1. Invoke the x86 Build MyProj shortcut that you have previously created.
2. Enter the following command to build a complete version of Windows CE:

   ```
   blddemo
   ```

   After Blddemo.bat has finished processing, the current directory should still be the \Wince211 directory.
3. Be sure the build completed successfully by verifying that the Windows CE operating system image, called Nk.bin, exists in the \Wince211\Release directory. The build process sets this directory using the environment variable %_FLATRELEASEDIR%.

   If Nk.bin does not exist, then the build of the Windows CE operating system image was not completed successfully. Before rebuilding, refer to “Preparing Your Development Workstation for Subsequent Projects” in the Windows CE Embedded Toolkit.

   If Nk.bin exists, the Windows CE operating system image for MyProj was built successfully, and you can now run this image on your PC-based hardware development platform.

9.3.4.5 Adding Files or Applications to the Windows CE Operating System Image

1. Copy the file or application to the \Wince211\Public\<projectname>\Oak\Files directory.
2. Edit the project.bib file in the same directory. Add a reference to the end of the file to include the files/app formatted:

   ```
   filename.ext  $(_FLATRELEASEDIR)\filename.ext    S
   ```

3. Build the image by calling the blddemo utility.

Note: The filename.ext should be replaced with the name of your file or application. The *S* signals the image builder to make the file a system file. Other switches are H (hidden) and U (uncompressed), and can be used in combinations (e.g., SH = hidden system file).

9.4 Windows CE Device Drivers

Device drivers are programs that provide an interface between the operating system software and platform hardware. In previous versions of Microsoft operating systems, these device drivers were implemented as special system executable files that run at the same privilege level as the operating system kernel. In Windows CE, device drivers are implemented as DLLs and run at the same privilege level as the user code.

9.4.1 Design Note

Some third party vendors such as System Design Group (www.systemdesign.com) and Annasoft systems (www.annasoft.com) have additional Windows CE device drivers for bootstrap loading, flash memory, the 69000 VGA controller, and the audio chips. Some of these are also listed at the Driver Repository for Windows CE Platforms website listed in Appendix A.

9.4.2 Dynamic Link Libraries (DLLs)

DLLs are software function libraries like their predecessor, the static library. When a static library is linked to a program, the library functions used by the program are copied into the final executable image. This increases the size of the program. When a program is linked with a DLL, the library functions used by the program are *not* copied into the final executable image; only a function “stub” is copied. This function stub serves to identify the DLL and the function within the DLL that is being called upon. The actual DLL code is loaded into a separate memory space and the link between the program and the library function is created by the operating system when the function call is made. In addition, multiple programs can use DLLs simultaneously, reducing the memory requirements of each program.

9.4.3 Windows CE Device Driver Model

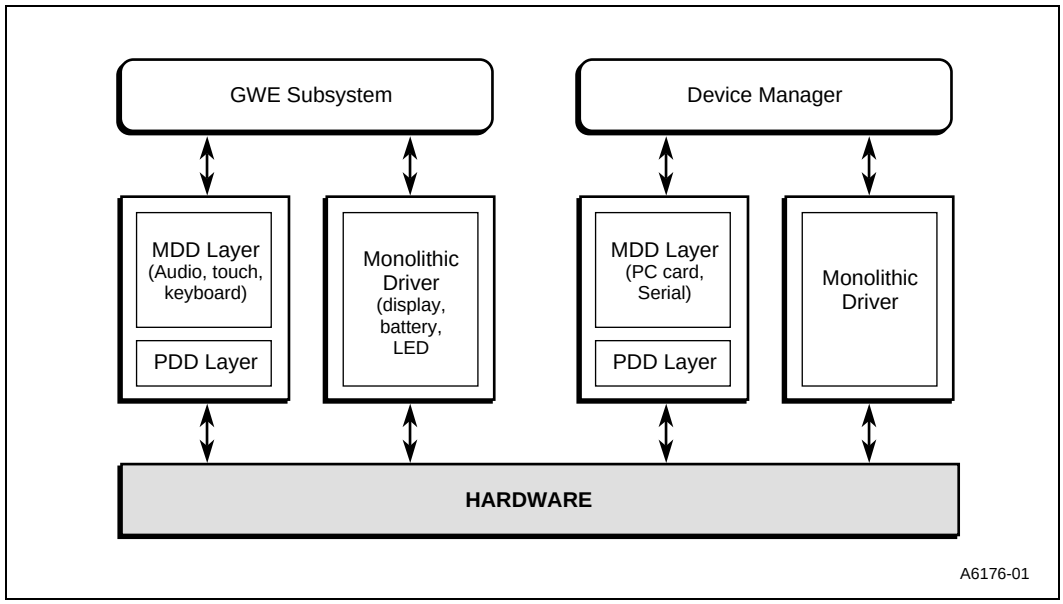
Windows CE has two types of device drivers as viewed from a software perspective: built-in and installable. Built-in drivers are typically for hardware that is native to the Windows CE target platform and are linked directly with the operating system image at build time. Some examples of this include keyboard, touch screen panel, battery, notification LED, PC card adapter, and audio hardware. Installable drivers are typically used for hardware that can be added to the platform. This type of driver is loaded by the operating system and exists as a stand-alone DLL. Some examples of this include modems, printers, digital cameras and PCMCIA cards. In either case, the terms *built-in* and *installable* relate to whether or not a particular device driver is linked directly with the operating system image or loaded at run time as a stand-alone DLL.

For Windows CE drivers, Microsoft provides a layer of source code known as the Model Device Driver (MDD) component. The MDD component defines the interface to the operating system for a given type of device driver and is common to all platforms that use that type of device. The MDD also defines the interface to the platform-specific driver code. This platform-specific driver code is referred to as the Platform Device Driver (PDD) component. The PDD component implements the interface to the hardware device. These two layers are compiled, then linked together to form the complete driver DLL.

Whether or not a given type of device driver is built-in or installable is pre-determined by the MDD/OS interface defined by Microsoft for that type of device. For built-in drivers, Microsoft has defined a set of *custom* MDD/OS interfaces. For installable drivers, Microsoft has defined a set of *common* MDD/OS interfaces.

Note: It is not required to implement Windows CE device drivers as MDD and PDD components. They may be implemented as a single, monolithic entity providing that the MDD/OS interface, as defined by Microsoft for a given device, is used. See Figure 6.

Figure 6. Device Driver Architecture



9.4.4 Interaction with Application Software

Installable device drivers are viewed as a special file in the file system by the application software. The interface exposed to the application corresponds to the standard Win32 file I/O functions such as OpenFile, ReadFile, DeviceIOControl, and others. The file system recognizes filenames as being special devices when they consist of exactly three uppercase letters, a single digit, and a colon (:). An example of this would be “COM1:”, which references the first serial port available on the platform.

Built-in device drivers are accessed by application software using API calls that Microsoft has implemented in the Windows CE operating system for each device.

9.4.5 Incorporating Device Drivers

Use Platform Builder’s GUI interface or CE toolkit to manually include the new drivers in the project directory and add them to the Windows CE image. In CE toolkit, first take the compiled driver or file and place it in the following directory:

%_WINCEROOT%\public\maxall\files

Note: *Maxall* can be changed to the name of your project directory.

Edit the project.bib file in the same directory to include your file(s) and run blddemo.

A serial or network cable is needed to monitor debug messages and the appropriate communication's programs must be running on the CE host system. The image must be downloaded to test the target board.

Refer to Microsoft's web site for information on new releases.

9.4.6 ACTiSYS IR 2000B

To configure the ACTiSYS IR 2000B component:

1. Disable port 2 in the BIOS.
2. Set the environment variable to use nscfir.dll instead of irsir.dll.
3. Add the following to the Maxall.bat file or to the batch file that is used:

   ```
   set IMGNSCFIR=1
   ```

4. To set up the registry correctly, modify the platform.reg file.
5. Find the following path and verify the following keys:
   [HKEY_LOCAL_MACHINE\Comm\NscIrda1\Parms]
   "BoardType"=dword:0
   "DongleType"=dword:4
   "ConfigBase"=dword:398

Note: The above keys are important keys; however other keys should also be checked for compatibility with your system (e.g., Bus Number and IRQ).

9.4.7 Audio

The card used in the sample design is the Creative Ensoniq* AudioPCI card. This configuration is valid for equivalent parts on the board, such as the Ensoniq ES1371 and an AC '97 codec. The following steps are required to configure the digital controller.

1. Include these driver files in the build:
   wavepdd.cpp
   wavepdd.h
   ensoniq.h
2. Edit the following source files:
   a. Makefile
   b. Make a directory in %_WINCEROOT%\platform\cepc\drivers for the driver.
      1. Name the directory. (In our design it is called it "ES1371WaveDev.")
      2. Copy files into directory.
      3. Edit file %_WINCEROOT%\platform\cepc\drivers\dirs to include this directory.
      4. Remove references to the WaveDev in "dirs" file
   c. Run "blddemo" from %_WINCEROOT%.

3. Set up configurations in VC++ to compile for Intel Architecture:
   a. Load Application.
      Check the Configuration list dialogue box for the desired configuration.
   b. Under “Build” menu, select “Configurations.”
   c. Click on the Add button.
   d. Choose a matching configuration from “Copy Settings from”:
      - For debug, select “Win32 (WCE xxxx) Debug.”
      - For release, select “Win32 (WCE xxxx) Release.”
        (xxxx signifies don't care)
   e. Choose a platform:
      - Choose WCE x86 to compile for target.
      - Choose WCE x86em to compile for emulator.
   f. Click on OK.
   g. Click on Close.

Appendix A Related Resources

Table 11. Related Intel Documents

Document	Order Number	URL
Celeron™ and Pentium® II Processor		
Celeron™ Processor at 266 MHz, 300 MHz, 300A MHz, 333 MHz, 366 MHz and 400 MHz datasheet	243658	http://www.intel.com/design/celeron/datashts/243658
Intel® Celeron™ Processor Specification Update	243748	http://www.intel.com/design/celeron/specupdt/243748.htm
370-Pin Socket (PGA370) Design Guidelines	244410	http://developer.intel.com/design/celeron/applnots/244410.htm
Pentium® II Processor AGTL+ Guidelines	243330	http://developer.intel.com/design/PentiumII/applnots/243330.htm
Pentium® II Processor Developer's Manual	243502	http://www.linear.com/aboutltc/functions/microprocessor.html
Pentium® II Processor Thermal Design Guidelines	243331	http://developer.intel.com/design/intarch/applnots/273331.htm
Pentium® II Processor Power Distribution Guidelines	243332	http://developer.intel.com/design/PentiumII/applnots/243332.htm
Pentium® II Processor Design for EMI	243334	http://developer.intel.com/design/PentiumII/applnots/243334.htm
VRM 8.2 DC-DC Converter Design Guidelines	243773	http://developer.intel.com/design/pentiumII/xeon/designgd/243773.htm
Write Combining Memory Guidelines	244422	http://developer.intel.com/design/pentiumII/applnots/244422.htm
P6 Family of Processors - Hardware Developer's Manual	244001	http://developer.intel.com/design/pentiumII/manuals/244001.htm
Intel® 440BX AGPset		
Intel® 440BX AGPset Desktop Design Guide	290634	http://developer.intel.com/design/chipsets/designex/290634.htm
Intel® 440BX AGPset Design Guide Update	290634	http://developer.intel.com/design/chipsets/designex/29064104.htm
Intel® 440BX AGPset: 82443BX Host Bridge/Controller Datasheet	290633	http://developer.intel.com/design/chipsets/datashts/290633.htm
Intel 440BX AGPset: 82443BX Host Bridge Controller Specification Update	290639	http://developer.intel.com/design/chipsets/specupdt/290639.htm
Intel® 82371AB PCI-to-ISA/IDE Xcelerator (PIIX4) datasheet	290562	http://developer.intel.com/design/chipsets/datashts/290562.htm
Intel® 82371EB (PIIX4) Specification Update	290635	http://developer.intel.com/design/chipsets/specupdt/290635.htm

Table 12. Related Specifications, Technical Papers, and Reference Books

Document	Location/URL
PCI Local Bus Specification, Revision 2.1	http://www.pcisig.com/specs.html
Accelerated Graphics Port Interface Specification 2.0	http://www.intel.com/pc-supp/platform/agfxport/index.htm
Pentium Pro and Pentium II System Architecture	http://www.mindshare.com
PCI System Architecture	http://www.mindshare.com
AGP System Architecture	http://www.mindshare.com
SEMTECH SC1185 DC/DC Converter Datasheet	http://www.semtech.com
MAXIM MAX1617 Remote Temperature Sensor	http://www.maxim-ic.com
Cypress CY2280 Clock Synthesizer/Driver and CY2318 SDRAM Buffer Datasheets	http://www.cypress.com
Standard Microsystems Corporation - Ultra I/O and Super I/O Controller Datasheets	http://www.smsc.com/main/document.html
Texas Instruments PCI1225 PC Card Controller	http://www.ti.com/sc/docs/products/analog/pci1225.html
Creative Labs ES1373 AC97 Digital Controller Datasheet	http://www.creative.com
Analog Devices AD1819 AC97 SoundPort Codec	http://products.analog.com/products/info.asp?product=AD1819
69000 HiQVideo Accelerator with Integrated Memory datasheet	http://www.chips.com/design/graphics/mobilegraphics/datashts/ds_69000.htm
Application Notes for the HiQVideo Family	http://www.chips.com/design/graphics/mobilegraphics/applnots/
Designing with Flash Memory in Windows CE Applications	http://developer.intel.com/design/flash/isf/wince/wince1.htm
Understanding Modularity in Microsoft Windows CE	http://www.microsoft.com/windowsce/embedded/resources/techpap.asp
Introducing the Windows CE Embedded Toolkit	http://www.microsoft.com/windowsce/embedded/resources/techpap.asp
Driver Repository for Windows CE Platforms	http://www.microsoft.com/windowsce/embedded/resources/driverrep.asp
Windows CE Driver for Intel 82559 Ethernet Controller	http://developer.intel.com/platforms/applied/software/559ce211.htm
Windows CE Driver for 69000 VGA Controller	http://www.annasoft.com/vdk69000.htm
Windows CE Platform Builder	http://www.microsoft.com/windowsce/Embedded/resources/pb.asp
Windows CE Platform Builder 2.12 Getting Started Guide	http://msdn.microsoft.com/library/techart/guide.htm
Windows CE Developer Documentation	http://www.microsoft.com/windowsce/Embedded/download/212doc.asp
Windows CE Developer's Handbook	http://www.sybex.com
Windows CE PC Hardware Reference Platform	http://www.microsoft.com/windowsce/Embedded/resources/refplat/cepcconfig.asp

Appendix B Schematics

B.1 Transaction Terminal Baseboard Schematics

Schematics are provided for the following items:

- Embedded Celeron Processor
- 440BX part 1 and 2
- SDRAM DIMM sockets 0 and 1
- ISA/PCI pullups
- Clocks and buffers
- 82559 Ethernet Controller
- PCI slots 0 and 1
- Reset and Powergood
- ATX power connector
- DC/DC Converter
- 69000 VGA Video part 1 and 2
- AC ’97 Audio part 1 and 2
- PCI1225 PCMCIA Controller
- StrataFlash
- PIIX4E part 1 and 2
- IDE connector
- USB connectors
- Ultra I/O
- ISA connector
- COMx, DB25, floppy
- Touch screen controller
- Super I/O (extra serial ports)
- Flash BIOS
- Unused gates

Celeron™ Processor Based Transction Terminal Sample Design

Table of Contents

THIS SCHEMATIC IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANT OTHERWISE ARISING OUT OF PROPOSAL, SPECIFICATION OR SAMPLE.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

Intel disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification. Intel does not warrant or represent that such use will not infringe such rights.

Copyright Intel Corporation 1999

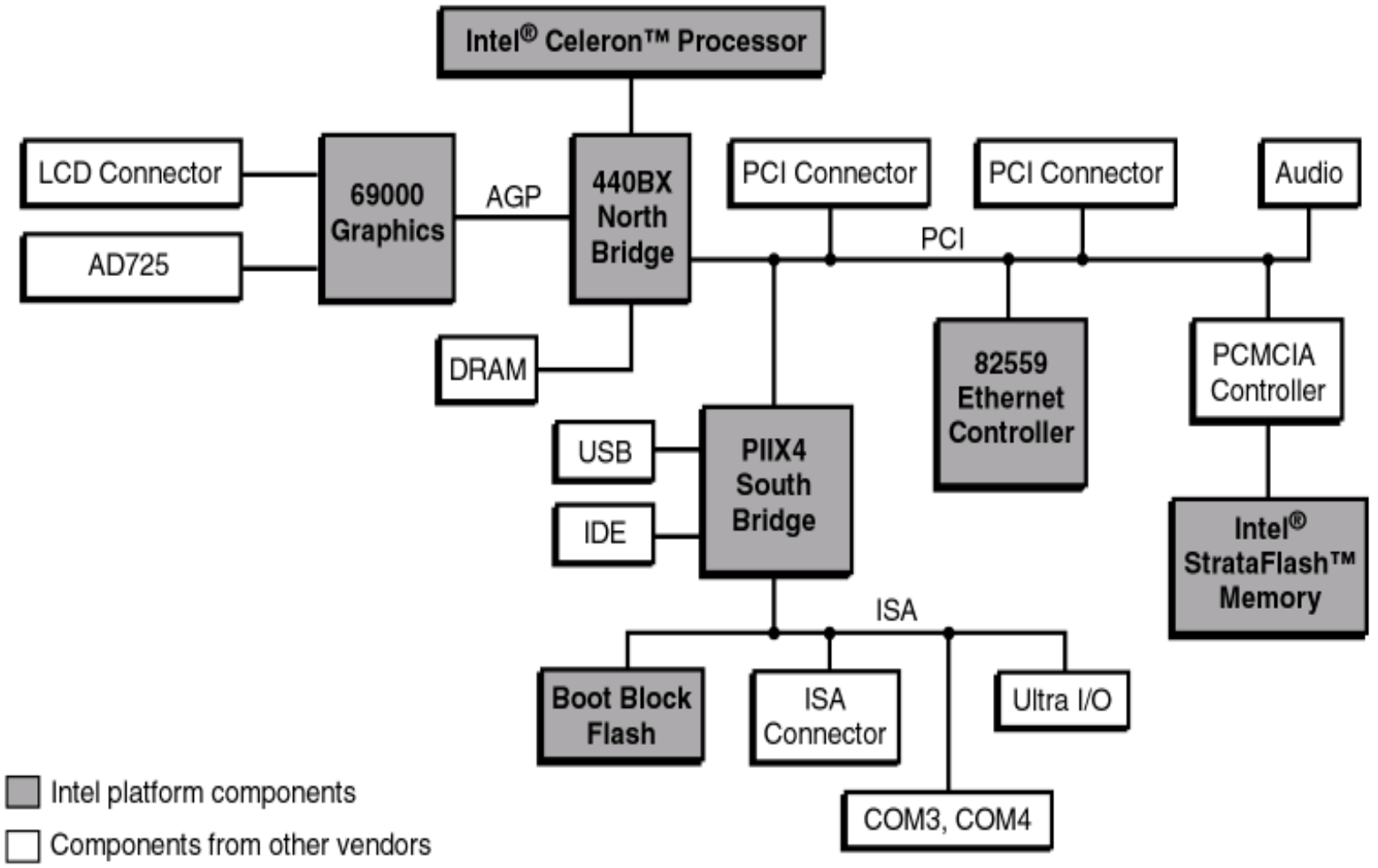
Revision History

8/15/1999	Revision A -- Initial release.

Intel Embedded Microprocessor Division

5000 W. Chandler Blvd.
Chandler AZ. 85226

Title: Title Page	Celeron™ Processor Based TT Sample Design	
Size: C	Document Number	Rev: A
Date: 6/16/1999	Sheet 1 of 34	



A7261-01

Intel Embedded Microprocessor Division
5000 W. Chandler Blvd.
Chandler AZ. 85226

Title: Celeron Processor Based TT Sample Design
TT Block Diagram

Size: C | Document Number | Rev: A

Date: 6/16/1999 | Sheet 2 of 34

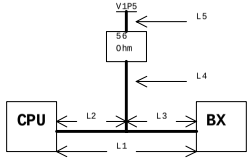
General Routing Requirements

>> PWB must be routed using eight (8) layers, with the following stackup:

Layer 1: Signal, V1P5 Bus, V2P5 Bus
Layer 2: GND
Layer 3: Power -- V3P3 Plane
Layer 4: Signal
Layer 5: Signal
Layer 6: Split Power Plane -- V5P0 Plane, V2P0 Island
Layer 7: GND
Layer 8: Signal

>> Adjacent signal layers must be orthogonal.
>> No right angles are to be used when routing, and serpentines must use curves.
>> Vias for decoupling capacitors must be kept as close as possible to the capacitor pad.
>> Trace impedance mst be 65 ohms, +/- 10%
>> Total board thickness must be .062".
>> Board material must be FR-4.
>> GND layers must not be split.
>> Top and bottom (outer) layers must be no less than 1/2 oz copper before plating, inner layers must be 1 oz copper.
>> Series terminating resistors must be kept as close to the driving pin as possible.
>> Daisy chain signals going to more that on point, do not use stubs.

GTL+ Bus Specific Routing Requirements



1.5" ≤ L1 ≤ 4.5"
0.0" ≤ L3 ≤ 1.0"
0.0" ≤ L4 ≤ 2.0"
0.0" ≤ L5 ≤ 0.1"

GTL+ GROUPS -- CPU PPGA370 Names

>> Address Signals: A#[31:3]
>> Data Signals: D#[63:0]
>> Control Signals:

BPRI#, DEFER#, RESET#, RS[2:0]#, TRDY#, PRDY#, ADS#, BNR#, BP[3:2]#, BPM[1:0]#, BR0#, DBSY#, DRDY#, HIT#, HITM#, LOCK#, REQ[4:0]#

>> Minimum trace width for L5 (V1P5) power bus bus is 150 mils.
>> Minimum trace width is 5 mils.
>> Minimum space between traces is 10 mils, this includes adjacent layers.
>> Minimum space between GTL+ traces and other types of traces is 25 mils, this includes adjacent layers.
>> GTL+ address, data, and control lines must be routed as separate groups and treated as different signal types.
>> GTL+ groups must be matched in length by 750 mils or less.

Memory Bus Specific Routing Requirements

>> Minimum trace width is 5 mils.
>> Minimum space between traces is 10 mils, this includes adjacent layers.
>> Minimum space between memory traces and other types of traces is 25 mils, this includes adjacent layers.
>> Memory address, data, and control lines must be routed as separate groups and treated as different signal types.
>> The MD, DQM, CS#, CKE, SRAS#, SCAS#, WE#, and MA signals (between BX and DIMMs) must be between 1" and 3" in length, matched to within 600 mils.
>> Groups must be matched in length by 750 mils or less.

Memory Bus GROUPS -- DIMM Names

>> Address Signals: MAA[13:0]
>> Data Signals: MD#[63:0]
>> Control Signals:

DClk, SMBDATA, SMBCLK, CSA[1:0]#, WE_A#, SCAS_A#, SRAS_A#

PCI Bus Specific Routing Requirements

>> The PIIX4E must be the last device on the PCI bus.
>> PCI bus max length must be less than 4".
>> Groups must be matched in length by 750 mils or less.

PCI Bus GROUPS

>> Address/Data Signals: AD[31:0]
>> Control Signals:

BUSMD1#, C/BE[3:0]#, REQ1#, GNT1#, INT[A:D]#, PCIRST#, FRAME#, TRDY#, IRDY#, STOP#, DEVSEL#, SERR#, PERR#, PAR, LOCK#

Clock Specific Routing Requirements

>> A clock trace must not alternate layers.
>> A clock trace must be separated by a minimum of 25 mils from any other trace, including serpentines, 11 mils is OK when going betwen pins or balls.
>> The CPU clock length must be between 1" and 9", and 460 mils shorter than the BX clock. The BX clock and all PCI clocks must be matched in length, except for the PMC clock which must be 2" shorter than all other PCI clocks.
>> SDRAM_CLK[1:0] must be matched in length, and between 1" and 3" in length.
>> CLKOUT, from the SDRAM clock driver to the BX, must be between 3.5" and 5.5" in length, and 2.5" longer than SDRAM_CLK[1:0].
>> Trace length between BX-DCLK0 and SDRAM clock driver (signal REF) must be between 1" and 6".

IDE Specific Routing Requirements

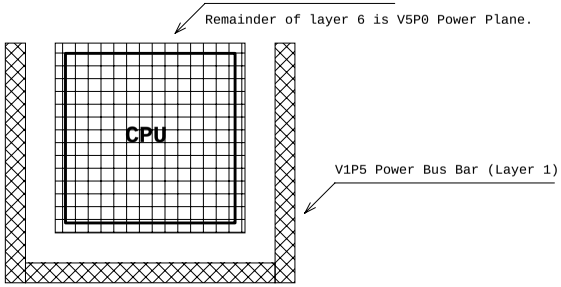
>> Place IDE conector within 3" of PIIX4E.
>> Place IDE series terminating resistors within 500 mils of PIIX4E.

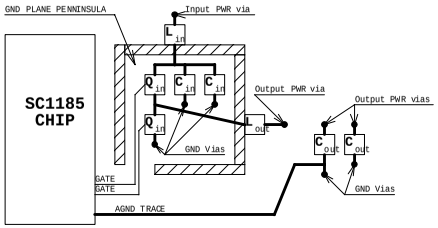
Power Supply Specific Routing Requirements

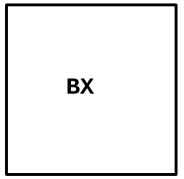
>> All unrelated signals and power planes must be kept away from the switching circuits.
>> All traces associated with the input power/ground connectors, and the capacitors connected to these connectors, must be routed with minimum length and maximum width.
>> See the Power Supply schematic sheet for complete restrictions.

General Placement Requirements

>> Place major components and connectors as shown in the Mechanical drawings.
>> Rotate the CPU and BX chips so as to minimize the length of the connections between the two, and place as shown in the diagram to the right:
>> The diagram below depicts the placement requirements for the SC1185 switching regulator circuits. A GND plane penninsula is used to contain the high current and noisy switching components. All of the bold traces and vias shown have high current and low impedance requirements. These trace widths should be 200 mils minimum. The channel which connects the GND penninsula to the main GND plane must be 500 mils in width.







>> No other signals may be routed underneath the GND plane penninsula.
>> No other power planes may exist underneath the GND plane penninsula.

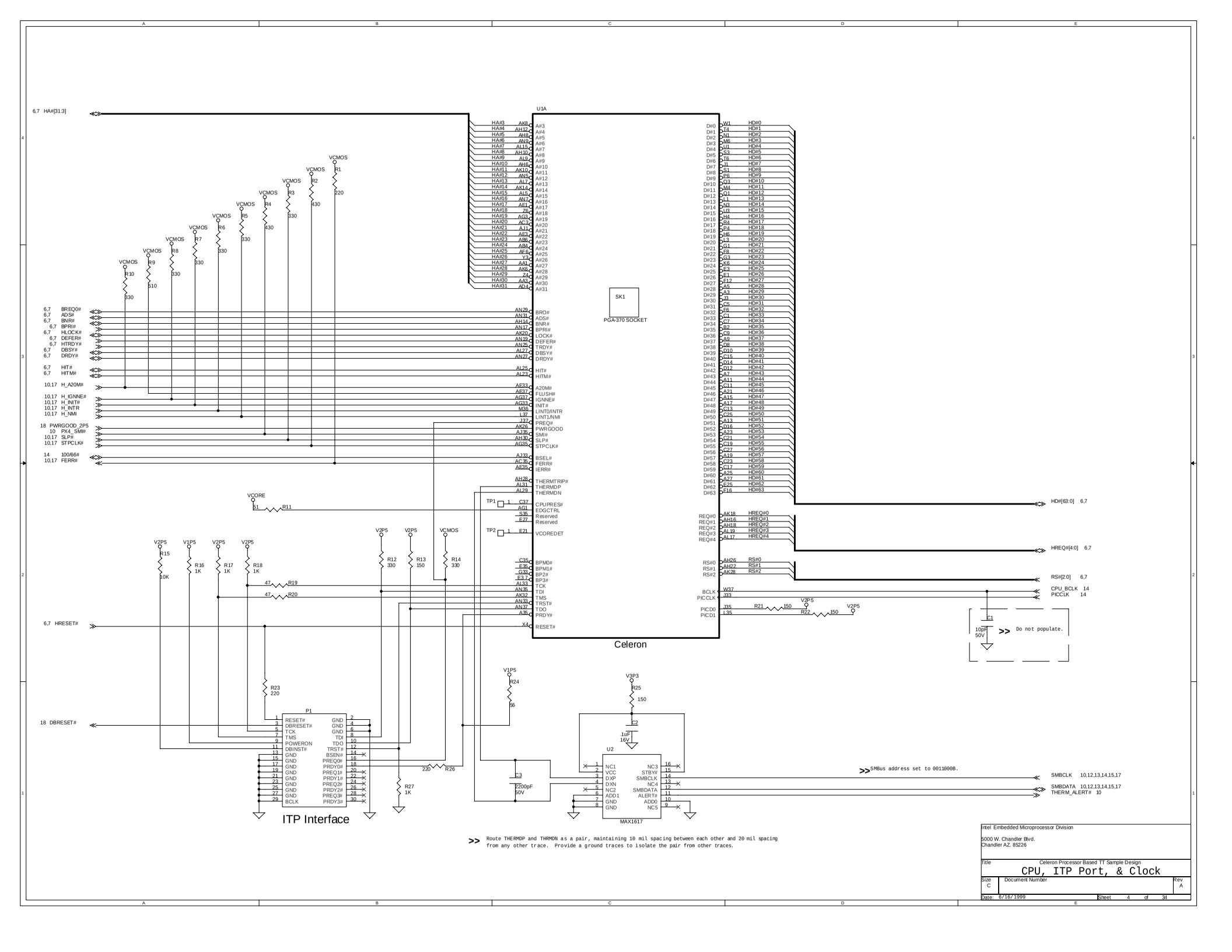
Intel Embedded Microprocessor Division

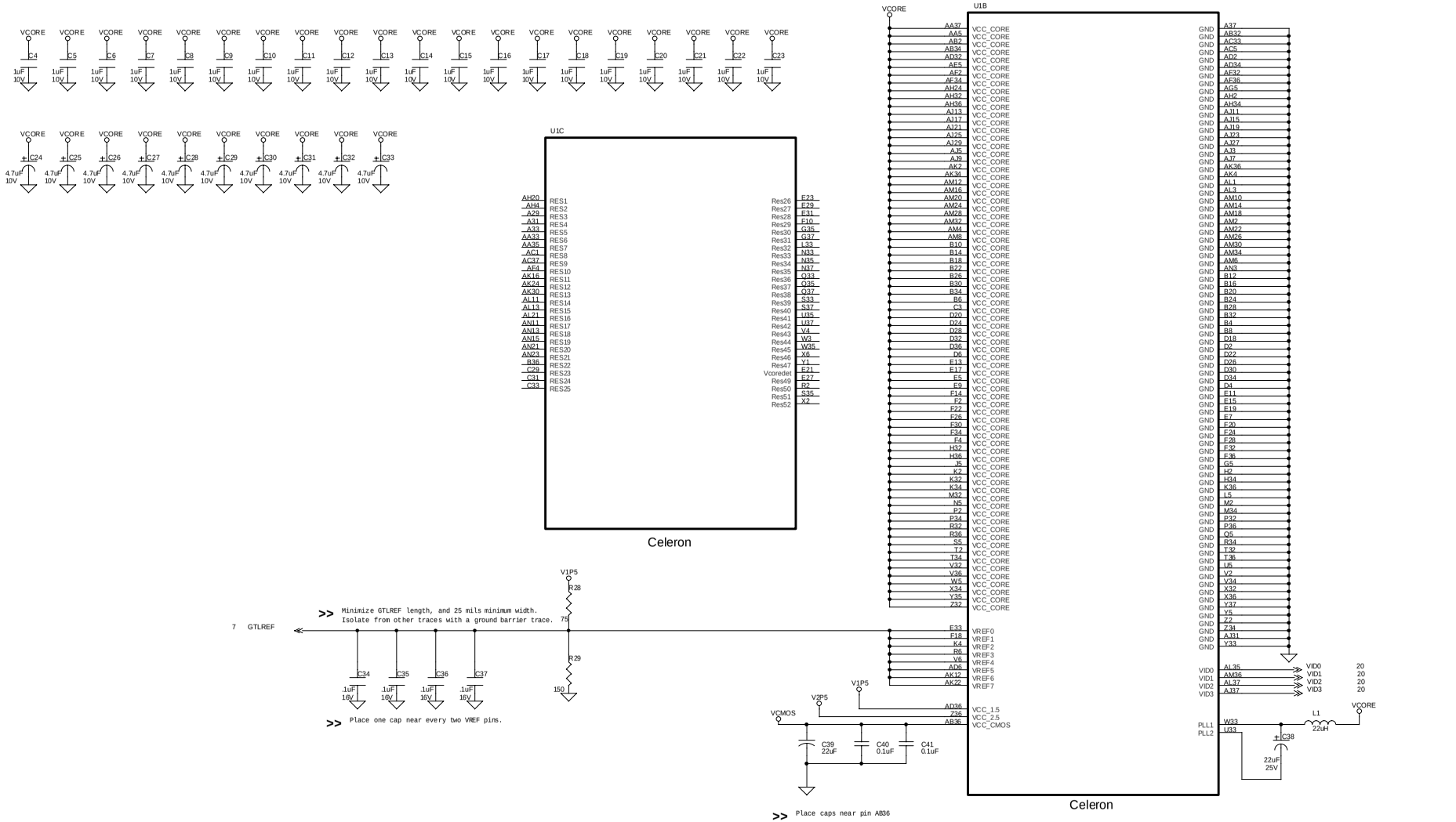
5000 W. Chandler Blvd.
Chandler AZ. 85226

Title: Celeron Processor Based TT Reference
Routing Guidelines/Restrictions

Size: C | Document Number | Rev: A

Date: 6/16/1999 | Sheet 3 of 34


U1A
SK1
PGA-370 SOCKET
Celeron
ITP Interface
P1
MAX1617
U2
Do not populate.
SMBus address set to 0011000B.
Route THERMDP and THRMDN as a pair, maintaining 10 mil spacing between each other and 20 mil spacing from any other trace. Provide a ground traces to isolate the pair from other traces.
Intel Embedded Microprocessor Division
5000 W. Chandler Blvd.
Chandler AZ. 85226
Title
Celeron Processor Based TT Sample Design
CPU, ITP Port, & Clock
Size C
Document Number
Rev A
Date: 6/16/1999
Sheet 4 of 34

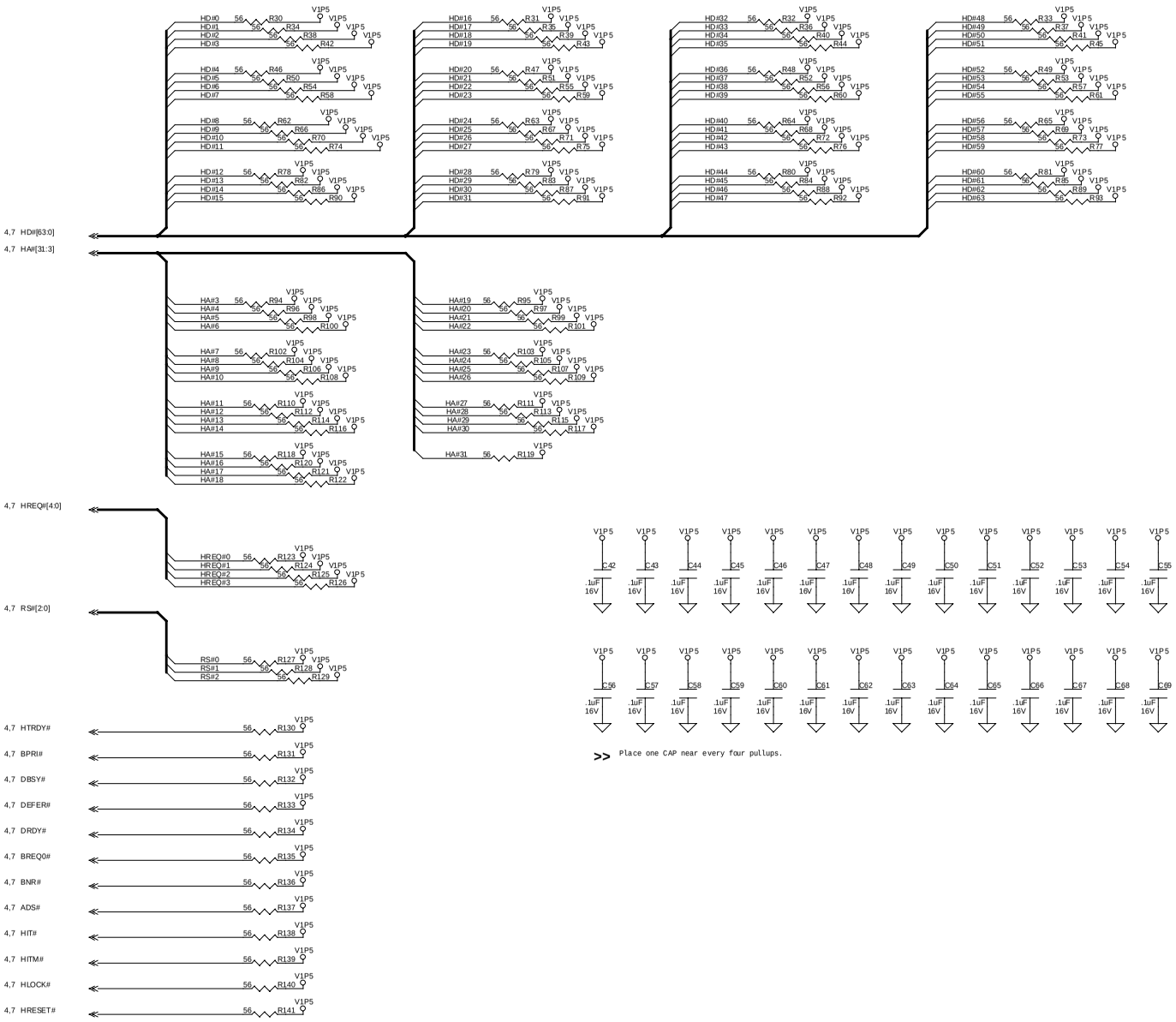


Intel Embedded Microprocessor Division

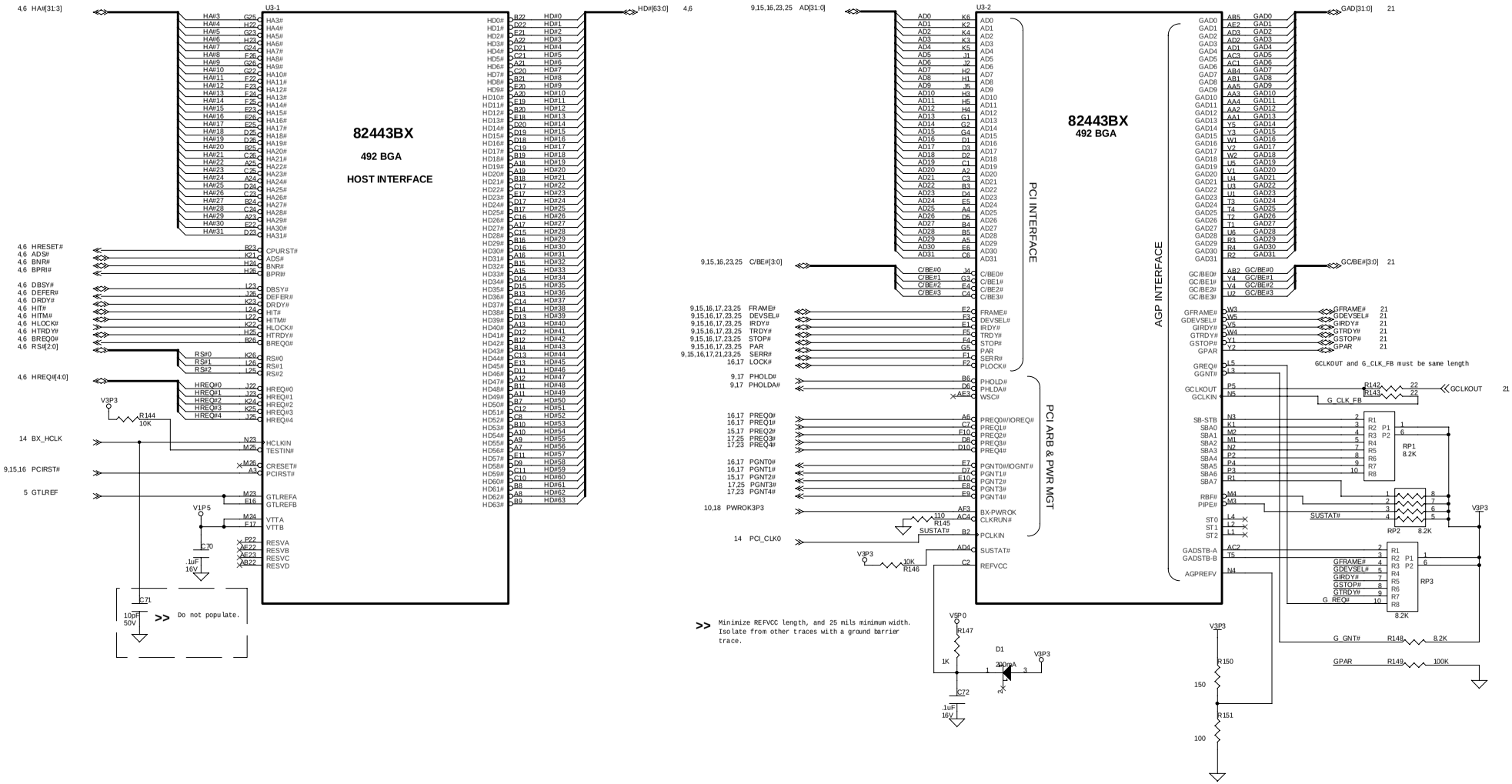
5000 W. Chandler Blvd.
Chandler AZ. 85226

Title	Celeron Processor Based TT Sample Design	
	CPU Power	
Size C	Document Number	Rev A
Date: 6/16/1999	Sheet 5 of 34	

GTL+ TERMINATION RESISTORS



Intel Embedded Microprocessor Division	
5000 W. Chandler Blvd. Chandler AZ. 85226	
Title	Celeron Processor Based TT Sample Design
GTL+ Termination	
Size C	Document Number — Rev A
Date: 6/16/1999	Sheet 6 of 34

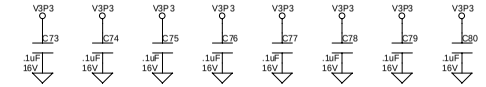


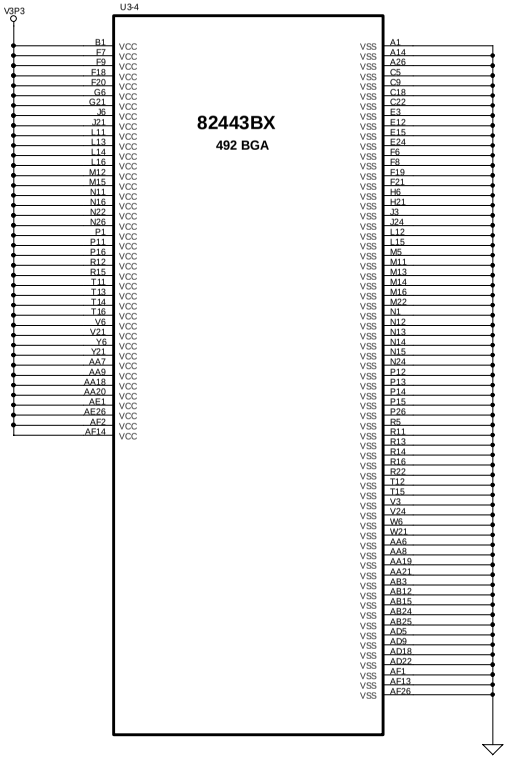
Intel Embedded Microprocessor Division

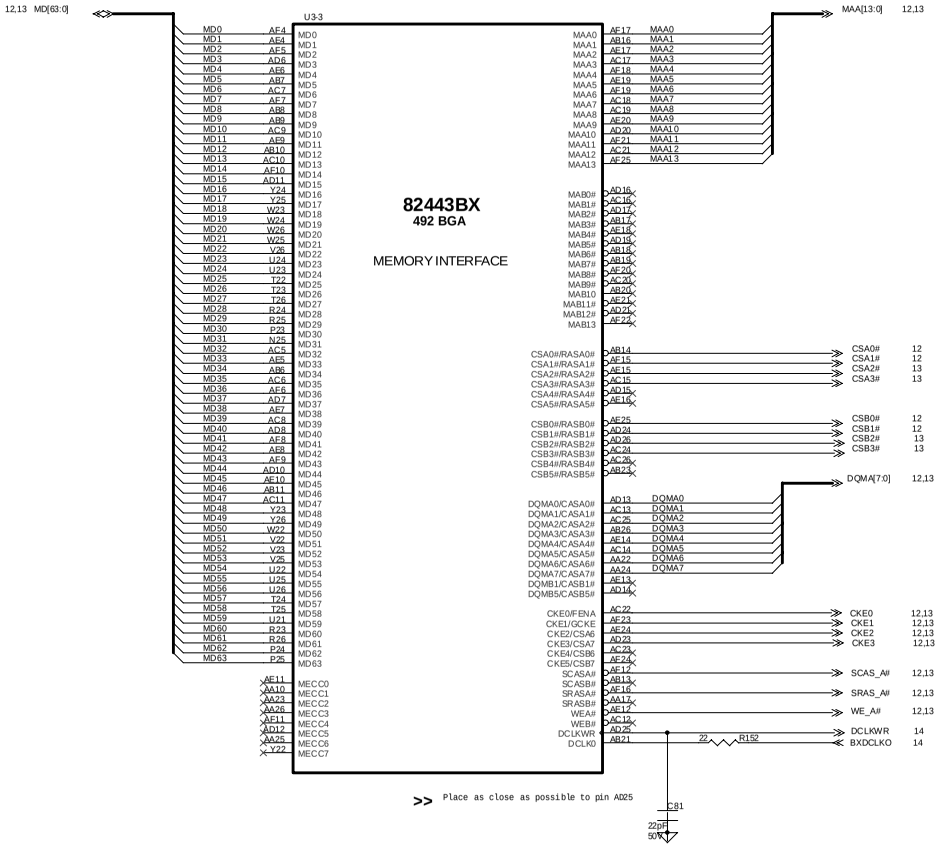
5000 W. Chandler Blvd.
Chandler AZ. 85226

Title	Celeron Processor Based TT Sample Design	
	440BX - Part One	
Size: C	Document Number	Rev: A
Date: 6/16/1999	Sheet 7 of 34	

>> Place decoupling caps around BX, at a minimum the four corners if routing prevents even distribution.



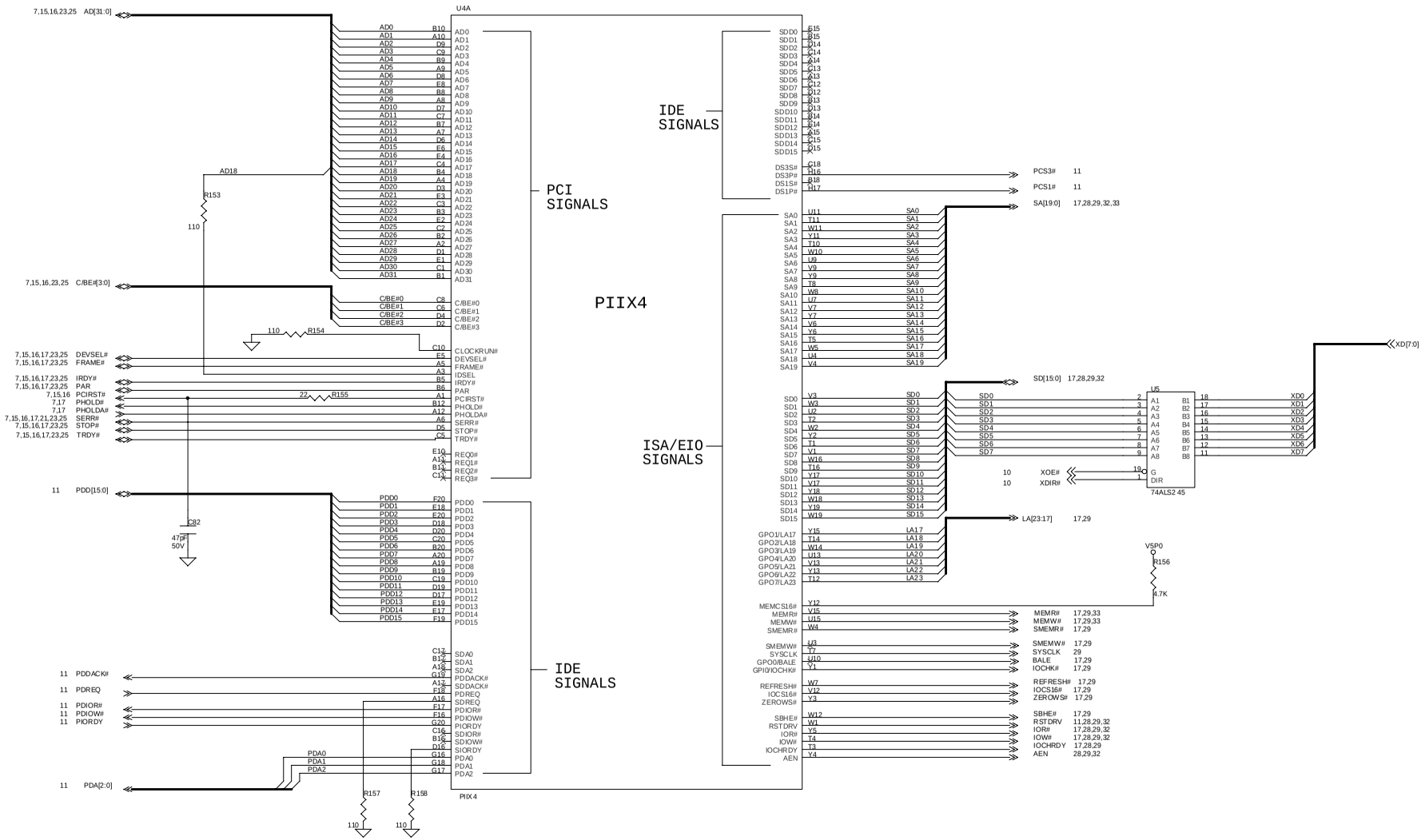




Intel Embedded Microprocessor Division

5000 W. Chandler Blvd.
Chandler AZ. 85226

Title	Celeron Processor Based TT Sample Design	
440BX Part Two & Clock		
Size C	Document Number	Rev A
Date: 6/16/1999	Sheet 8 of 34	



Intel Embedded Microprocessor Division

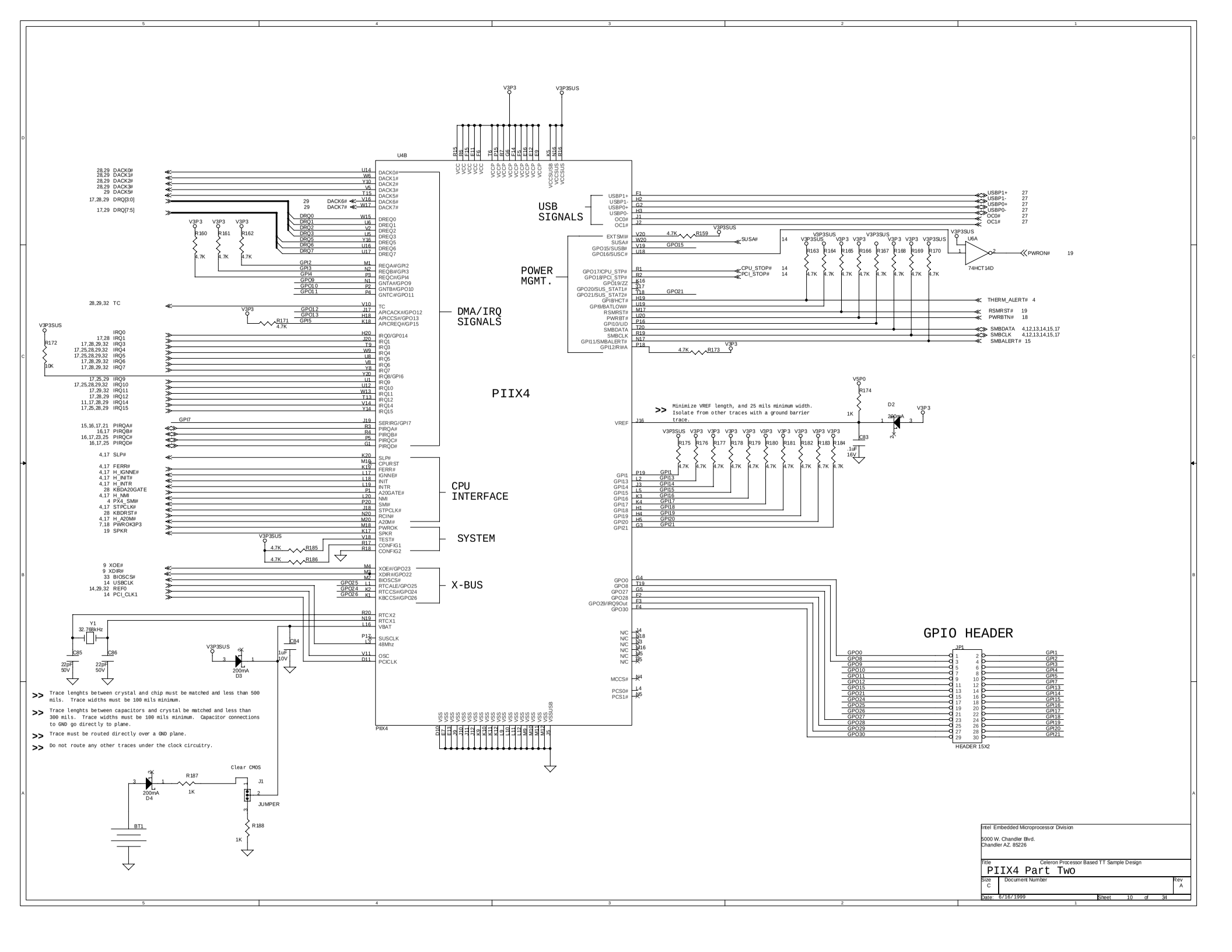
5000 W. Chandler Blvd.
Chandler AZ. 85226

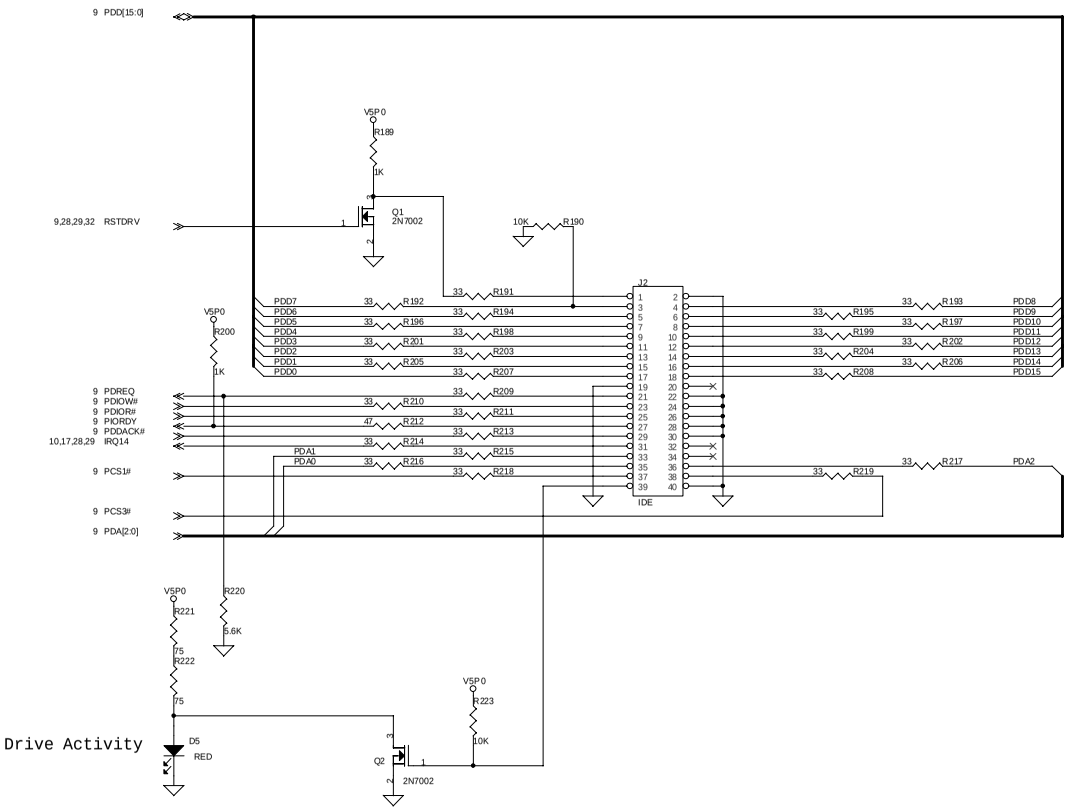
Title: Celeron Processor Based TT Sample Design

PIIX4 Part One

Size: C | Document Number | Rev: A

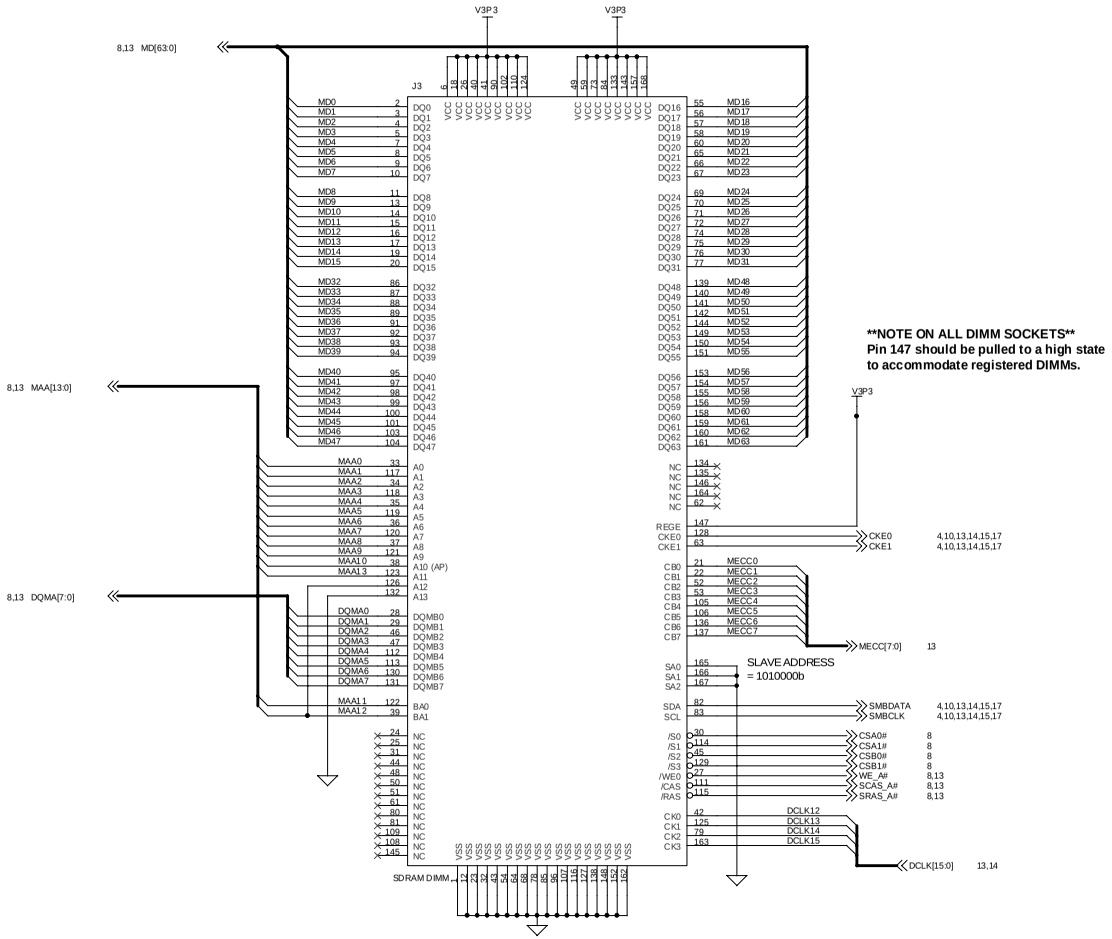
Date: 8/16/1999 | Sheet 9 of 34


U4B
PIIX4
DMA/IRQ SIGNALS
PIIX4
CPU INTERFACE
SYSTEM
X-BUS
USB SIGNALS
POWER MGMT.
GPIO HEADER
HEADER 15X2
JUMPER
Clear CMOS
>> Minimize VREF length, and 25 mils minimum width. Isolate from other traces with a ground barrier trace.
>> Trace lenghts between crystal and chip must be matched and less than 500 mils. Trace widths must be 100 mils minimum.
>> Trace lenghts between capacitors and crystal be matched and less than 300 mils. Trace widths must be 100 mils minimum. Capacitor connections to GND go directly to plane.
>> Trace must be routed directly over a GND plane.
>> Do not route any other traces under the clock circuitry.
Intel Embedded Microprocessor Division
5000 W. Chandler Blvd.
Chandler AZ. 85226
Title
Celeron Processor Based TT Sample Design
PIIX4 Part Two
Size C
Document Number
Rev A
Date: 6/16/1999
Sheet 10 of 34



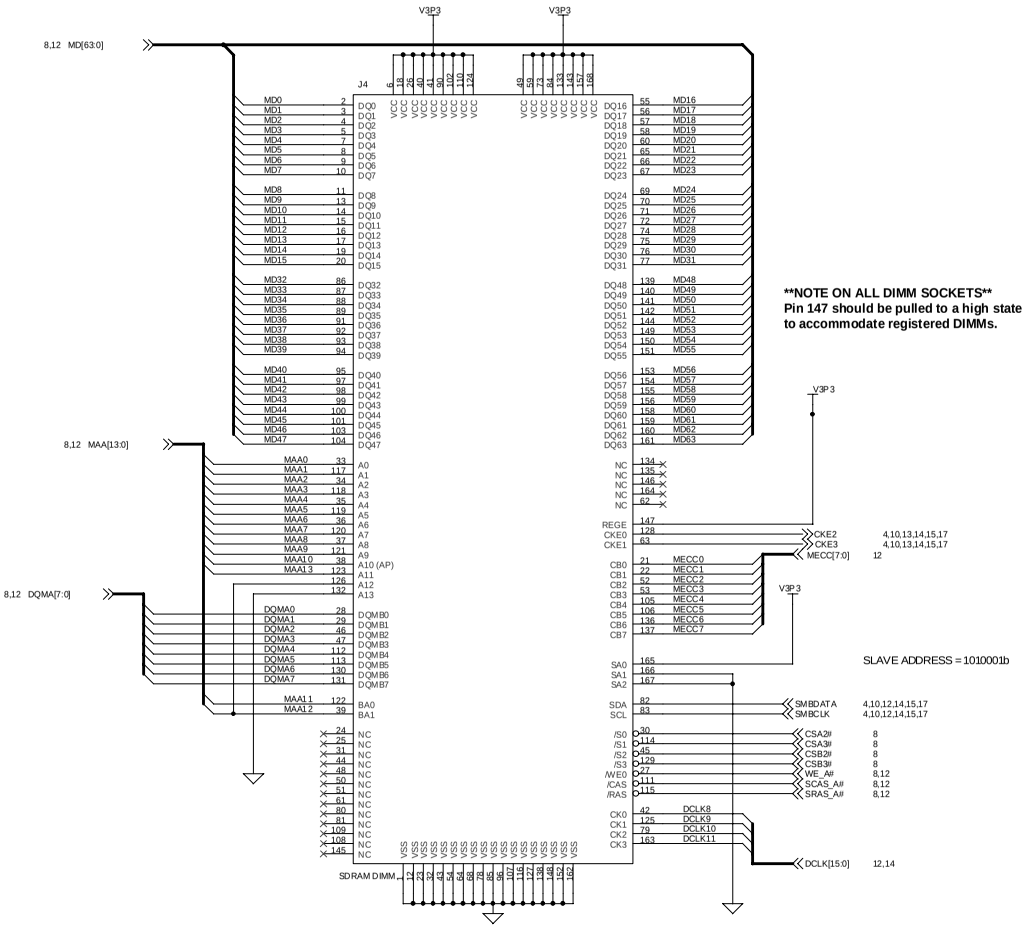
Intel Embedded Microprocessor Division			
5000 W. Chandler Blvd. Chandler AZ. 85226			
Title	Celeron Processor Based TT Sample Design		
IDE Interface			
Size C	Document Number		Rev A
Date: 6/16/1999	Sheet 11 of 34		

DIMM SOCKET 0

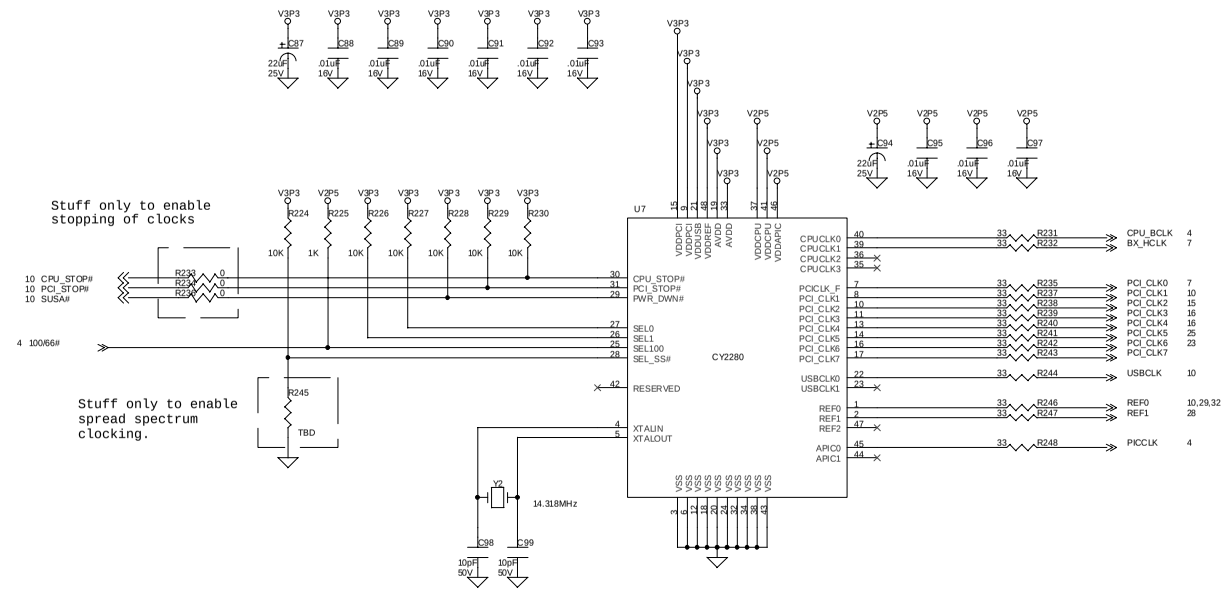


Intel Embedded Microprocessor Division	
5000 W. Chandler Blvd. Chandler AZ. 85226	
Title	Celeron Processor TT Sample Design Memory Socket 0
Size C	Document Number — Rev A
Date: Monday, August 09, 1999	Sheet 12 of 34

DIMM SOCKET 1



Intel Embedded Microprocessor Division	
5000 W. Chandler Blvd. Chandler AZ. 85226	
Title	Celeron Processor TT Sample Design
	Memory Socket 1
Size C	Document Number — Rev A
Date: Monday, August 09, 1999	Sheet 13 of 34

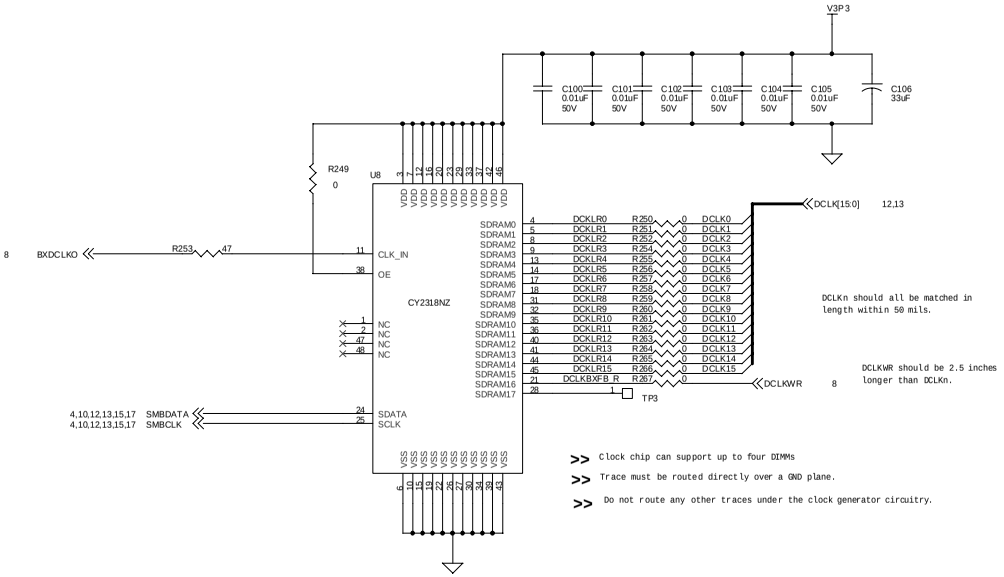


>> Trace lenghts between crystal and chip must be matched and less than 500 mils. Trace widths must be 100 mils minimum.

>> Trace lenghts between capacitors and crystal be matched and less than 300 mils. Trace widths must be 100 mils minimum. Capacitor connections to GND go directly to plane.

>> Trace must be routed directly over a GND plane.

>> Do not route any other traces under the clock generator circuitry.

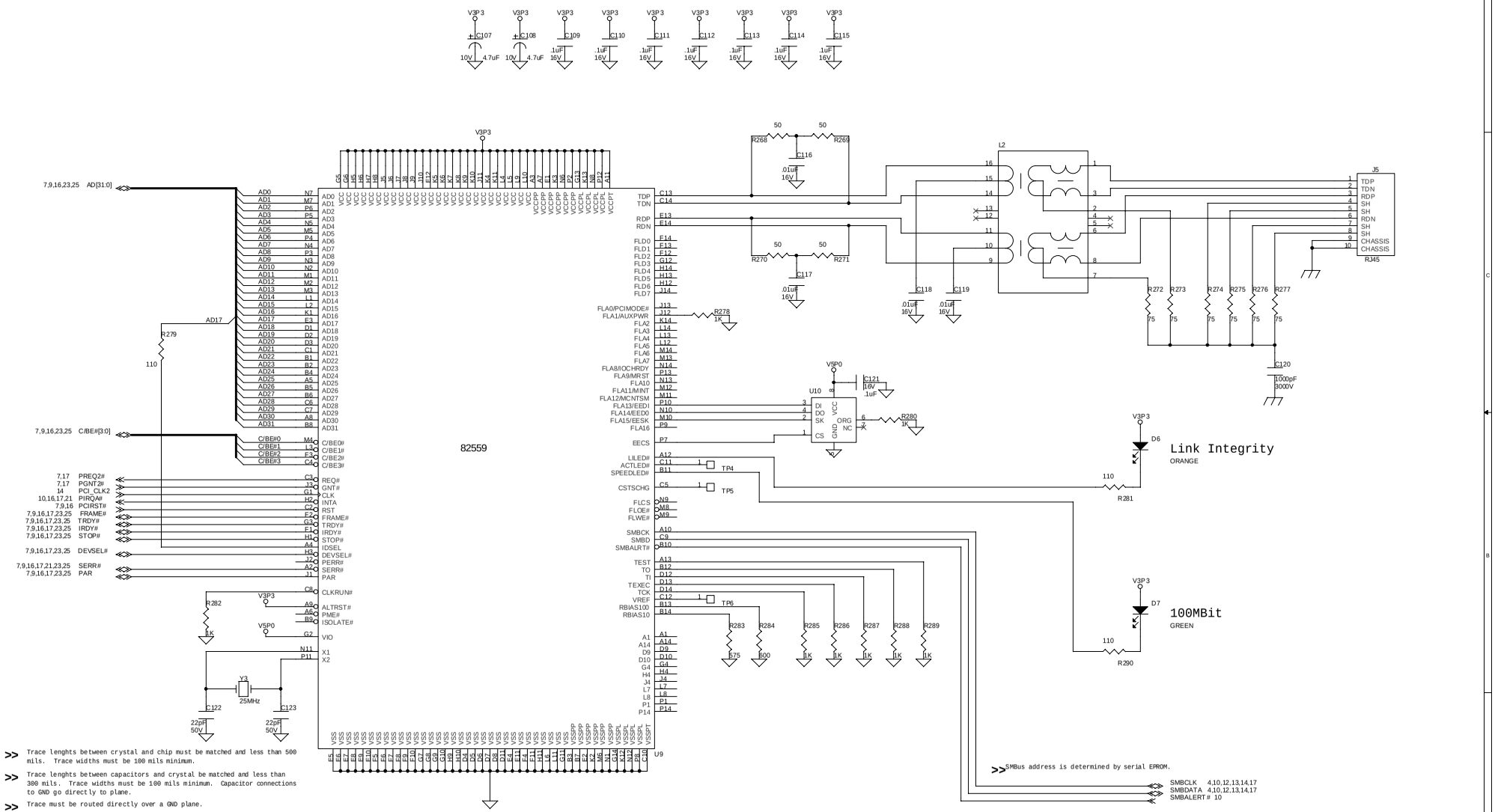


>> Clock chip can support up to four DIMMs

>> Trace must be routed directly over a GND plane.

>> Do not route any other traces under the clock generator circuitry.

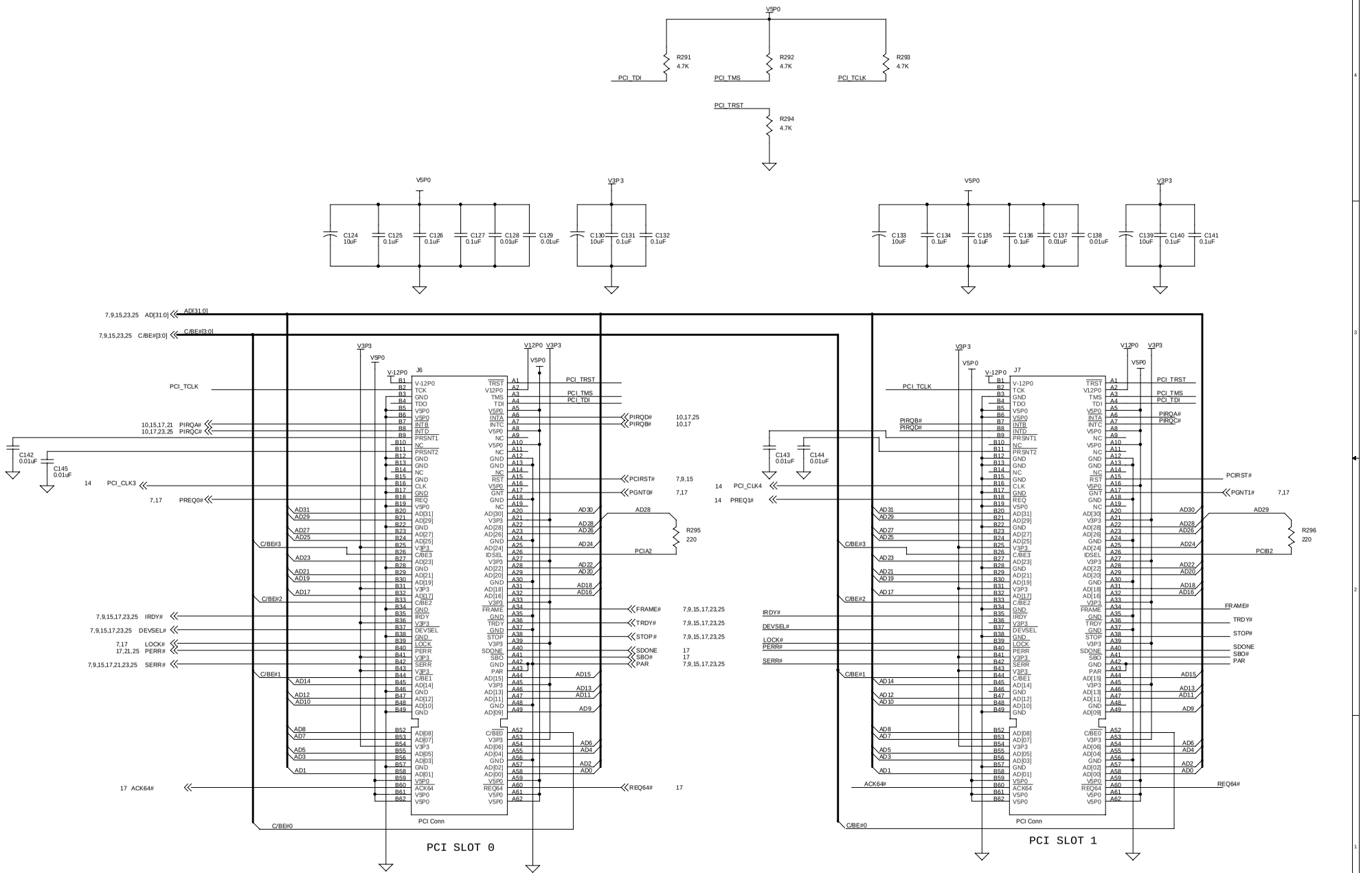
Intel Embedded Microprocessor Division	
5000 W. Chandler Blvd. Chandler AZ. 85226	
Title	Celeron Processor Based TT Sample Design
Clock & Clock Buffers	
Size C	Document Number — Rev A
Date: 6/16/1999	Sheet 14 of 34



Intel Embedded Microprocessor Division

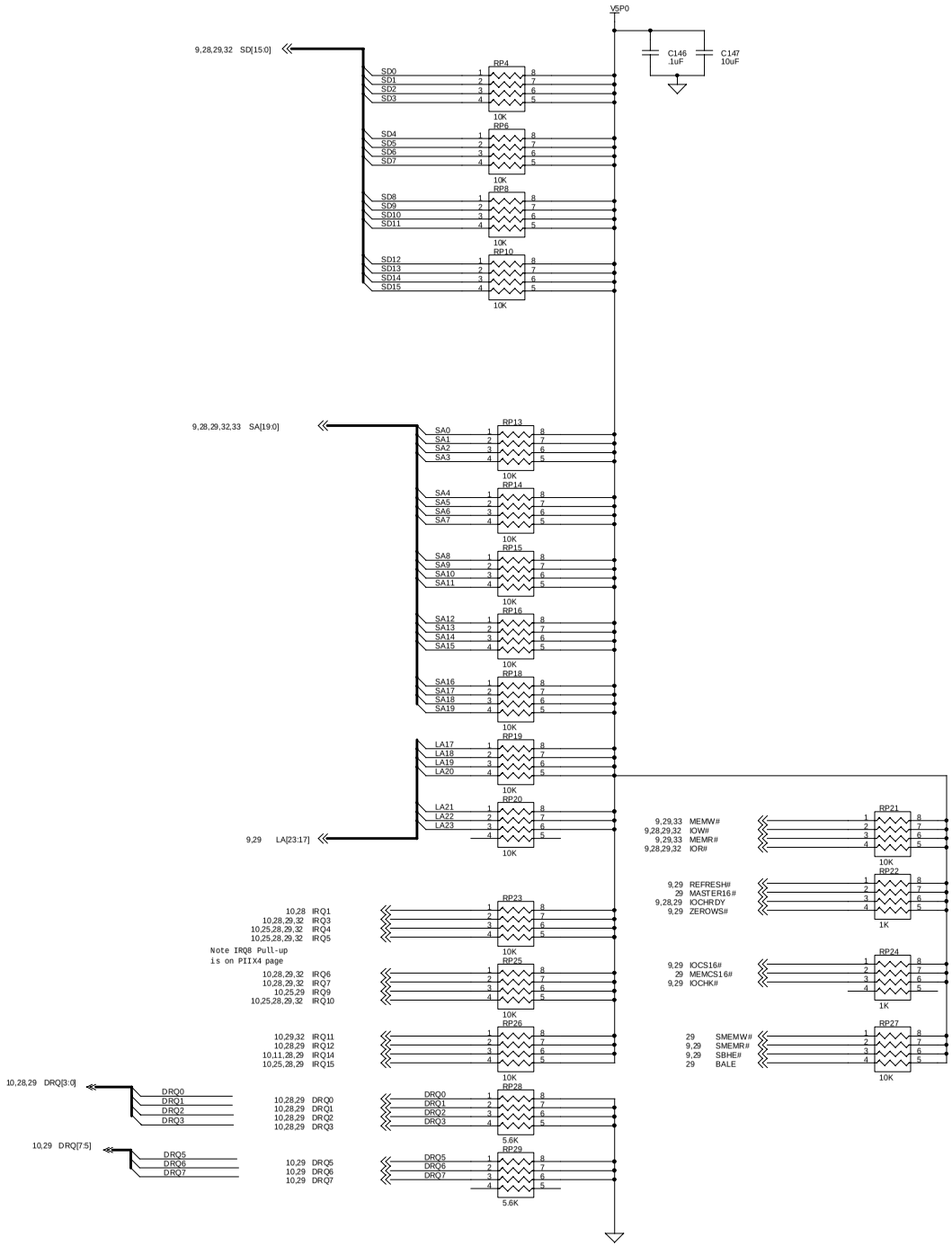
5000 W. Chandler Blvd.
Chandler AZ. 85226

Title	Celeron Processor Based TT Sample Design	
	Ethernet Port	
Size C	Document Number	Rev A
Date: 6/16/1999	Sheet 15 of 34	

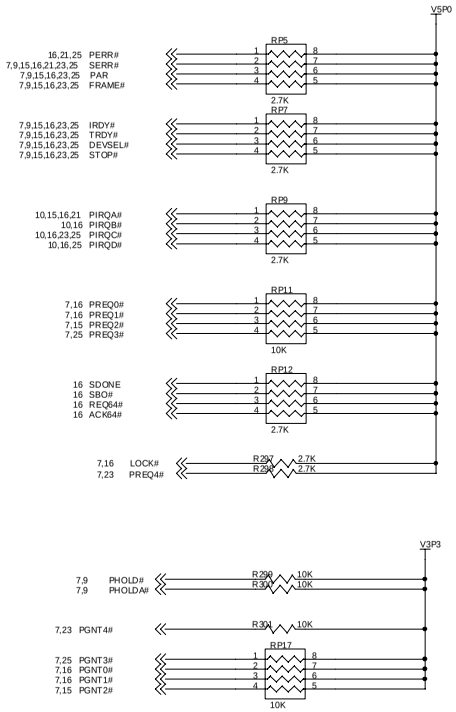


Intel Embedded Microprocessor Division	
5000 W. Chandler Blvd. Chandler AZ. 85226	
Title	Celeron Processor Based TT Sample Design
	PCI Connectors
Size C	Document Number
Rev A	
Date:	Sheet 16 of 34

ISA Pullups

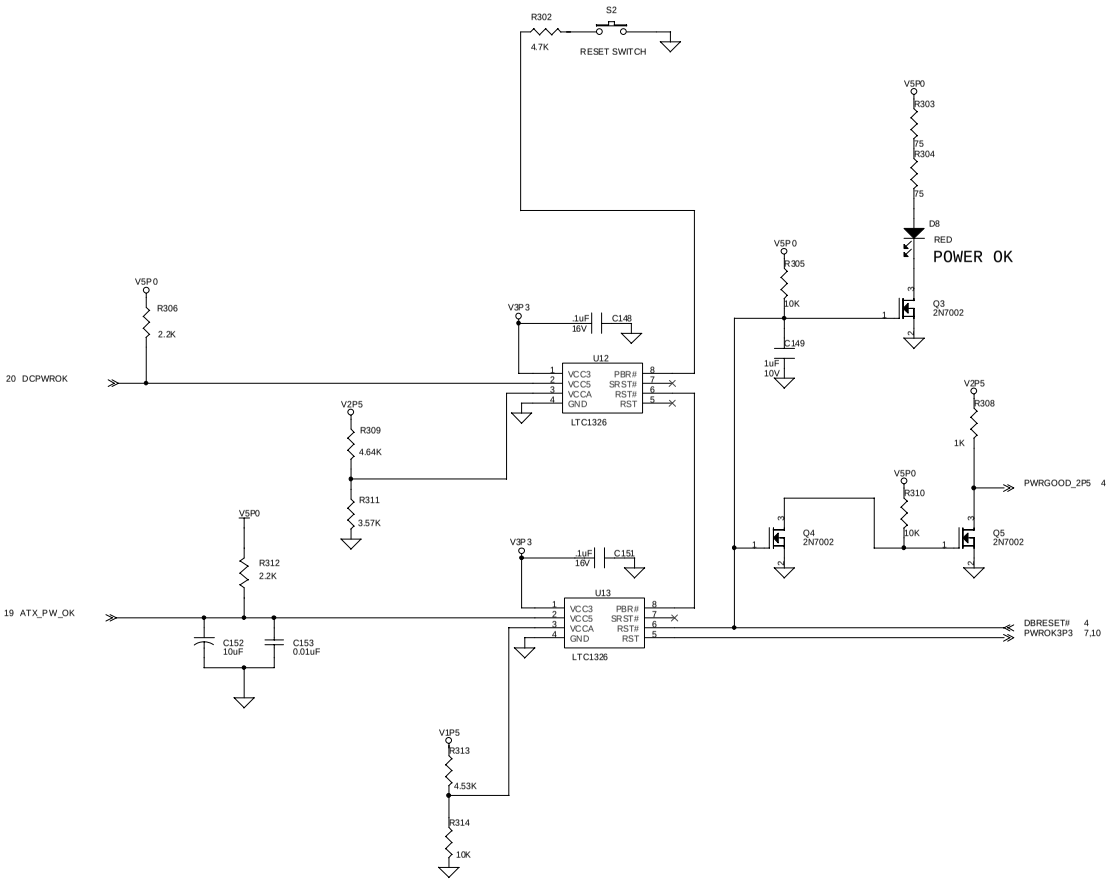


PCI Pullups

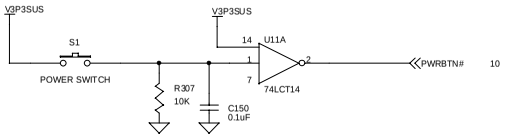


Intel Embedded Microprocessor Division			
5000 W. Chandler Blvd. Chandler AZ. 85226			
Title	Celeron Processor Based TT Sample Deisgn		
ISA/PCI Pullups			
Size C	Document Number		Rev A
Date:	Sheet 17 of 34		

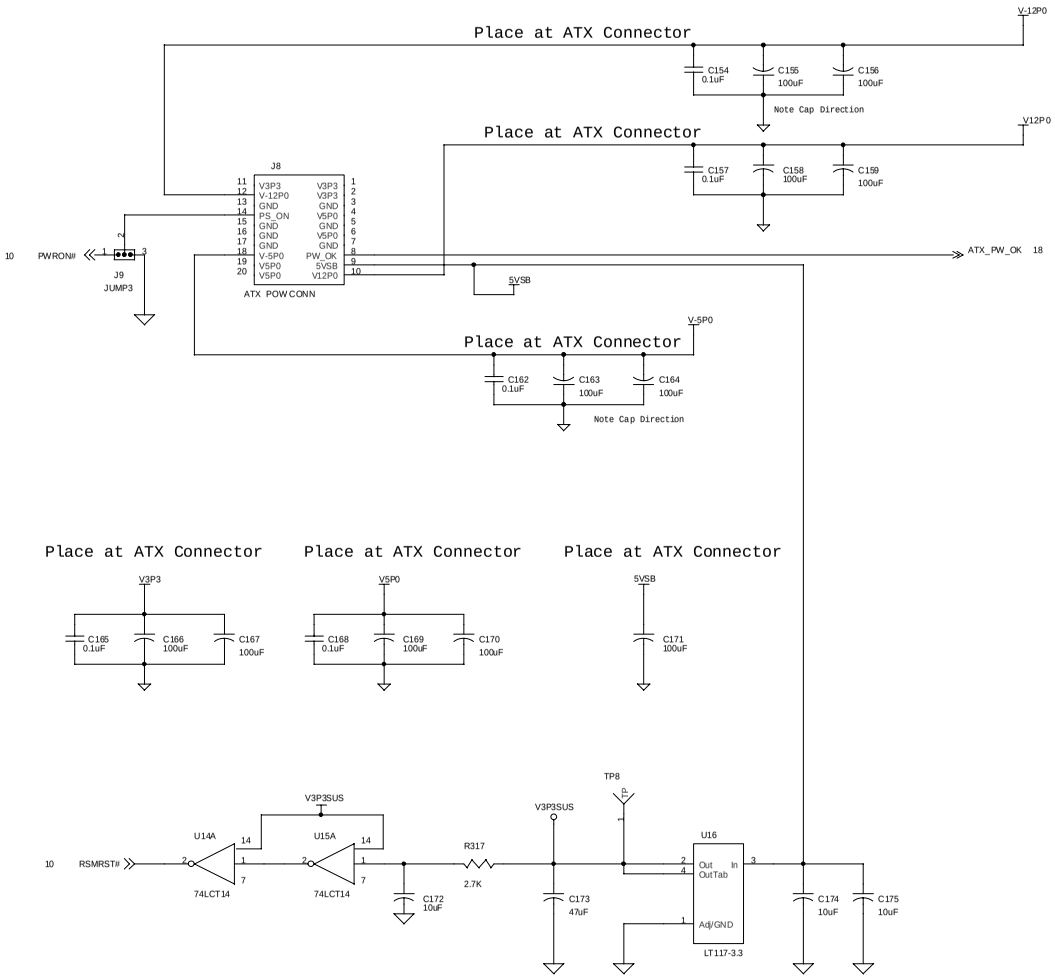
RESET AND POWER GOOD CIRCUITRY



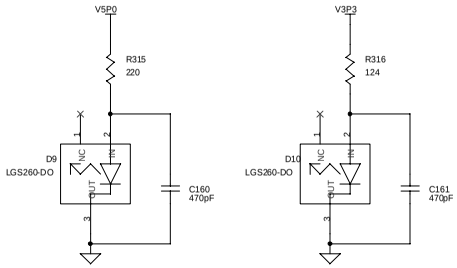
ATX POWER SWITCH CIRCUITRY



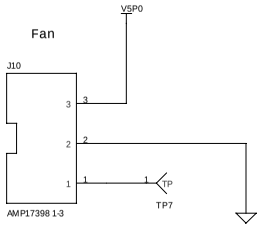
Intel Embedded Microprocessor Division 5000 W. Chandler Blvd. Chandler AZ. 85226		
Title: Celeron Processor Based TT Sample Design Reset Circuitry		
Size: C	Document Number	Rev: A
Date: 6/16/1999	Sheet 18 of 34	

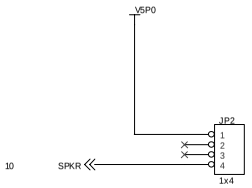


Power Indicators



Note: Add screen marking for V5_0 LED, V3_3 LED

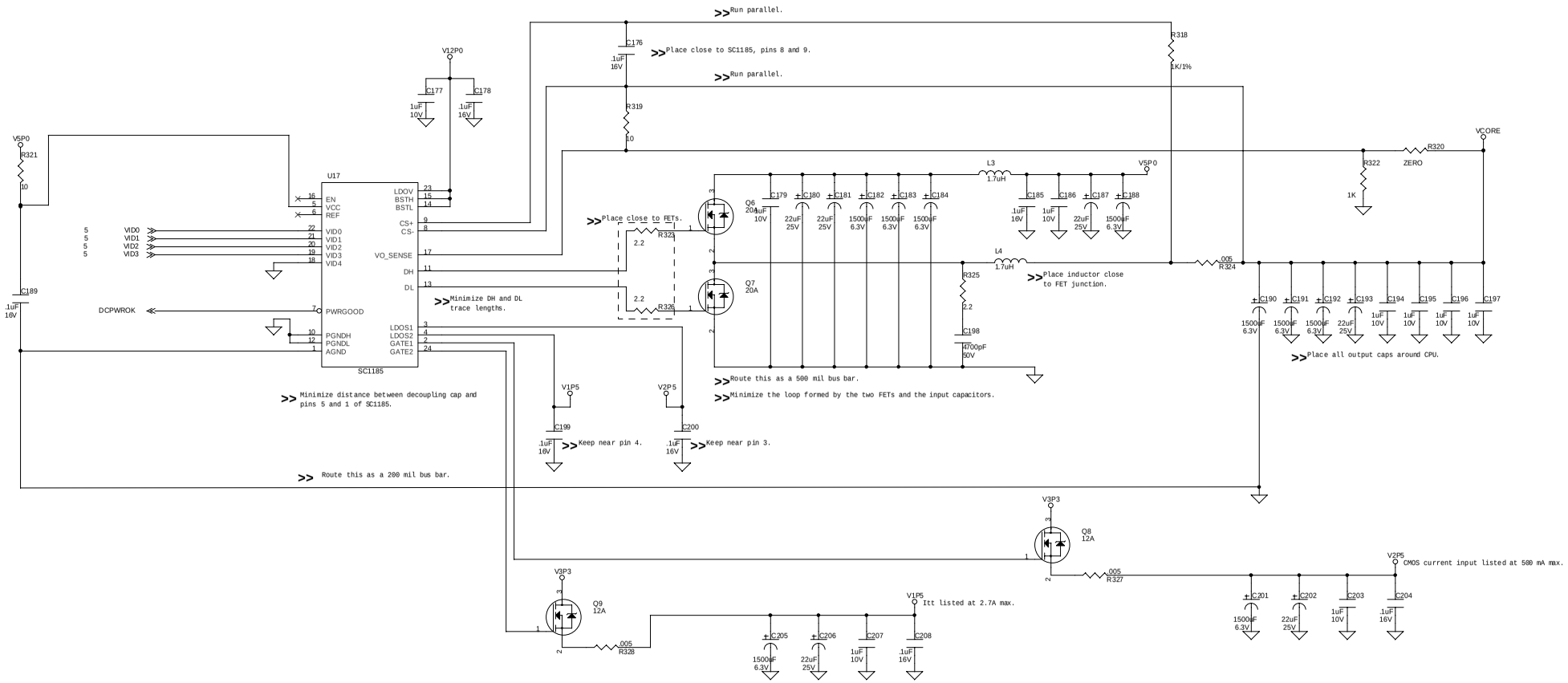




SPEAKER HEADER

NOTE 1: Locate Fan Header next to Processor Assembly
Note 2: Connect V5_0 feed for analog supplies AVDD1 and CLKAVDD1 at one point only on PCB
Note 3: Place power connector decoupling at ATX connector
NOTE 4: Note direction of caps on -12V, -5V supplies.

Intel Embedded Microprocessor Division 5000 W. Chandler Blvd. Chandler AZ. 85226		
Title	Celeron Processor Based TT Sample Design ATX Power Connector	
Size C	Document Number	Rev A
Date:	Sheet 19 of 34	



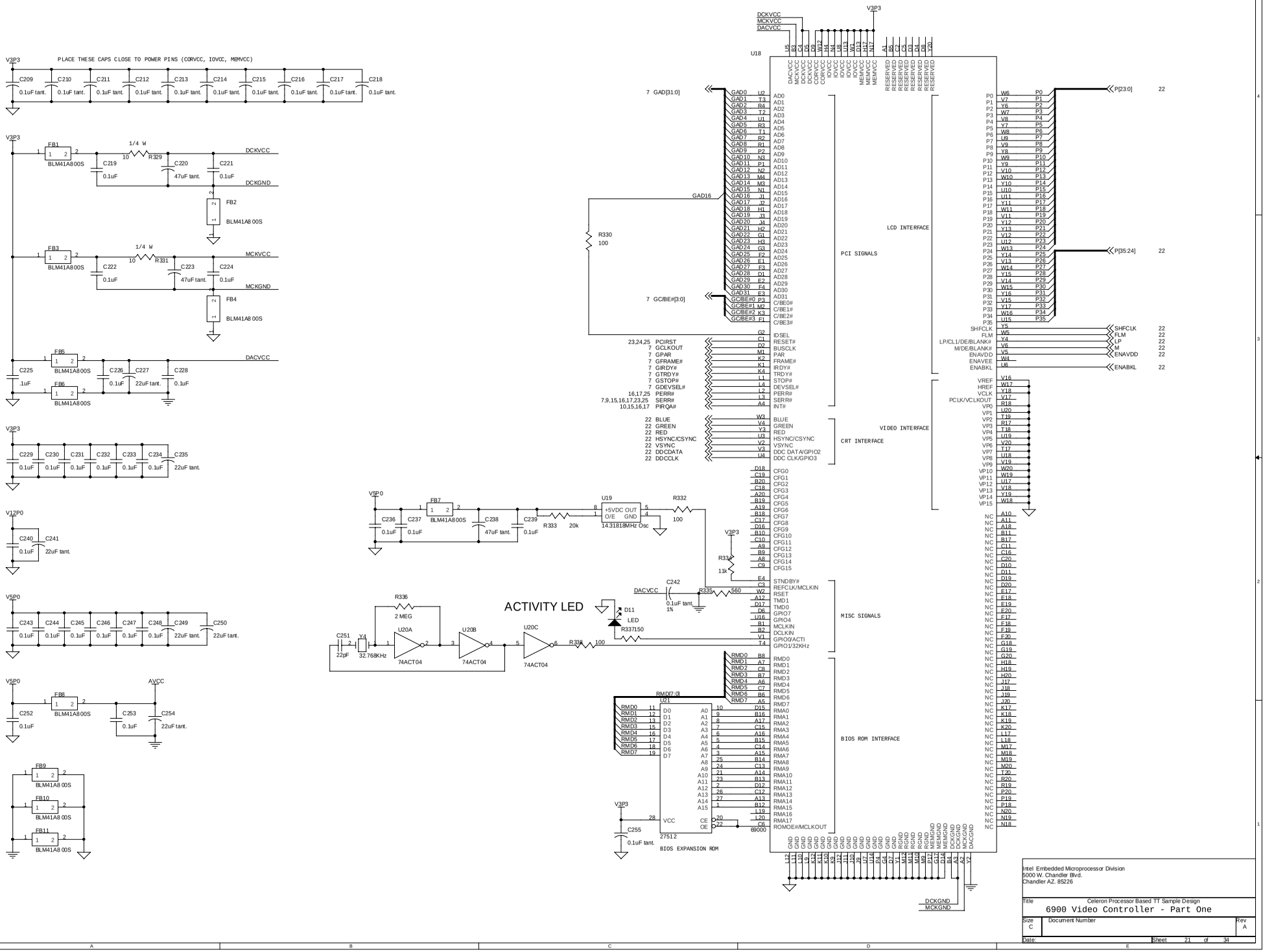
Intel Embedded Microprocessor Division

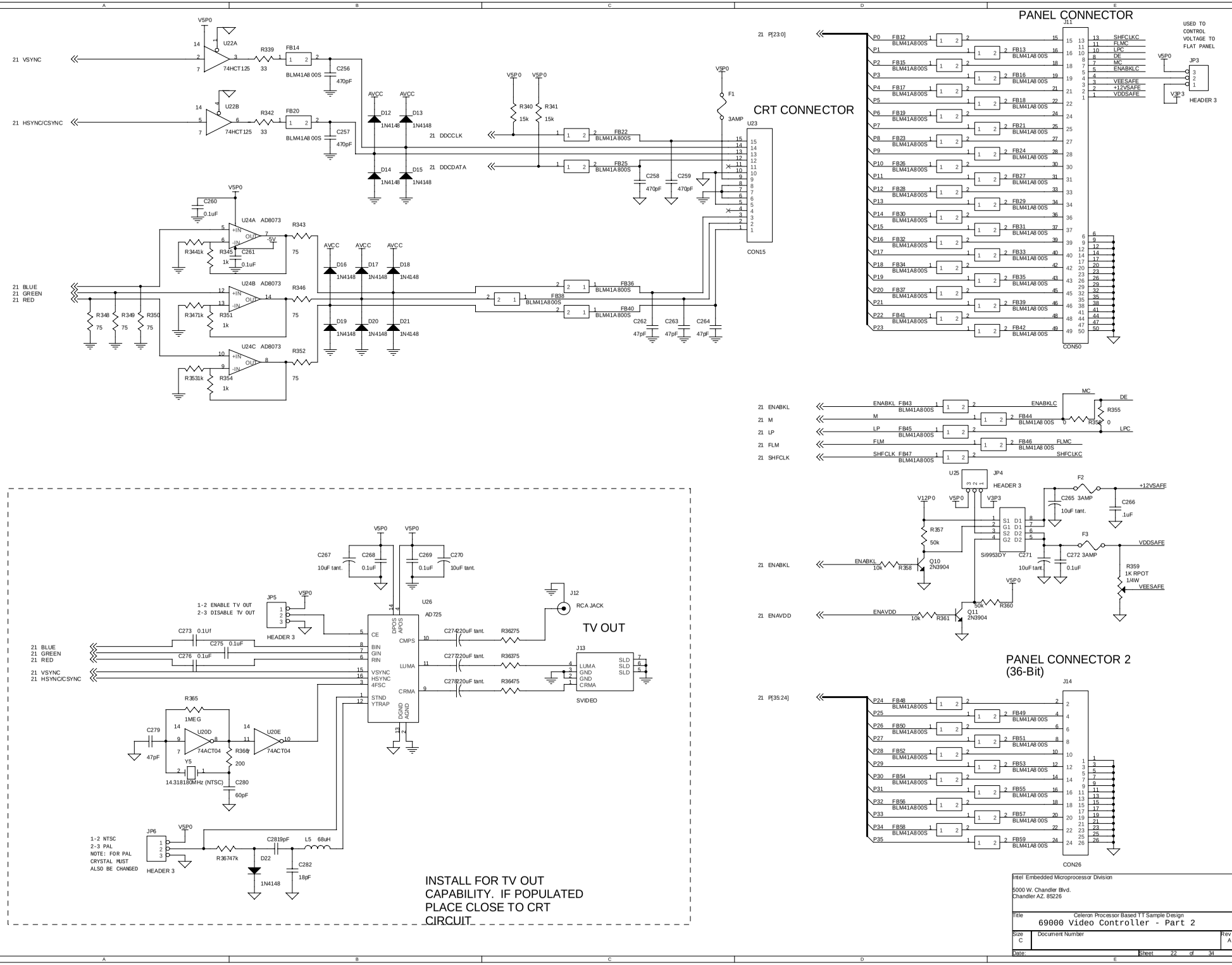
5000 W. Chandler Blvd.
Chandler AZ. 85226

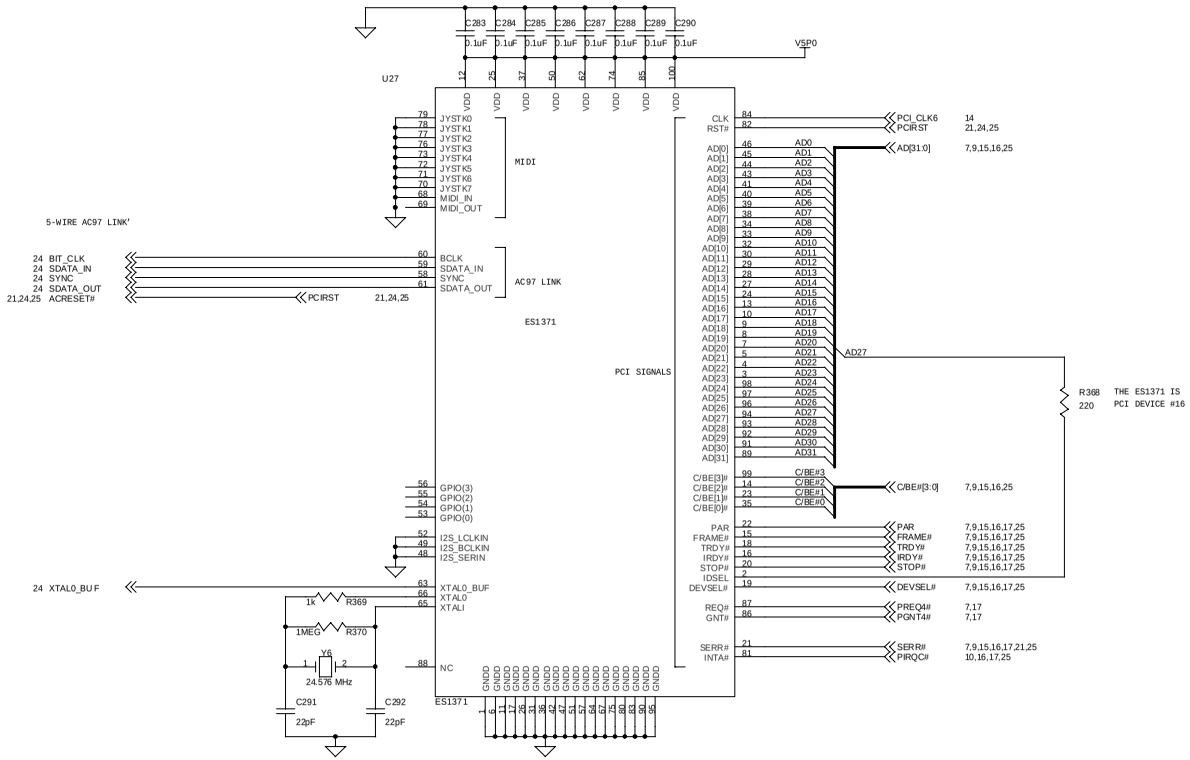
Title: Celeron Processor Based TT Sample Design
DC to DC Convertor

Size: C | Document Number | Rev: A

Date: 6/16/1999 | Sheet 20 of 34


PLACE THESE CAPS CLOSE TO POWER PINS (CORVCC, IOVCC, MEMVCC)
V3P3
C209 C210 C211 C212 C213 C214 C215 C216 C217 C218
0.1uF tant.
FB1 BLM41A800S
1/4 W
R329 10
DCKVCC
DCKGND
C219 0.1uF
C220 47uF tant.
C221 0.1uF
FB2 BLM41A800S
FB3 BLM41A800S
R331 10
MCKVCC
MCKGND
C222 C223 C224
FB4 BLM41A800S
FB5 FB6 BLM41A800S
DACVCC
C225 .1uF
C226 C227 C228
0.1uF 22uF tant.
C229 C230 C231 C232 C233 C234 C235
22uF tant.
V12P0
C240 C241
V5P0
C243 C244 C245 C246 C247 C248 C249 C250
C252 C253 C254
AVCC
FB8 FB9 FB10 FB11 BLM41A800S
U18
DCKVCC MCKVCC DACVCC
7 GAD[31:0]
GAD16
R330 100
7 GC/BE#[3:0]
PCI SIGNALS
LCD INTERFACE
P[23:0] 22
P[35:24] 22
SHFCLK FLM LP/CL1/DE/BLANK# M/DE/BLANK# ENAVDD ENAVEE ENABKL
VIDEO INTERFACE
CRT INTERFACE
23,24,25 PCIRST
7 GCLKOUT
7 GPAR
7 GFRAME#
7 GIRDY#
7 GTRDY#
7 GSTOP#
7 GDEVSEL#
16,17,25 PERR#
7,9,15,16,17,23,25 SERR#
10,15,16,17 PIRQA#
22 BLUE
22 GREEN
22 RED
22 HSYNC/CSYNC
22 VSYNC
22 DDCDATA
22 DDCCLK
MISC SIGNALS
BIOS ROM INTERFACE
V5P0
FB7 BLM41A800S
C236 C237 C238 C239
U19 +5VDC OUT O/E GND
14.31818MHz Osc
R333 20k
R332 100
R334 11k
C242 0.1uF tant. 1%
R335 560
ACTIVITY LED
D11 LED
R337 150
R336 2 MEG
C251 22pF
Y4 32.768KHz
U20A U20B U20C 74ACT04
R338 100
RMD7:0
U21
27512
BIOS EXPANSION ROM
C255 0.1uF tant.
DCKGND MCKGND
Intel Embedded Microprocessor Division
5000 W. Chandler Blvd.
Chandler AZ. 85226
Title
Celeron Processor Based TT Sample Design
6900 Video Controller - Part One
Size C
Document Number
Rev A
Date:
Sheet 21 of 34


CRT CONNECTOR
PANEL CONNECTOR
PANEL CONNECTOR 2
(36-Bit)
TV OUT
INSTALL FOR TV OUT
CAPABILITY. IF POPULATED
PLACE CLOSE TO CRT
CIRCUIT
USED TO
CONTROL
VOLTAGE TO
FLAT PANEL
CON15
CON50
CON26
HEADER 3
SVIDEO
RCA JACK
1-2 ENABLE TV OUT
2-3 DISABLE TV OUT
1-2 NTSC
2-3 PAL
NOTE: FOR PAL
CRYSTAL MUST
ALSO BE CHANGED
Intel Embedded Microprocessor Division
5000 W. Chandler Blvd.
Chandler AZ. 85226
Title
Celeron Processor Based TT Sample Design
69000 Video Controller - Part 2
Size
C
Document Number
Rev
A
Date:
Sheet 22 of 34

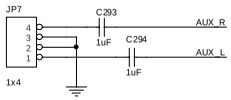


Intel Embedded Microprocessor Division

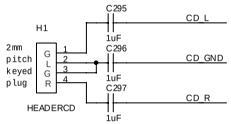
5000 W. Chandler Blvd.
Chandler AZ. 85226

Title	Celeron Processor Based TT Sample Design AC 97 Audio - Part One	
Size C	Document Number	Rev A
Date:	Sheet 23 of 34	

AUX INPUT

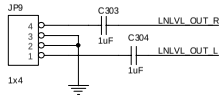


CDROM AUDIO HEADER

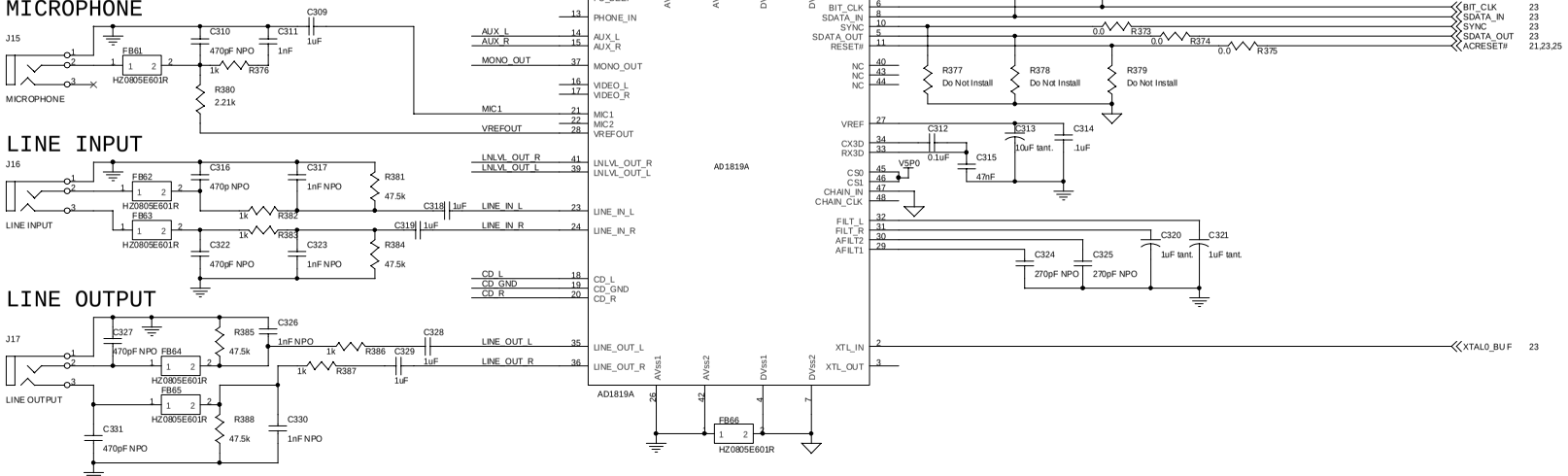


MONO OUT

LINE LEVEL OUT



MICROPHONE



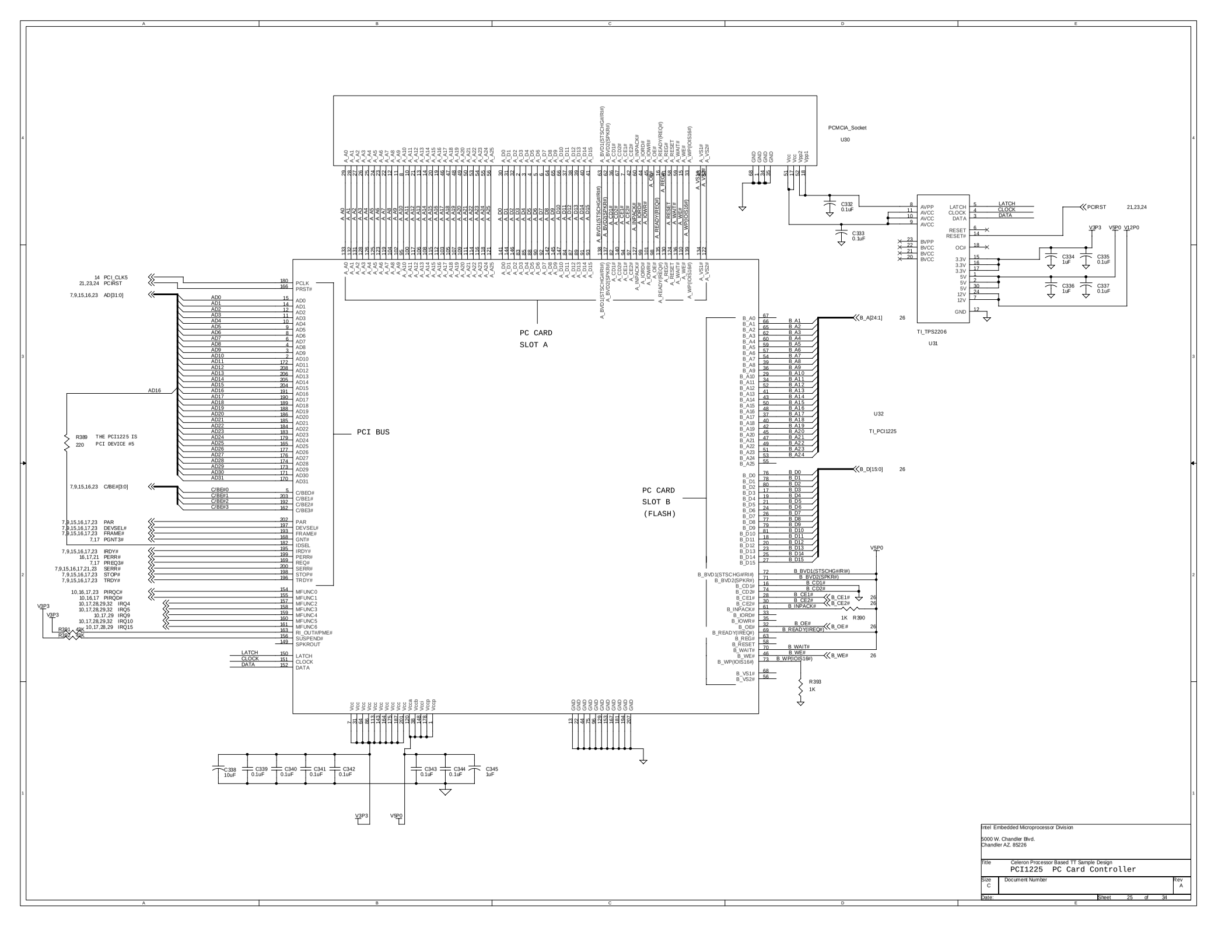
LINE INPUT

LINE OUTPUT

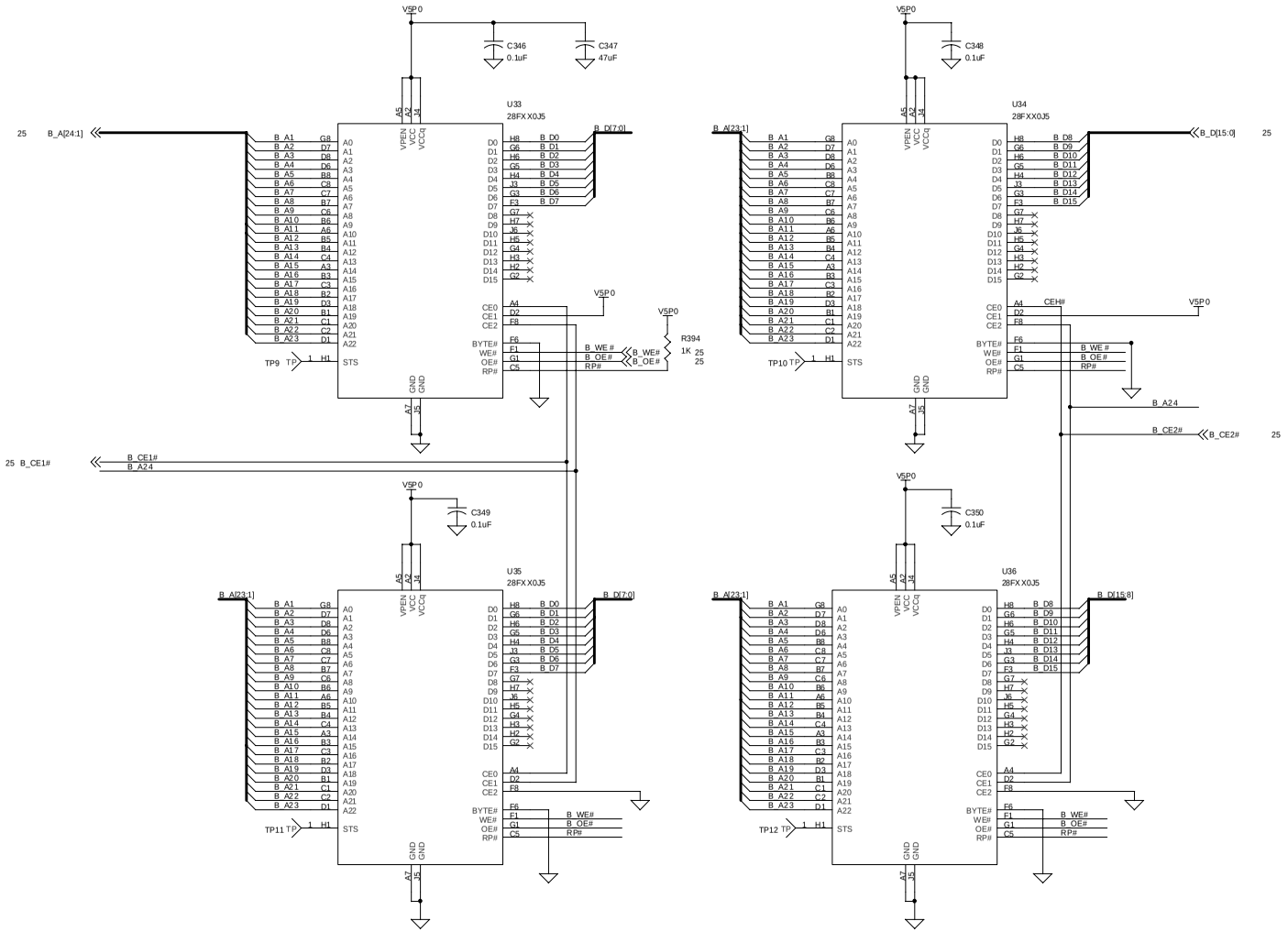
ALL RESISTORS AND CAPACITORS ARE 1/8 W

SEE APPLICATION NOTE FOR LAYOUT GUIDELINES

Intel Embedded Microprocessor Division 5000 W. Chandler Blvd. Chandler AZ. 85226		
Title	Celeron Processor Based TT Sample Design AC 97 Audio - Part Two	
Size C	Document Number	Rev A
Date:	Sheet 24 of 34	



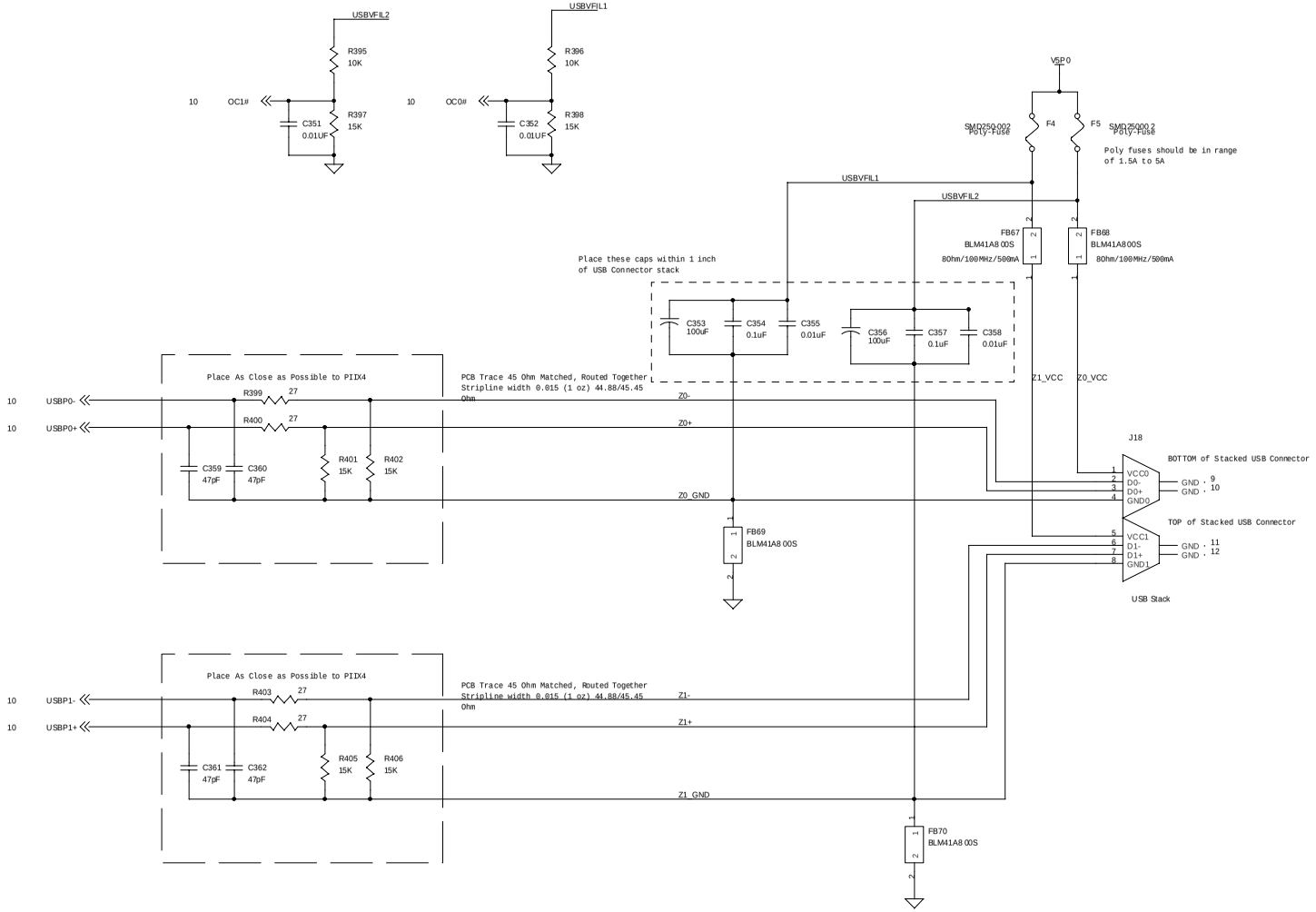
Intel Embedded Microprocessor Division	
5000 W. Chandler Blvd. Chandler AZ. 85226	
Title	Celeron Processor Based TT Sample Design PCI1225 PC Card Controller
Size: C	Document Number:
Rev: A	
Date:	Sheet 25 of 34



Intel Embedded Microprocessor Division

5000 W. Chandler Blvd.
Chandler AZ. 85226

Title	Celeron Processor Based TT Sample Design StrataFlash	
Size C	Document Number	Rev A
Date:	Sheet 26 of 34	



NOTE 1: USB differential traces route together (Z0- & Z0+) and (Z1- & Z1+).
Must be 45 Ohm Matched
Stripline width 0.015 (for 1 oz)->44.88/45.45 Ohm.

NOTE 2: Protect differential traces w/ guard traces or
double space to any other signal.

NOTE 3: Place ferrites at connector.

NOTE 4: Poly-fuse min 1.5A
max 5A.

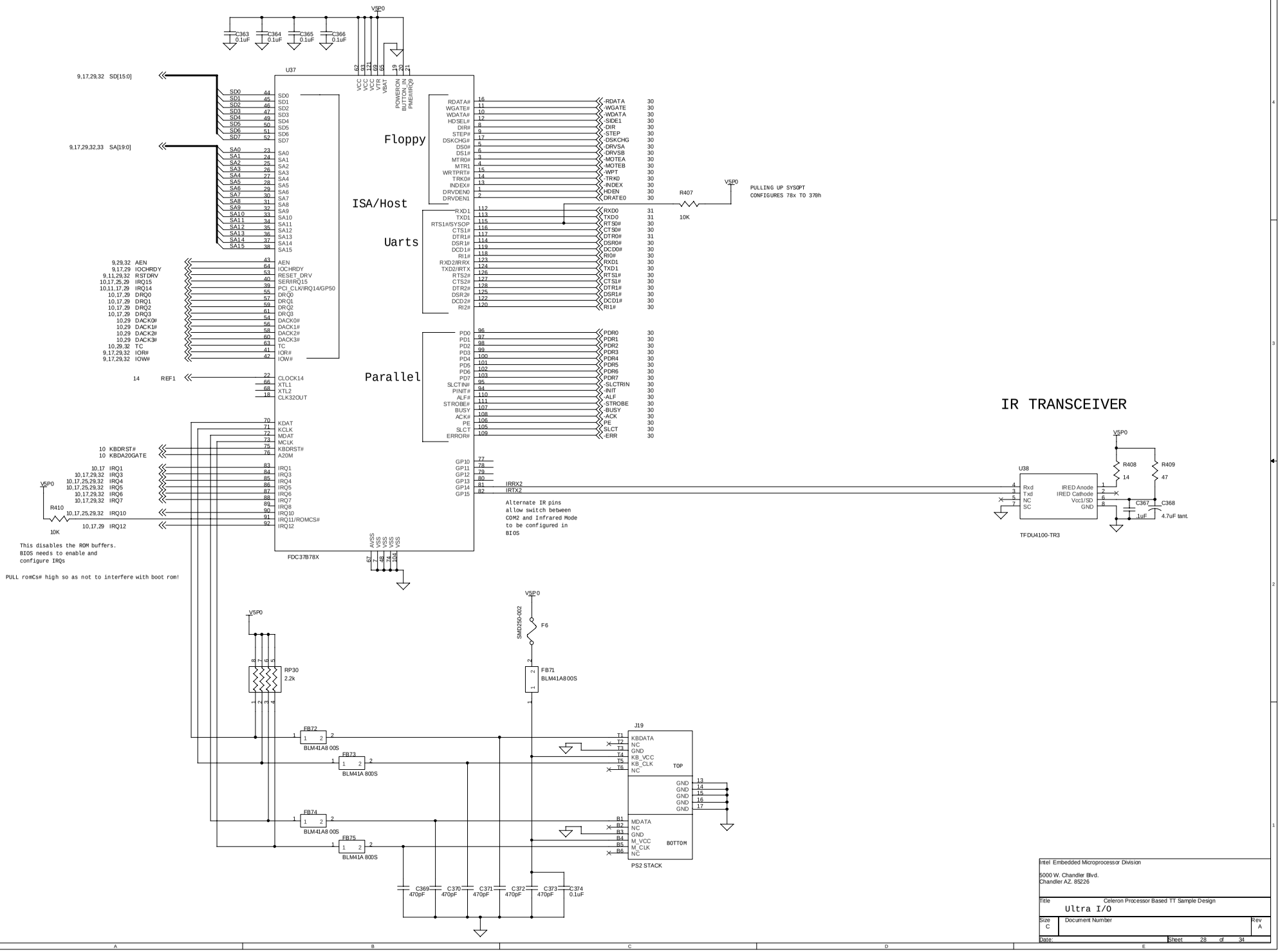
Intel Embedded Microprocessor Division

5000 W. Chandler Blvd.
Chandler AZ. 85226

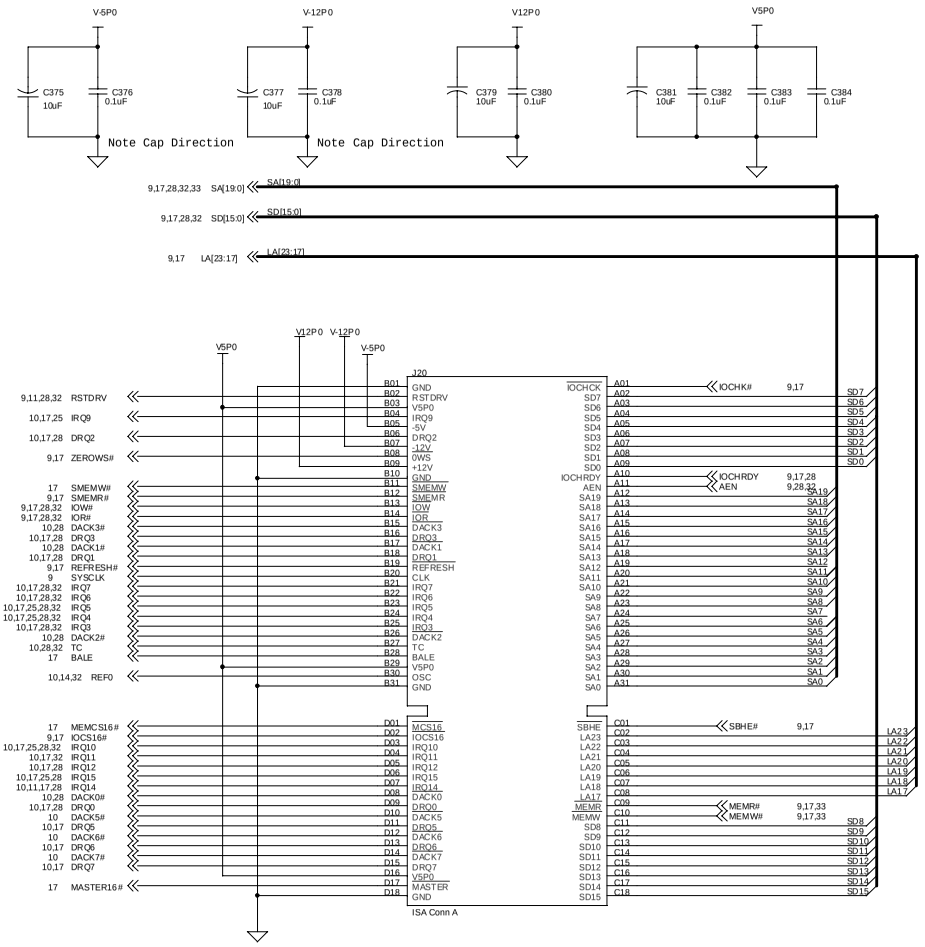
Title: Celeron Processor Based TT Sample Design
USB Connectors

Size: C | Document Number | Rev: A

Date: | Sheet 27 of 34


U37
FDC37B78X
Floppy
ISA/Host
Uarts
Parallel
IR TRANSCEIVER
U38
TFDU4100-TR3
PS2 STACK
J19
This disables the ROM buffers.
BIOS needs to enable and
configure IRQs
PULL romCs# high so as not to interfere with boot rom!
PULLING UP SYSOPT
CONFIGURES 78x TO 370h
Alternate IR pins
allow switch between
COM2 and Infrared Mode
to be configured in
BIOS
Intel Embedded Microprocessor Division
5000 W. Chandler Blvd.
Chandler AZ. 85226
Title
Celeron Processor Based TT Sample Design
Ultra I/O
Size
C
Document Number
Rev
A
Date:
Sheet 28 of 34

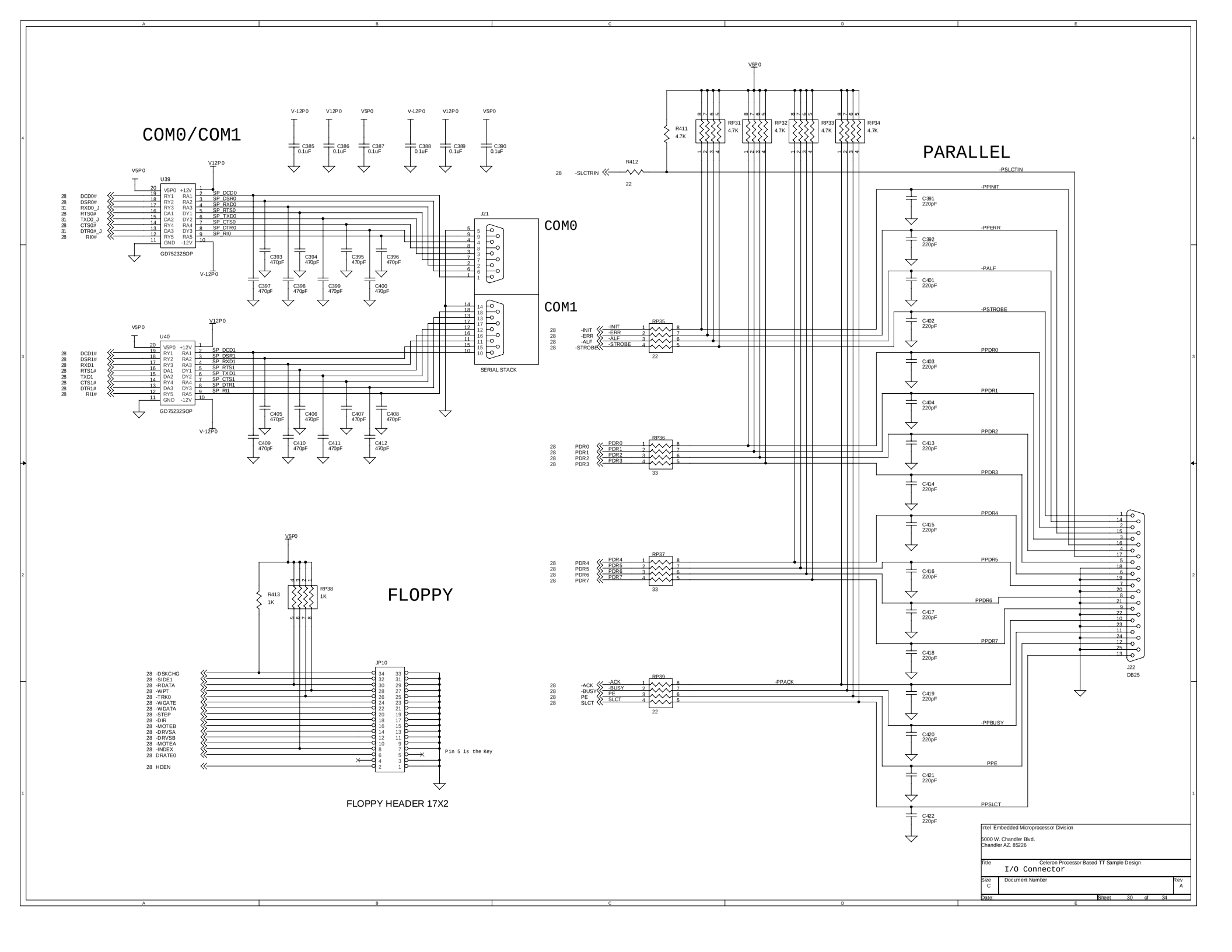
ISA Slot(s)

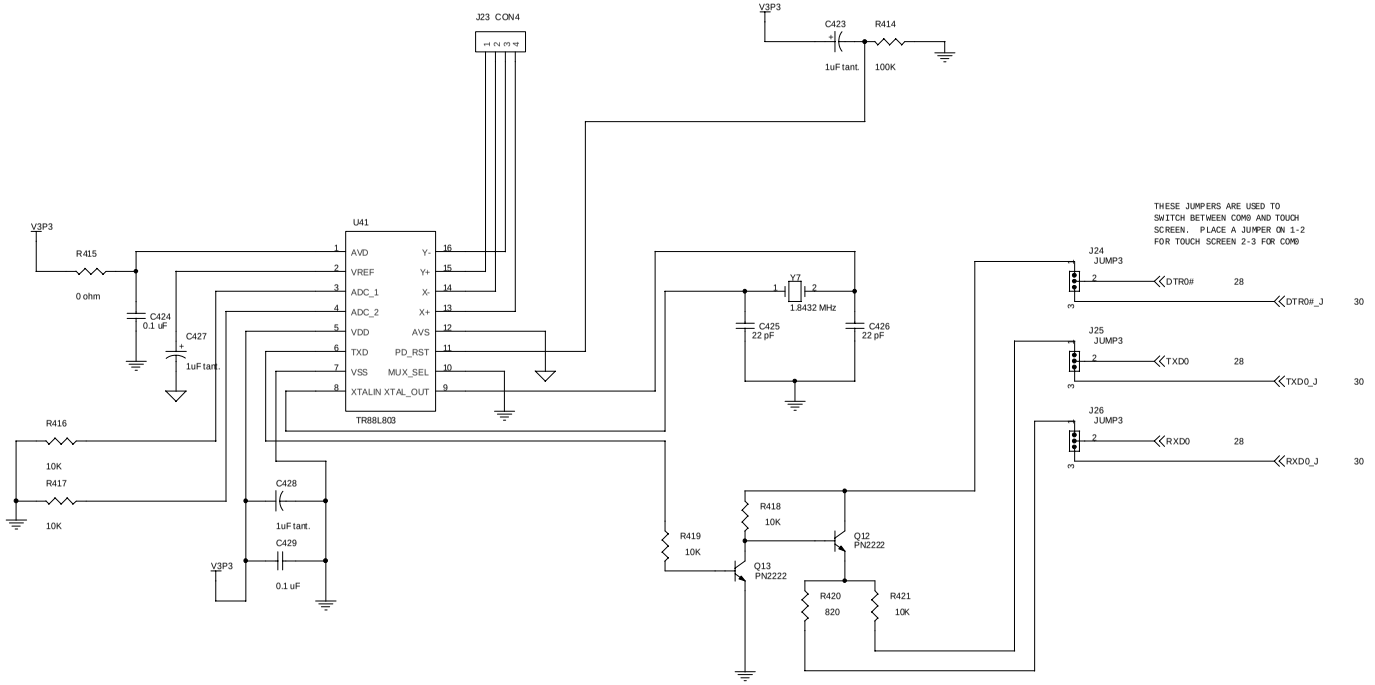


Intel Embedded Microprocessor Division

5000 W. Chandler Blvd.
Chandler AZ. 85226

Title	Celeron Processor Based TT Sample Design ISA Connector	
Size C	Document Number	Rev A
Date:	Sheet 29 of 34	


COM0/COM1
V-12P0
V12P0
V5P0
C385 0.1uF
C386 0.1uF
C387 0.1uF
C388 0.1uF
C389 0.1uF
C390 0.1uF
U39
GD75232SOP
SP_DCD0
SP_DSR0
SP_RXD0
SP_RTS0
SP_TXD0
SP_CTS0
SP_DTR0
SP_RI0
DCD0#
DSR0#
RXD0_J
RTS0#
TXD0_J
CTS0#
DTR0#_J
RI0#
C393 470pF
C394 470pF
C395 470pF
C396 470pF
C397 470pF
C398 470pF
C399 470pF
C400 470pF
J21
COM0
COM1
U40
SP_DCD1
SP_DSR1
SP_RXD1
SP_RTS1
SP_TXD1
SP_CTS1
SP_DTR1
SP_RI1
DCD1#
DSR1#
RXD1
RTS1#
TXD1
CTS1#
DTR1#
RI1#
C405 470pF
C406 470pF
C407 470pF
C408 470pF
C409 470pF
C410 470pF
C411 470pF
C412 470pF
SERIAL STACK
FLOPPY
R413 1K
RP38 1K
JP10
-DSKCHG
-SIDE1
-RDATA
-WPT
-TRK0
-WGATE
-WDATA
-STEP
-DIR
-MOTEB
-DRVSA
-DRVSB
-MOTEA
-INDEX
DRATE0
HDEN
Pin 5 is the Key
FLOPPY HEADER 17X2
PARALLEL
R411 4.7K
RP31 4.7K
RP32 4.7K
RP33 4.7K
RP34 4.7K
R412 22
-SLCTRIN
-PSLCTIN
-PPINIT
-PPERR
-PALF
-PSTROBE
PPDR0
PPDR1
PPDR2
PPDR3
PPDR4
PPDR5
PPDR6
PPDR7
-PPBUSY
PPE
PPSLCT
-PPACK
RP35
RP36
RP37
RP39
-INIT
-ERR
-ALF
-STROBE
PDR0
PDR1
PDR2
PDR3
PDR4
PDR5
PDR6
PDR7
-ACK
-BUSY
PE
SLCT
C391 220pF
C392 220pF
C401 220pF
C402 220pF
C403 220pF
C404 220pF
C413 220pF
C414 220pF
C415 220pF
C416 220pF
C417 220pF
C418 220pF
C419 220pF
C420 220pF
C421 220pF
C422 220pF
J22
DB25
Intel Embedded Microprocessor Division
5000 W. Chandler Blvd.
Chandler AZ. 85226
Title
Celeron Processor Based TT Sample Design
I/O Connector
Size C
Document Number
Rev A
Date:
Sheet 30 of 34



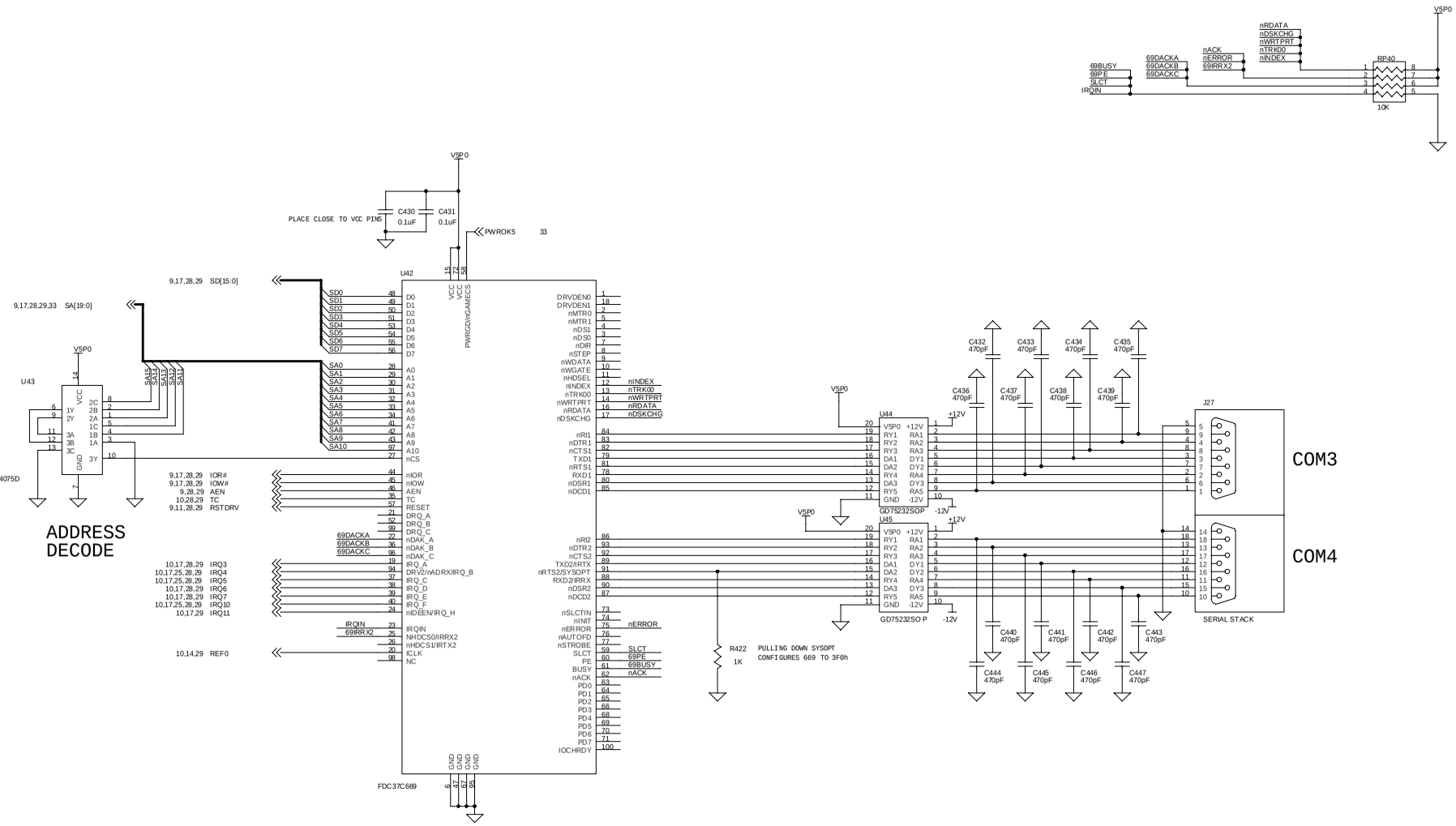
Intel Embedded Microprocessor Division

5000 W. Chandler Blvd.
Chandler AZ. 85226

Title: Celeron Processor Based TT Sample Design
Touch Screen Controller

Size: C | Document Number | Rev: A

Date: | Sheet 31 of 34

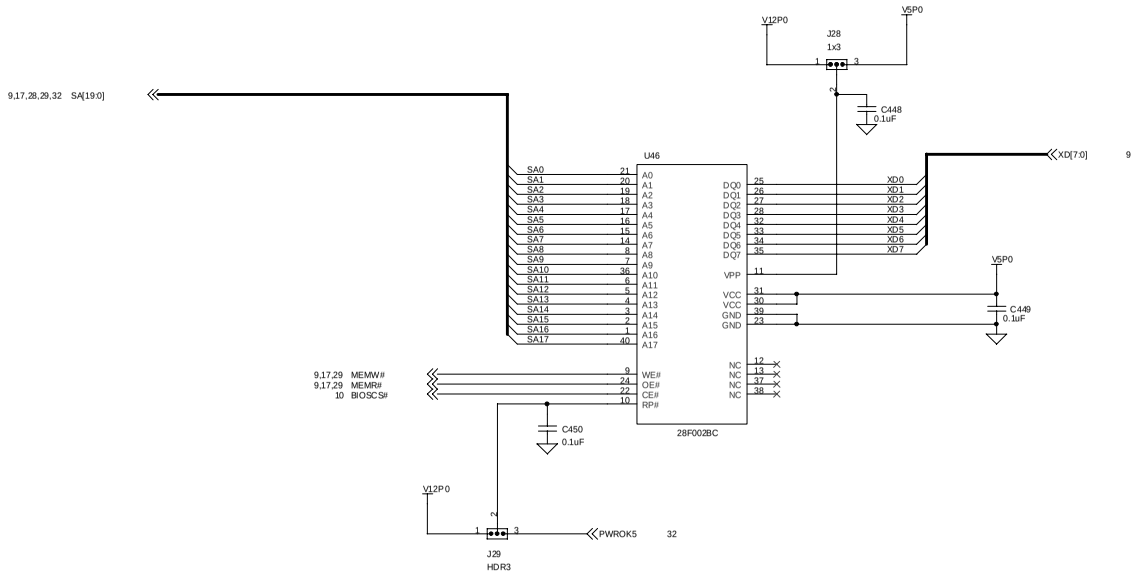


THIS SUPER I/O IS USED SOLELY TO ADD SERIAL PORTS TO THE BOARD.

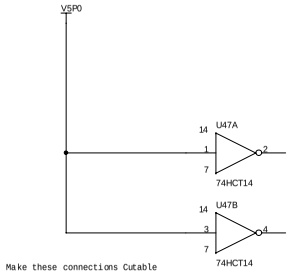
Intel Embedded Microprocessor Division

5000 W. Chandler Blvd.
Chandler AZ. 85226

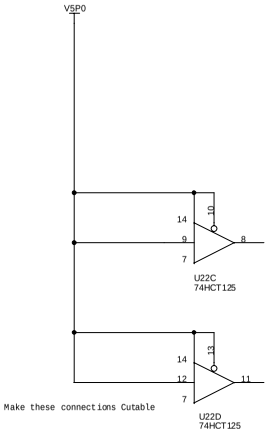
Title	Celeron Processor Based TT Sample Design Additional Serial Ports	
Size C	Document Number	Rev A
Date:	Sheet 32 of 34	



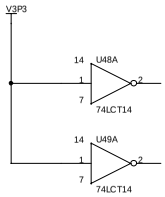
Intel Embedded Microprocessor Division 5000 W. Chandler Blvd. Chandler AZ. 85226		
Title	Celeron Processor Based TT Sample Design BIOS	
Size C	Document Number	Rev A
Date:	Sheet 33 of 34	



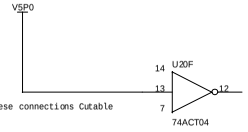
Make these connections Cutable



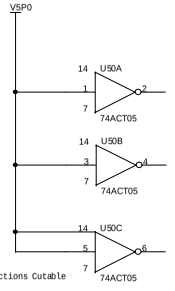
Make these connections Cutable



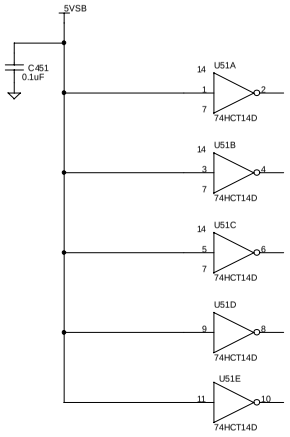
Make these connections Cutable



Make these connections Cutable



Make these connections Cutable



Make these connections Cutable

Intel Embedded Microprocessor Division 5000 W. Chandler Blvd. Chandler AZ. 85226		
Title: Celeron Processor Based TT Sample Design Unused Gates		
Size: C	Document Number	Rev: A
Date:	Sheet 34 of 34	