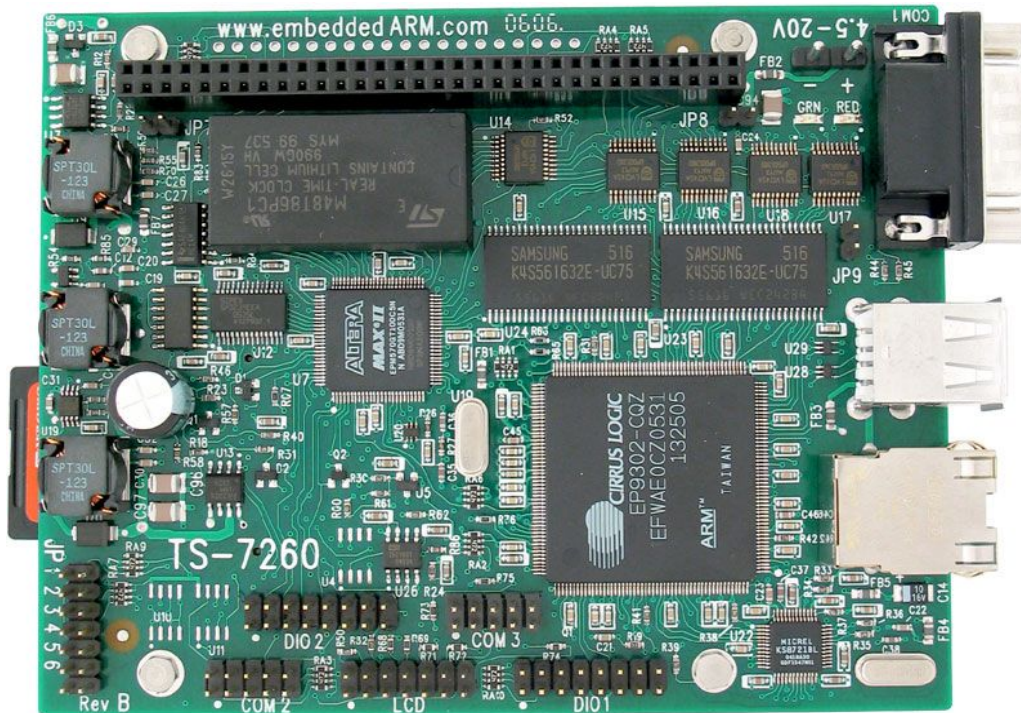


# TS-7260 Hardware Manual



## Feedback and Update to this Manual

To help our customers make the most of our products, we are continually making additional and updated resources available on the **Technologic Systems website** ([www.embeddedARM.com](http://www.embeddedARM.com)).

These include manuals, application notes, programming examples, and updated software and firmware. Check in periodically to see what's new!

When we are prioritizing work on these updated resources, feedback from customers (and prospective customers) is the number one influence. If you have questions, comments, or concerns about your Embedded Computer, please let us know at [support@embeddedARM.com](mailto:support@embeddedARM.com).

## Limited Warranty

Technologic Systems warrants this product to be free of defects in material and workmanship for a period of one year from date of purchase.

During this warranty period Technologic Systems will repair or replace the defective unit in accordance with the following process:

A copy of the original invoice must be included when returning the defective unit to Technologic Systems, Inc.

This limited warranty does not cover damages resulting from lightning or other power surges, misuse, abuse, abnormal conditions of operation, or attempts to alter or modify the function of the product.

*This warranty is limited to the repair or replacement of the defective unit. In no event shall Technologic Systems be liable or responsible for any loss or damages, including but not limited to any lost profits, incidental or consequential damages, loss of business, or anticipatory profits arising from the use or inability to use this product.*

Repairs made after the expiration of the warranty period are subject to a repair charge and the cost of return shipping. Please, contact Technologic Systems to arrange for any repair service and to obtain repair charge information.

## FCC Advisory Statement

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used properly (that is, in strict accordance with the manufacturer's instructions), may cause interference to radio and television reception. It has been type tested and found to comply with the limits for a Class A computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the owner will be required to correct the interference at his own expense.

If this equipment does cause interference, which can be determined by turning the unit on and off, the user is encouraged to try the following measures to correct the interference:

- Reorient the receiving antenna.
- Relocate the unit with respect to the receiver.
- Plug the unit into a different outlet so that the unit and receiver are on different branch circuits.
- Ensure that mounting screws and connector attachment screws are tightly secured.
- Ensure that good quality, shielded, and grounded cables are used for all data communications.

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. The following booklets prepared by the Federal Communications Commission (FCC) may also prove helpful:

- How to Identify and Resolve Radio-TV Interference Problems (Stock No. 004-000-000345-4)
- Interface Handbook (Stock No. 004-000-004505-7)

These booklets may be purchased from the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402.

**TABLE OF CONTENTS**


---

<b>1 INTRODUCTION.....</b>	<b>6</b>
1.1 About this Manual.....	6
1.2 TS-72XX Series.....	6
1.3 Product Overview.....	6
1.4 Benefits.....	7
Out-of-the-Box Productivity.....	7
Impressive Performance.....	7
Low Power Consumption.....	7
1.5 Features.....	7
1.6 Configurability.....	8
On-board Options.....	8
External Accessories.....	8
1.7 PC/104 Peripherals.....	9
1.8 TS-ARM Development Kit.....	10
1.9 Software and Support.....	11
Linux OS Support.....	11
Other OS Support.....	11
<b>2 GETTING STARTED.....</b>	<b>12</b>
2.1 Installation Procedure.....	12
Handling the Board Safely.....	12
Setup and Installation Instructions.....	12
Setup Tools.....	12
Setup Procedure.....	12
Disconnecting AC Power.....	12
2.2 Console and Power Up.....	12
2.3 Boot Sequence.....	13
2.4 Loading or Transferring Files.....	14
Transferring Files via the Ethernet Port.....	14
Transferring Files via Flash Memory Device.....	14
Zmodem Downloads.....	14
<b>3 HARDWARE COMPONENTS.....</b>	<b>15</b>
3.1 Processor.....	15
Cirrus EP9302.....	15
MMU.....	16
Interrupts.....	16
3.2 Memory.....	17
On-Board SDRAM.....	17
Battery Backed SRAM.....	17
On-Board NAND Flash.....	18
USB Flash Drive or Compact Flash Card.....	18
SD Memory Card.....	19
3.3 Glue Logic CPLD.....	19
3.4 Real-Time Clock.....	20
3.5 Watchdog Timer.....	20
<b>4 COMMON INTERFACES GENERAL INFORMATION.....</b>	<b>22</b>

4.1	Serial Ports .....	22
4.2	Digital I/O.....	23
4.3	A/D Converters.....	24
<b>5</b>	<b>CONNECTORS AND HEADERS.....</b>	<b>25</b>
5.1	10/100 Base-T Ethernet Connector.....	25
5.2	USB Connector.....	25
5.3	SD Card Connector – CPLD.....	26
5.4	COM1 Connector.....	26
5.5	COM2 Header.....	27
	Optional RS-485 Support.....	27
	RS-485 Quick-Start Procedure.....	28
	Automatic RS-485 TX Enable .....	28
5.6	COM3 Header.....	29
5.7	DIO1 Header.....	30
	SPI Interface.....	31
	Matrix Keypad.....	32
5.8	DIO2 Header.....	32
	Optional COM4 and COM5 .....	33
	Optional SD-Card Socket.....	33
	Cirrus 5 Channel A/D Converter.....	33
	Interpreting Cirrus A/D Converter.....	34
	External Reset/Shutdown Control.....	35
	TS-XDIO.....	35
	TS-XDIO Programming Considerations .....	37
5.9	LCD Header.....	38
	LCD Interface.....	38
5.10	JTAG Header.....	39
5.11	Power Supply Header.....	40
	Power Management.....	40
<b>6</b>	<b>PC/104 BUS EXPANSION.....</b>	<b>42</b>
6.1	Adding Serial Ports .....	43
6.2	Adding Ethernet Ports .....	44
6.3	Adding CAN Bus .....	44
6.4	Adding Video Support .....	44
<b>7</b>	<b>LEDS, JUMPERS AND BUTTONS.....</b>	<b>45</b>
7.1	Status LEDs.....	45
7.2	Jumpers.....	45
7.3	Buttons.....	46
<b>8</b>	<b>SPECIFICATIONS.....</b>	<b>47</b>
<b>9</b>	<b>FURTHER REFERENCES.....</b>	<b>48</b>
	APPENDIX A: DOCUMENT HISTORY.....	49
	APPENDIX B: MEMORY AND REGISTER MAP.....	50
	APPENDIX C: DOWNLOADS - SCHEMATICS AND MECHANICAL DRAWING.....	52
	APPENDIX D: TS-ARM SBC FEATURE MATRIX.....	53

**APPENDIX E: CONTACT TECHNOLOGIC SYSTEMS.....54**

## 1 INTRODUCTION

---

### 1.1 About this Manual

---

This manual is intended to provide the user with an overview of the board and benefits, complete features specifications, and set up procedures. It contains important safety information as well.

### 1.2 TS-72XX Series

---

The TS-72XX series Single Board Computers (SBC's) run on a 200 MHz ARM9 processor with power as low as 1/2 Watt. Low board complexity, low component count, and low power/heat makes for an extremely reliable embedded engine. The TS-72XX SBC's are available in thousands of configurations, many of which are Commercial off the Shelf (COTS) and available to ship today.

The EP9302 processor from Cirrus is the highly integrated 200Mhz ARM9 processor that the TS-72XX SBC's are built around and includes an on-chip 10/100 ethernet, USB, serial, and Flash/SDRAM controller. For example, on the TS-7200 model there is 32 Mb of Micron SDRAM running at 66 Mhz and 8 Mb Intel Strata flash on-board. A supplemental PLD provides glue logic, watchdog timer, Compact Flash IDE, and 8 bit PC/104 support. Integer CPU performance is about 20% faster than our 133 Mhz x86 offerings.

Even with the standard power consumption of 2 Watts, the TS-72XX SBC's run without fans or heat sinks in the temperature range of -20° to +70°C. Extended Temperature -40° to +85°C is also standard, but CPU clock must be decreased to about 166MHz for higher temperatures. Digital Signal Processing (DSP) is enabled through a standard 5 channel, 12bit A/D converter (Optional 8 channel, 12 bit A/D converter), 20 DIO lines and 2 standard serial ports.

The 8/16 bit PC/104 interface enables additional functionality through Technologic Systems' broad product line of PC/104 peripheral daughter boards. The TS-7KV adds video, CAN, Com Ports, and A/D conversion. The TS-ETH10 allows the addition of Ethernet ports. The TS-CAN adds CAN connectivity. The TS-Modem boards add both wired and cell phone capabilities.

The TS-72XX rugged ARM9 SBC's have found their way into many embedded applications. Customers are using the TS-72XX series SBC's in: energy generation, manufacturing process control, traffic management, printing system management, communication infrastructure, website hosting, data gathering and laboratory test equipment. We use a TS-7200 to host our complete website and to prepare and test your SBC prior to shipping.

### 1.3 Product Overview

---

The **TS-7260** is a compact, full-featured Single Board Computer (SBC) based upon the Cirrus EP9302 ARM9 CPU. The EP9302 features an advanced 200 MHz ARM920T processor design with a memory management unit (MMU) that allows support for high level operating systems such as Linux, Windows CE, and other. As a general-purpose processor, it provides a standard set of peripherals on board and a full set of Technologic Systems featured peripherals via the standard PC/104 Bus.

**Note**

This board includes software power control for on board peripherals to minimize power consumption (as low as 1/2 Watt) for use in power sensitive designs – ideal for battery-powered applications.

## 1.4 Benefits

---

### *Out-of-the-Box Productivity*

Technologic Systems Linux products get you to your application quickly. Our Single Board Computers boot directly to Linux as shipped. There is no complicated CMOS setup or configuring of a Linux derivative Operating System to source, define, and load. Technologic Systems has pre-configured each SBC in flash memory.

The **TS-7260's** user can power up the board and immediately begin application development. Of course, should you wish to configure your own version of Linux or use a different operating system, this is easy too. Technologic Systems provides the solution to fast application development without tedious OS configuration.

### *Impressive Performance*

The ARM920T's 32-bit architecture, with a five-stage pipeline, delivers very impressive performance at very low power. The EP9302 CPU has a 16 KB instruction cache and a 16 KB data cache to provide zero-cycle latency to the current program and data, or they can be locked to guarantee no-latency access to critical sections of instructions and data. For applications with instruction-memory size restrictions, the ARM920T's compressed Thumb instruction set can be used to provide higher code density and lower Flash storage requirements.

As a benchmark, the **TS-7260's** CPU integer performance, at a supplied 200 MHz, is about twice as fast as the Technologic Systems 133MHz 586-based products.

### *Low Power Consumption*

The **TS-7260** is a design optimized for low power consumption. This design allows for power consumption as low as a ¼ watt. There are three on-board switching power supply circuits that generate all of the internal voltages required by the **TS-7260**. This allows for a very flexible input power source of 4.5 to 20 VDC and can be unregulated. The on-board ADC measures input voltage for battery or solar applications, allowing software monitoring of input voltage. It is also possible to control and minimize power consumption using software programming to enable shutdown of USB, LCD, Ethernet, and RS232 level converters.

## 1.5 Features

---

The **TS-7260** comes **standard** with these features:

- ✓ Requires less than 1 watt for full-speed operation
- ✓ 1/4 watt at lower speeds
- ✓ High efficiency switching-mode power supply (4.5-20VDC Unregulated Input)
- ✓ **TS-Linux** Embedded Operating System Installed
- ✓ 200Mhz ARM9 CPU with MMU
- ✓ 32 MB on-board NAND flash (Boots to TS-Linux)
- ✓ 32 MB SDRAM
- ✓ 2 USB 2.0 Compatible OHCI ports (12 Mbit/s Max)
- ✓ 10/100 Ethernet port
- ✓ 3 serial ports standard
- ✓ 30 total DIO pins
- ✓ **TS-XDIO** (8 schmitt-trigger DIO - PWM, quadrature counter, edge counter, pulse timer)
- ✓ 2-channel on-board 12-bit A/D Converter

- ✓ Watchdog timer
- ✓ PC/104 expansion bus
- ✓ SPI bus interface
- ✓ Alphanumeric LCD and matrix keypad interfaces
- ✓ Small size 3.8 x 4.8 inches (9.7 cm x 11.5 cm)
- ✓ Operating Temperature Range: Fanless from -20° to +70°C
- ✓ Extended Temperature -40° to +85°C standard at lower CPU clock speeds (166Mhz)



**Note**

The TS-XDIO interface is the standard feature. The SD Card flash socket (on Revision B) and the two extra COM ports are the available options for **TS-7260**. However, these three features can not be simultaneously selected. **Only one** functionality can be ordered for a single user configuration.

## 1.6 Configurability

The **TS-7260** can be configured for your application using the following available on-board options and external accessories:

### On-board Options

- ✓ **TS-7xxx-yyy-zzzF**: up to “yyy” MB of on-board SDRAM and “zzz” MB of Flash memory upgrade for TS-7xxx board models. For example, **TS-7250-64-128F** selects model **TS-7250** upgrade with 64 MB of SDRAM and 128 of NAND Flash.

<i>Model (xxx)</i>	<i>SDRAM (yyy)</i>	<i>on-board Flash (zzz)</i>
<b>TS-7200</b>	Up to 64 MB	Up to 16 MB
<b>TS-7250</b>	Up to 64-128 MB	Up to 64-128-256 MB
<b>TS-7260</b>	Up to 64-128 MB	Up to 64-128-256 MB

- ✓ **OP-ADC**: 12 bit 8 channel A/D Converter (**TS-7200** and **TS-7250**)
- ✓ **OP-2TTLCOM**: Two additional COM (COM4 and COM5) ports with TTL levels (**TS-7260**)
- ✓ **OP-SDSOCKET**: One SD Card socket for additional flash memory (**TS-7260**)
- ✓ **OP-BBRTC**: on-board sealed-battery backed RTC
- ✓ **OP-TMPSENSE**: High-precision temperature sensor
- ✓ **OP-485-FD**: RS-485 full duplex interface on COM2
- ✓ **OP-485-HD**: RS-485 half duplex interface on COM2
- ✓ **OP-16BIT-104**: 16-bit PC/104 Connector
- ✓ **OP-STHRU-104**: 16-bit Pass-Thru PC/104 Connector
- ✓ **OP-ROHS**: RoHS directive compliant built board



**Note**

The **TS-7260** SBC can be built compliant with the RoHS (Restriction of Hazardous Substances ) Directive. Contact Technologic Systems for RoHS support.

### External Accessories

- ✓ **CF-512-LIN**: 512 MB Compact Flash Card with full ARM tool chain and Debian installed (**TS-7200**)



- ✓ **SD-256:** 256 MB SD Flash Card with full ARM tool chain installed and Debian (**TS-7260**)
- ✓ **SD-512:** 512 MB SD Flash Card with USB Interface, full ARM tool chain installed and Debian (**TS-7260**)
- ✓ **USB Flash Drive:** 256 MB with full ARM tool chain installed and Debian
- ✓ **OP-EJECT:** Compact Flash Ejector (**TS-7200**)
- ✓ **WIFI-G-USB:** Linux-supported USB 802.11g WiFi transceiver for wireless networking
- ✓ **OP-LCD-LED:** Alphanumeric 2x24 LCD with back light and cable
- ✓ **OP-KPAD:** Matrix keypad with cable
- ✓ **PS-5VDC-REG:** Regulated 5VDC Power Supply, 110 VAC Input
- ✓ **TS-ENC720:** Metal Enclosure with Power Converter
- ✓ **RC-DB9:** COM2 adapter cable to DB-9

In addition, a complete set of interfacing cables, connectors, and enclosures is available.

**Note**

Check our website at [www.embeddedARM.com](http://www.embeddedARM.com) for an updated list of options and external accessories

## 1.7 PC/104 Peripherals

Technologic Systems offers many add-on peripherals to complete the requirements of your application. These products directly interface with the **TS-7260** using the PC/104 bus, adding a wide variety of functionalities at very reasonable prices. Some of these functionalities includes:

- ✓ **NVRAM** - adds 32K, 128K, 1MB or 2MB bytes of battery-backed SRAM. Battery backed SRAM provides non-volatile memory with very fast write times and unlimited write cycles, unlike Flash memory. This can be very important if the data is constantly being updated several times per minute, since Flash devices can wear-out after a few million write cycles. It also eliminates the latency that Flash memory has during write cycles. This resource is a byte-wide memory device using a lithium battery that will last a minimum of 10 years with or without power applied.
- ✓ Analog **VIDEO** interface – When a video monitor is needed.
- ✓ **CAN** Bus – Useful for automotive applications.
- ✓ **Modems** – Phone Line or GSM Cellular modems.
- ✓ Additional **Ethernet** ports.
- ✓ Additional **DIO** interface with either 24 or 64 new lines..
- ✓ Additional 12 bit **ADC** and **DAC**: useful for industrial automation applications.
- ✓ Additional **COM** and **Parallel** ports – Make more communication channels available.
- ✓ Power-over-Ethernet
- ✓ **Radios** – Long-range wireless radios, Xbee modules

**Note**

New PC/104 boards are always in development. Contact Technologic Systems or visit the [PC/104 peripherals page](#) at our website for a complete and updated list of additional functionalities that can be added to the **TS-7260** using the PC/104 bus. You can also contact Technologic Systems about your [custom project design](#).

## 1.8 TS-ARM Development Kit

The TS-ARM Development Kit for the **TS-7260** Single Board Computer includes all equipment necessary to boot into the operating system of choice and start working. The development kit is highly recommended for a quick start on application development.

The TS-ARM Development Kit contains a 256 or 512 MB Flash drive (Compact Flash for 7200, USB for 7250 and 7260) which includes:



- ✓ a self-hosting ARM installation of the **Debian Linux 2.0** distribution compiled for ARM
- ✓ gcc 2.95.4 and gcc 3.0 compiler with full tool-chain
- ✓ Build tools and source for JFFS/YAFFS file system in on-board NAND Flash.
- ✓ Hardware test routines source code and other example source code
- ✓ Debian package system: apt-get, tasksel, dselect

The development kit additionally includes:

- ✓ USB Compact Flash reader for **TS-7200**
- ✓ 5 VDC regulated power supply (international versions available)
- ✓ NULL modem cable
- ✓ Adapter cable from 10-pin header to DB9
- ✓ Various cables for connection DIO, LCD, Keypad, etc.
- ✓ Development CD with complete TS-Kernel source, manuals, example code, etc.
- ✓ Printed supporting documentation for TS-72XX's Hardware, Linux for ARM and Development Kit.



### Note

Single Board Computer is not included on the Development Kit (sold separately).

## 1.9 Software and Support

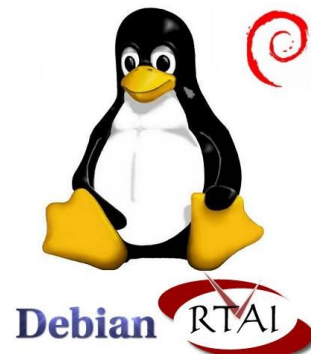
Technologic Systems provides:

- ✓ Free system software and documentation updates available on our web site
- ✓ Free technical support by phone, fax, or email
- ✓ 30-day, money back guarantee on evaluation units
- ✓ One-year, full warranty

### Linux OS Support

The ARM processor (the EP9302) comes from Cirrus and the platform is very similar to the Cirrus EDB9302 evaluation board. Cirrus has strongly promoted running Linux on this chip and has done most of the legwork in creating a patch set to the Linux 2.4 kernels, but we have also had to modify the Linux Kernel (TS-Kernel) so it can support the 8MB on-board Flash chip (via mtd drivers), the compact flash IDE driver, and the A/D converter. If you want to use Linux and aren't tied to the x86 architecture, the TS-72XX boards can be very cost-effective.

The TS-72XX SBC's are shipped standard with the compact TS-Linux embedded operating system installed in the on-board Flash memory. The full-featured Debian Linux can also be used with an NFS root file system or larger Flash drives, such as Compact Flash cards, SD cards and USB flash drives. The TS-Kernel used is based upon the version 2.4.26, patched and compiled for the Cirrus EP9302 ARM920T processor, and is real-time capable through RTAI.



The root file system used by the Linux OS can be any of the following:

- ✓ JFFS/YAFFS file system image in the on-board Flash (if using RedBoot, it should include the option `root=/dev/mtdblock1` to instruct the kernel to boot here)
- ✓ EXT2 file system image in the Compact Flash card (if using RedBoot, it should include the option `root=/dev/hda`)
- ✓ NFS root, via Ethernet port (if using RedBoot, it should include the option `root=/dev/nfs nfsroot=<IP>:<DIRECTORY> ip=dhcp`)



#### Note

The TS-Kernel supports the **Real-Time** Application Interface (RTAI project), making the embedded operating system capable of handling applications with hard real-time restrictions.

### Other OS Support

The **TS-7260** can be loaded with other operating systems such as Windows CE, NetBSD, etc. Technologic Systems will provide support for these, and possibly other operating systems, in the future. Currently, only **Linux** and **NetBSD** are supported on the **TS-7260**.

## 2 GETTING STARTED

---

### 2.1 Installation Procedure

---

Before performing any set up or placement procedures, take the precautions outlined in this section.

#### *Handling the Board Safely*

Be sure to take appropriate Electrostatic Discharge (ESD) precautions. Disconnect the power source before moving, cabling, or performing any set up procedures.

**Warning**  
Inappropriate handling may cause damage to the board.

#### *Setup and Installation Instructions*

Follow these guidelines for safety and maximum product performance:

- ✓ Observe local health and safety requirements and guidelines for manual material handling

#### *Setup Tools*

Depending on placement and cabling, you may need the following tools:

- ✓ Small flat-blade screwdriver
- ✓ Small Phillips screwdriver

#### *Setup Procedure*

After locating, setting up, grounding, and cabling the **TS-7260**:

- ✓ Apply power
- ✓ Monitor the **TS-7260** using a terminal emulator to verify that the board is operating properly

**Note**  
Your board might include a screw power connector on the power input. Notice this connector is removable. Please pull this connector off before applying power.

#### *Disconnecting AC Power*

- ✓ Unplug from the power source.
- ✓ Disconnect other cables as required.

### 2.2 Console and Power Up

---

The TS-72XX SBC's have no video controller or keyboard interface. This was done to keep the board size small and the cost low. COM1 is typically used as a console port to interface the TS-72XX to a standard terminal emulation program on a Host PC.

An ANSI terminal or a PC running a terminal emulator is required to communicate with your Embedded PC. Simply connect an ANSI terminal (or emulator) to COM1 (DB9 female connector) using a null modem cable (this is included in the TS-ARM Development Kit), using serial parameters of 115,200 baud, 8 data bits, no parity, no flow control, 1 stop bit, and make sure jumper **JP2** is installed. If you are running Linux, the minicom program works well, Windows users can run the Hyperterm application. Technologic Systems offers a null modem cable with both 25 pin and 9 pin connectors at each end as part number CB7-05. Some systems also require the 10-pin header to 9-pin Sub-D adapter

which is P/N: RC-DB9.

The console can be changed to COM2 by installing **JP4** (with JP2 also installed). If your application does not require a console or both COM ports are required, then removing the jumper **JP2** easily disables all console output.

Connect a regulated 5VDC, (1A minimum) power source using the included 2 screw terminal strip/connector. Please note the polarity printed on the board. The boot messages, by default, are all displayed on COM1 at 115200 baud.

## 2.3 Boot Sequence

---

The boot sequence has four distinct stages:

1. TS-BOOTROM messages
2. RedBoot ROM monitor messages
3. Linux Kernel messages
4. Login prompts

Upon power up, the board executes proprietary Technologic Systems boot-code, TS-BOOTROM, then immediately executes RedBoot. RedBoot is a feature rich boot-ROM monitor, that allows manipulation of the on-board flash, JFFS2/YAFFS2 images, loading and execution of a kernel or executable from either tftp (trivial ftp), serial console, or from flash, and offers GDB debugging stubs.

If not interrupted by the user within one second, a pre-existing RedBoot script is executed, loading a default Linux kernel into memory from on-board flash. This will cause the pre-existing JFFS2/YAFFS2 file-system to boot.

One can view the RedBoot defaults for the board, as well as the default script, by entering at the RedBoot command prompt:

```
$ fconfig -l
```

The defaults can be changed by simply entering “fconfig” at the RedBoot prompt and answering the prompts. A final chance to write or discard the changes to the board will be given by RedBoot.

The default script instructs RedBoot to load the Linux kernel from the flash, and instruct the Linux kernel to use the JFFS2/YAFFS2 image on the flash chip for its root file-system. The Linux kernel must be loaded into memory address **0x00218000**. Loading the kernel from flash is done automatically by RedBoot in the default script with the following command:

```
$ fis load vlinux
```

After loading the kernel, the default script then executes the kernel with the following command:

```
$ exec -c "console=ttyAM0,115200 root=/dev/mtdblock1"
```

After the TS-Kernel is loaded, The login prompt is displayed. Type “root” at the login prompt, without a password. A Bash shell login prompt will then appear. At this point, the **TS-7260** is executing the run-time Technologic Systems Linux kernel and accepting user commands.

## 2.4 Loading or Transferring Files

---

Three methods are available for transferring files between a desktop PC and your **TS-7260**: Ethernet downloads, flash memory devices, and Zmodem downloads. Full descriptions of each are detailed below. Other programs that use serial ports to transfer should work as well.

### *Transferring Files via the Ethernet Port*

The default JFFS Linux root file system includes a small FTP server that can be used for uploading/downloading of files across an Ethernet network. Simply point your preferred FTP client to your **TS-7260** IP address (default is 192.168.0.50). You can login as root or any valid user previously created from the useradd utility. By default, the JFFS image will not accept anonymous FTP.

### *Transferring Files via Flash Memory Device*

The **TS-7260** removable Compact Flash card, an SD card or an USB flash memory card can be used to easily move files from a host system. We suggest using a low-cost SanDisk USB Compact Flash card or SD card interface for your host system. USB memory devices need no extra accessory to connect to the host PC. The flash memory devices can then be hot swapped (inserted or removed without rebooting the host PC).

### *Zmodem Downloads*

Using the Zmodem protocol to send files to and from the **TS-7260** SBC is simple and straightforward. The only requirement is a terminal emulation program that supports Zmodem, and virtually all do. If you are using Windows 95 or later for your development work, the HyperTerminal accessory works well.

To download a file to the **TS-7260** from your host PC, execute lrz at the Linux command line on the **TS-7260** (while using console-redirectation from within your terminal emulator) and begin the transfer with your terminal emulator. In HyperTerminal, this is 'Send File...' from the 'Transfer' menu.

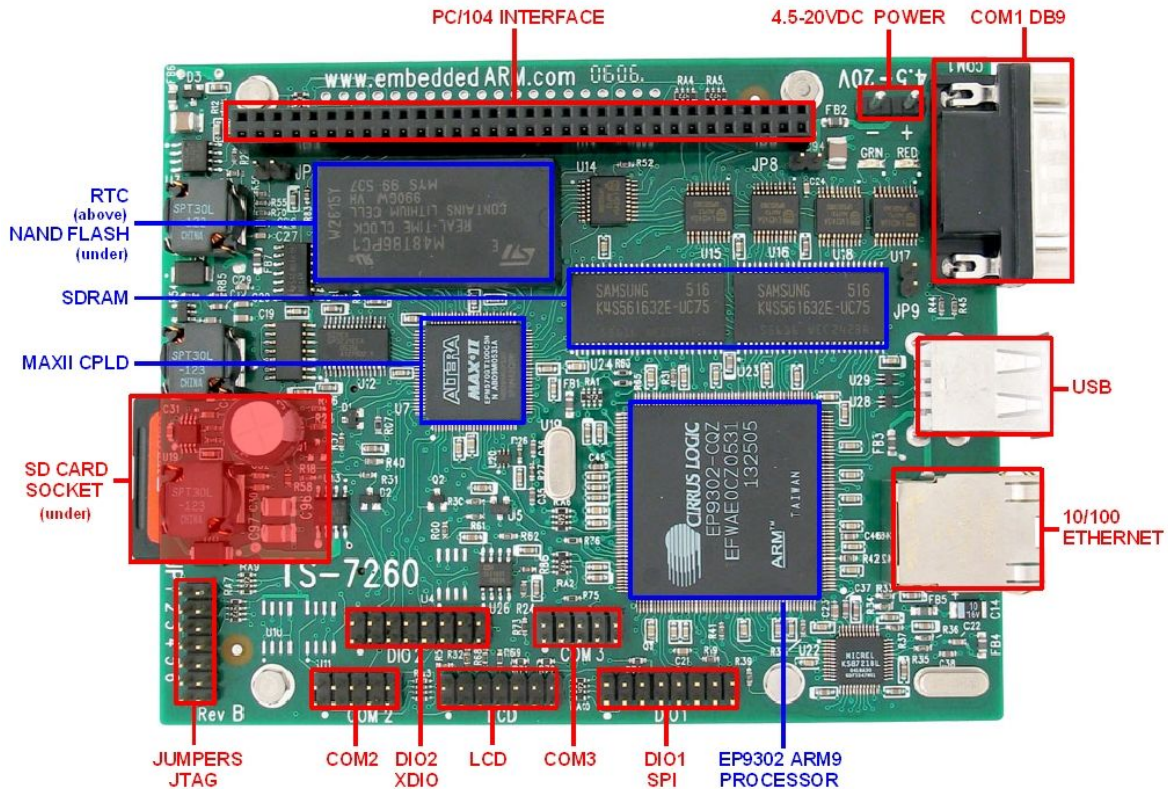
To upload a file from the **TS-7260** to your host PC, execute lsz <FILENAME> at the Linux command line on the **TS-7260** and start the transfer in your terminal emulator. Many emulators, HyperTerminal among them, will automatically begin the transfer themselves.

Occasionally there may be errors in transmission due to background operations. This is not a problem -- Zmodem uses very accurate CRC checks to detect errors and simply resends bad data. Once the file transfer is complete the file is completely error free. For best results when using HyperTerminal, the hardware handshaking must be enabled in HyperTerminal.

### 3 HARDWARE COMPONENTS

The following picture shows where the main headers, connectors and most important hardware components are located on the **TS-7260**. Understanding this picture will help you to follow the header-connector oriented organization of this manual. The blue marked objects on the picture are the on-board chips and components, while the red ones are the various on-board headers and connectors for peripherals.

**Picture: TS-7260 Hardware Components**



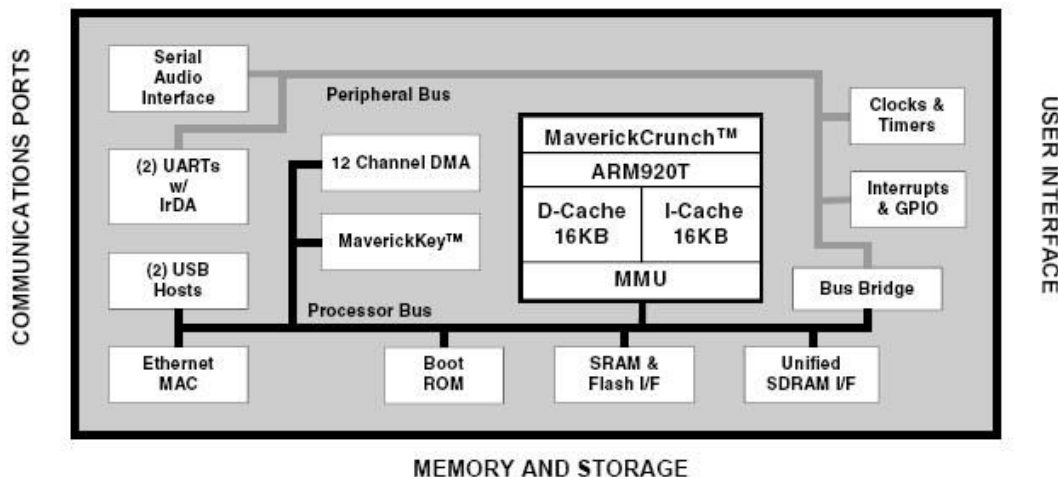
#### 3.1 Processor

##### *Cirrus EP9302*

The EP9302 features an advanced 200 MHz ARM920T processor design with a memory management unit (MMU) that allows support for high-level operating systems such as Linux, Windows CE, and other embedded operating systems. The ARM core operates from a 1.8 V supply, while the I/O operates at 3.3 V with power usage between 100 mW and 750 mW (dependent on speed). As a general-purpose processor, it provides a standard set of peripherals on board and a full set of Technologic Systems add-on peripherals via the standard PC/104 Bus.

The ARM920T's 32-bit architecture, with a five-stage pipeline, consisting of fetch, decode, execute, memory, and write stages, delivers very impressive performance at very low power. The EP9302 CPU has a 16 KB instruction cache and a 16 KB data cache to provide zero-cycle latency to the current program and data, or they can be locked to guarantee no-latency access to critical sections of instructions and data. For applications with instruction-memory size restrictions, the ARM920T's compressed Thumb instruction set can be used to provide higher code density and lower Flash storage requirements.

Picture: Cirrus EP9202 Block Diagram



EP9302 key features include:

- ✓ ARM (32-bit) and Thumb (16-bit compressed) instruction sets
- ✓ 32-bit Advanced Micro-Controller Bus Architecture (AMBA)
- ✓ 16 kbyte Instruction Cache with lockdown
- ✓ 16 kbyte Data Cache (programmable write-through or write-back) with lockdown
- ✓ MMU for Linux®, Microsoft® Windows® CE and other operating systems
- ✓ Translation Look Aside Buffers with 64 Data and 64 Instruction Entries
- ✓ Programmable Page Sizes of 1 Mbyte, 64 kbyte, 4 kbyte, and 1 kbyte
- ✓ Independent lockdown of TLB Entries

For further information about the EP9302 features, refer to the [EP9301 User's Guide](#).



**Note**

The EP9302 is identical silicon to the EP9301 except it is rated to run at 200 Mhz, instead of 166 Mhz. The available [EP9301 User's Guide](#) can still be used as the main reference manual.

**MMU**

The EP9301 features a Memory Management Unit, enabling high level operating systems such as Embedded Linux and Windows CE to run on the **TS-7260**. In the same way, the Linux TS-Kernel takes advantage of the MMU functionality.

The MMU is controlled by page tables stored in system memory and is responsible for virtual address to physical address translation, memory protection through access permissions and domains, MMU cache and write buffer access. In doing so, software applications can access larger "virtual" memory space than the available physical memory size, allowing multiple programs to run and use the system memory simultaneously.

For further information about the MMU functionalities, refer to the EP9301 User's Guide.

**Interrupts**

The EP9302 interrupt controller allows up to 54 interrupts to generate an Interrupt Request (IRQ) or Fast Interrupt Request (FIQ) signal to the processor core. Thirty-two hardware priority assignments are provided for assisting IRQ vectoring, and two levels are provided for FIQ vectoring. This allows time critical interrupts to be processed in the shortest time possible.



Internal interrupts may be programmed as active high or active low level sensitive inputs. GPIO pins programmed as interrupts may be programmed as active high level sensitive, active low level sensitive, rising edge triggered, falling edge triggered, or combined rising/falling edge triggered.

The EP9302 interrupt controller also includes the following features:

- ✓ Supports 54 interrupts from a variety of sources (such as UARTs, GPIO and ADC)
- ✓ Routes interrupt sources to either the ARM920T's IRQ or FIQ (Fast IRQ) inputs
- ✓ Three dedicated off-chip interrupt lines operate as active high level sensitive interrupts
- ✓ Any of the 19 GPIO lines maybe configured to generate interrupts
- ✓ Software supported priority mask for all FIQs and IRQs

**Note**

For peripheral driver development purpose, notice that the external IRQ lines 5,6 and 7, which are ISA/X86 architecture based, are mapped to EP9302 external interrupt lines 22, 33 and 40, respectively. For further information about interrupts, including the EP9302 interrupt controller and map, refer to the [EP9301 User's Guide](#), chapter 5.

### 3.2 Memory

**TS-7260** uses three type of memory. The SDRAM is the fast access volatile memory used to run applications by the processor and the on-board flash is the non-volatile memory used for storage purpose. Flash memory may also be added using USB memory drivers.

#### ***On-Board SDRAM***

The **TS-7260** uses 32 MB SDRAM technology to provide 32 or 64 MB of high-speed volatile memory. The memory is soldered directly to the board, making the **TS-7260** more reliable in high-vibration environments.

The **TS-7260's** RAM is not contiguous in the physical memory map of the EP9302. But the MMU is programmed to remap the blocks of RAM to appear as a contiguous block of memory at the very beginning of the virtual memory map. In the case of a 256 Megabit SDRAM chip (32 MB), it is located at 0 through 32 MB in the virtual memory map.

Refer to the MMU section of this manual to understand how the physical memory is mapped and the virtual memory is translated.

**Note**

It is possible to use larger sizes of the SDRAM chip than the standard 32 MB one. The **TS-7260** is designed to accommodate both 32 MB and 64 MB chips, providing up to 128 MB of RAM memory. Contact Technologic Systems for larger SDRAM sizes.

#### ***Battery Backed SRAM***

There is a peripheral board available for the **TS-7260** named TS-NVRAM that adds 32K bytes or 128 Kbytes or 512K bytes of battery-backed SRAM. Battery backed SRAM provides non-volatile memory with very fast write times and unlimited write cycles, unlike Flash memory. This can be very important if the data is constantly being updated several times per minute, since Flash devices can wear-out after a few million write cycles. It also eliminates the latency that Flash memory has during write cycles, since Flash technology write cycles are about 10-100 times slower than read cycles.

The TS-NVRAM peripheral board is located at the PC/104 memory space base address of 0x11AA\_0000. This resource is a byte-wide memory device using a lithium battery that is guaranteed to last a minimum of 10 years with or without power applied.

### **On-Board NAND Flash**

The **TS-7260** uses a NAND Flash chip for its on-board Flash resource. The physical address of the Flash chip is **0x6000\_0000**. The first 16KB is reserved for the TS-BOOTROM code. The TS-BOOTROM code initializes various internal configuration registers for proper operation, and initializes and tests the SDRAM. The last 3 MB are reserved for the RedBoot ROM monitor, RedBoot FIS (Flash Image System) and RedBoot FCONFIG (Flash configuration).

The Linux kernel shipped by default is pre-loaded in the FIS and the default boot script and Ethernet MAC address are contained in the FCONFIG. You may also use the RedBoot FIS to store and load images that contain eCos applications or other OS/RTOS boot loaders.

The rest of the on-board flash is used for the Linux YAFFS2 file system. This is a journaling file system that is aware of the wear-out mechanism of the NAND flash and incorporates ECC algorithms at the file system level to maximize Flash lifetime. It is also extremely tolerant of power failures during file write sequences.

The entire Flash chip can be write-protected by removing Jumper 3. When JP3 is not installed, the Flash chip becomes a read-only resource.



#### **Note**

It is possible to use larger sizes of the NAND Flash than the standard 32 MB chip. The **TS-7260** is designed to accommodate both 32 MB and 128 MB chips, providing up to 256 MB of on-board flash. Contact Technologic Systems for larger Flash sizes.



#### **Note**

The YAFFS1 file system runs on the **TS-7260** boards that feature NAND chips with 512 byte page size, enabling up to 128 MB of on-board flash. The YAFFS2 file system supports the new NAND technology, with 2k page size, hence it will be installed on **TS-7260** boards that are configured with 128-256 MB of flash.

### **USB Flash Drive or Compact Flash Card**

Additional non-volatile storage may be added with a USB flash drive or a Compact Flash card. These devices supply additional non-volatile storage either for data or for a complete operation system distribution, such as Debian. A tar-file of Debian is available on the Technologic Systems website. Alternatively, the developer's kit includes a USB flash thumb-drive or Compact Flash card pre-loaded with Debian.

Flash memory provided by these devices behaves much as a hard drive does with sizes ranging from 32MB to 1GB. These products are inherently more rugged than a hard drive since they are completely solid-state with no moving parts. However, they have the added advantage of being removable media

Use of a Compact Flash card with TS-7250 SBC or higher requires a USB Compact flash adapter, which will also be included in the TS-ARM Development Kit if requested. The USB flash drive has the advantage over a CF card in that the USB drive can be hot swapped.



#### **Note**

Drivers are available in the TS-Linux distribution to support USB flash drives. One can load Debian OS with two scripts provided by the on-board flash TS-Linux file system. First, invoke `/usr/bin/loadUSBModules.sh`, then run the script `/usr/bin/loadUSB.sh` to chroot into the Debian OS.

### SD Memory Card

Technologic Systems has a full license for using the additional SD features which are reserved for members of the SD Card Association. This has allowed us to design both the hardware logic core and software specifically tuned to the capabilities of the **TS-7260** CPU using the official SD specification documents. Since both a Linux driver module and an ARM9 object file containing OS-independent access routines are provided to customers purchasing the board hardware, customers do not have to seek SD licensing themselves.

SD Memory Card technology provides large capacity and fast access combined with a compact and slim profile, making it very appealing for a wide range of next generation products and applications. In addition, SD Cards feature content protection, planned capacity growth, high-speed data transfer, and a write protect switch. These devices supply additional non-volatile storage either for data or for a complete operation system distribution, such as Debian, to be used with the **TS-7260** SBC.

The Technologic System SD Card core is a very small implementation and can be integrated on the **TS-7260** CPLD and/or the TS-7300 FPGA. Four 8-bit registers are available for the software layer to control the SD Card hardware:

**Table: SD Card core registers**

I/O Addr	Name	Description
BASE + 0	SDCMD	SD Command register
BASE + 2	SDDAT	SD Data register
BASE + 4	SDSTATE	SD State register
BASE + 6	SDCTRL	SD Control register

### 3.3 Glue Logic CPLD

The TS-72XX ARM SBC's include a CPLD (a Xilinx 9572 on TS-7200 and TS-7250, or an Altera MAXII on TS-7260) which is responsible for taking control over the internal components communication through glue logic implementation. For instance, the CPLD is used to control the NAND flash through internal registers configuration.

The CPLD handles control signals on the PC104 bus, has a watchdog timer, enables jumper settings reading, handles the reset button, interfaces to the real-time clock and controls the EEPROM chip select. It also implements peripheral features that, together with EP9302 modules, makes available an advanced set of communication ports, DIO pins, ADC converters, and others.

The inclusion of a CPLD on the SBC allows customized programming for customers with special needs, without having to do a more expensive board redesign. For example, the MAXII CPLD on the TS-7260 can be configured with three different cores:

- ✓ 2TTLCOM option: 2 extra TTL-only serial ports with TX enable signals and that includes a very simple GPIO core (data direction register and data register only).
- ✓ TS-XDIO option: uber-GPIO that can do quadrature, PWM, freq-counter, pulse timing, IRQ and DRQ, etc
- ✓ SDSOCKET option: a special core for a SD interface that requires a special Linux driver module to be of use.

The CPLD can be programmed using the JTAG header and special software/hardware supporting tools. Contact Technologic Systems for support on CPLD programming software and tools.

### 3.4 Real-Time Clock

The **TS-7260** optionally supports a Non-volatile Battery-backed real-time clock (RTC) which is soldered onto the board. This option uses an ST Micro M48T86PC1 module for the real-time clock function. This module contains the lithium battery, 32.768 kHz crystal, and a RTC chip with 114 bytes of battery-backed CMOS RAM. It will maintain clock operation for a minimum of 10 years in the absence of power.

The 114 bytes of non-volatile RAM, physically located in the RTC chip, are available to the user. Contact Technologic Systems for driver support.

The RTC is accessed using two registers. The write-only index register is located at physical address location **0x1080\_0000** and the RTC data register is location at physical address location **0x1170\_0000**. These are byte-wide registers with the Index Register property of write only. The Data Register has a read/write property. Valid Index Register values are between 0 and 127, decimal. The first 14 index locations are used for accessing the RTC Time and Date registers. The next 114 locations are non-volatile RAM locations.

This option is NOT compatible with the TS-5620, a peripheral board that also uses an ST Micro RTC module for real-time clock functionality. While the two options are mutually exclusive, it is possible to use the TS-5620 peripheral board on a **TS-7260** that does not have the on-board RTC option installed. Any source code that utilizes the RTC is compatible with both optional installations. The TS-Kernel shipped with the boards includes support for the TS-5620 peripheral board.

### 3.5 Watchdog Timer

The **TS-7260** incorporates a Watchdog Timer (WDT) unit. The WDT can be used to prevent a system “hanging” due to a software failure. The WDT causes a full system reset when the WDT times out, allowing a guaranteed recovery time from a software error. To prevent a WDT timeout, the application must periodically “feed” the WDT by writing a specific value to a specific memory location.

**Table: Watchdog Control Registers**

Register	Address	Access
WDT Control register	0x2380_0000	Read/Write
WDT Feed register	0x23C0_0000	Write Only

The WDT Control register must be initialized with the timeout period desired. This may be as short as 250 mS or may be as long as 8 seconds. After the WDT has been enabled, the WDT counter begins. The application software can reset this counter at any time by “feeding” the WDT. If the WDT counter reaches the timeout period, then a full system reset occurs.

**Table: Watchdog Timeout Register**

Value	MSB	MID	LSB	Timeout Period
0x00	0	0	0	Watchdog Disabled
0x01	0	0	1	250 mS
0x02	0	1	0	500 mS
0x03	0	1	1	1 second
0x04	1	0	0	-- Reserved
0x05	1	0	1	2 seconds
0x06	1	1	0	4 seconds
0x07	1	1	1	8 seconds

In order to load the WDT Control register, the WDT must first be “fed”, and then within 30 uS, the WDT control register must be written. Writes to this register without first doing a “WDT feed”, have no affect.

In order to clear the WDT counter (feeding the watchdog), a value of Hex 05 must be

written to the WDT Feed register.

By default, a user process does not have the physical address space (access) of the watchdog registers mapped. When using the Linux OS, the watchdog can be reached from user C code by using the `mmap()` system call on the `/dev/mem` special file to map the areas of physical address space into process user address space.

**!** **Warning**  
Use only the Watchdog Timer implemented by Technologic Systems in the CPLD. The Watchdog Timer included in the EP9302 has serious problems.

## 4 COMMON INTERFACES GENERAL INFORMATION

The purpose of this section is to provide general information about the common interfaces, such as Serial Ports and Digital Input/Output, which appear in more than one header or connector of the **TS-7260**. For further information on these features, refer to the Connectors and Headers section of this manual.

### 4.1 Serial Ports

The **TS-7260** have two asynchronous serial ports (COM1 and COM2) which provide a means to communicate with external serial devices. Each is independently configured as a 16C550- type COM port that is functionally similar to a standard PC COM port. These ports have 16-byte FIFOs in both the receive and the transmit UART channels. Both COM ports can support all standard baud rates up through 230.4Kbaud. Both COM ports may be configured to use a DMA channel (useful when very high baud rates are being used).

COM1 and COM2 UARTs can generate:

- ✓ Four individually maskable interrupts from the receive, transmit, and modem status logic blocks
- ✓ A single, combined interrupt that is asserted if any of the individual interrupts are asserted and unmasked

The COM1 port can also support the HDLC protocol. Refer to the Cirrus EP9301 User's Guide for more details. The COM2 port can optionally support RS-485 half or full duplex levels.

The **TS-7260** has either three or five total asynchronous serial ports and the capability to add more via the PC/104 Expansion Bus. Three serial ports are standard on-board and an additional two ports on-board may be provided when the OP-2TTLCOM option is installed. (Should this option be included, the TS-XDIO capabilities are not available). Each is independently configured and is functionally similar to a PC COM port. The characteristics of each serial port differ and are noted in the following table:

**Table: TS-7260's Serial Ports Characteristics**

	On-board by Default			On-board Optional		PC/104 Port(1-4)
	COM1	COM2	COM3	COM4	COM5	
<b>Maximum Baud Rate</b>	230.4Kbps	230.4Kbps	115.2Kbps	115.2Kbps	115.2Kbps	1000Kbps
<b>Minimum Baud Rate</b>	110bps	110bps	2400bps	2400bps	2400bps	no lower limit
<b>Data bits</b>	7, 8	7, 8	7, 8	8	8	7, 8, 9
<b>Parity</b>	Yes	Yes	Yes (Data Bits=7)	Yes (Data Bits=7)	Yes (Data Bits=7)	Yes
<b>RTS/CTS</b>	Yes	No	Yes	No	No	Yes
<b>Signal Level</b>	RS232	RS232	RS232 (TTL available if OP-2TTLCOM)	TTL	TTL	RS232
<b>FIFO</b>	16 bytes	16 bytes	5 bytes (RX only)	5 bytes (RX only)	5 bytes (RX only)	16 bytes
<b>RS-485</b>	No	Optional	No	No	No	Optional
<b>Connector</b>	DB9	10-pin Header	10-pin Header	DIO2 Header	DIO2 Header	10-pin Header

## 4.2 Digital I/O

There are 30 Digital Input/Output (DIO) lines available on the **TS-7260**. These are available on three headers labeled “DIO1,” “DIO2,” and “LCD”, which may also assume another primary function, like the serial port option on DIO2 header. In this case, if the primary function associated with the header pin is not required by the user application, the signal is then available as a general purpose digital IO.

In the following table, signal quantities shown in bold are those that are either dedicated or possibly available as general purpose Digital IO. The header labeled LCD can be used as 11 DIO lines or as an alphanumeric LCD interface. The header labeled DIO1 has 9 DIO pins available. In addition to the DIO signals, each header also has a power pin and Ground available. The LCD Header has 5V power available, the DIO1 Header has 3.3V power, and the DIO2 Header provides a ground signal.

**Table: Digital IO Signal Availability on the TS-7260**

	<b>DIO1 Header</b>	<b>DIO2 Header</b>	<b>LCD Header</b>
<b>Total Header Pins</b>	16	16	14
<b>Shared Primary Function</b>	0	10	11
<b>Digital IO Dedicated</b>	9	0	0

The DIO2 header is a 16 pin header with 10 shared multi-function pins. These pins can be XDIO, standard DIO, SD card interface (signals are run down to the SD card connector on the bottom of the **TS-7260**), or can have extra COM ports present on them. See the DIO2 section below for more information on configuring and using the DIO2 pins.

Three pins on the DIO header are used to bring out the EP9302 SPI bus. By using some of the DIO pins as peripheral Chip Select signals, a complete interface is available for SPI peripherals. It is also possible to bring out a fourth SPI bus function [SPI\_Frame] by adding a 10 ohm resistor in the position labeled R1 on **TS-7260** boards. This signal is not required for many SPI peripherals but it may prove useful in some applications.

All of the DIO lines are programmable as either inputs or outputs and the direction of each I/O pin can be individually programmed. All DIO control registers are 8-bits wide and aligned on word (32-bit) boundaries. For all registers, the upper 24 bits are not modified when written and are always read back as zeros. Every DIO pin has two registers used to access it, an 8-bit data register and an 8-bit data direction register (DDR). The DDR controls whether each DIO pin is an input or an output (“1” = output). Writing to the data register only affects pins that are configured as outputs. Reading the data register always returns the state of the DIO pin.

Many of the DIO pins (DIO\_0 thru DIO\_8 and LCD\_0 thru LCD\_7) can also be programmed to cause interrupts. The interrupt properties of these pins are individually configurable. Each interrupt can be either high or low level sensitive or either rising or falling edge triggered. It is also possible to enable debouncing on any of these interrupts. Debouncing is implemented using a 2-bit shift register clocked by a 128 Hz clock. Refer to the **EP9301 User's Guide** and see Chapter 21 for details

The 5V power output pin on the LCD Header is current limited to 1000mA to prevent excessive current. The 3.3V power on the DIO1 Header is current limited to approximately 1 Amp by a 3.3V regulator.

### 4.3 A/D Converters

---

The EP9302 A/D converter is standard on all **TS-7260** boards. The Cirrus EP9302 features a 5 channel, 12-bit Analog to Digital Converter with an analog multiplexor, with an input range of 0 to 3.3 V.

On the **TS-7200**, only the top and bottom channels (ADC0 and ADC4) are brought out on the DIO1 Header. On the **TS-7250**, the dedicated A/D Header brings out all the 5 EP9302 A/D channels. The **TS-7260** uses 3 of the 5 EP9302 A/D channels for on-board measurements, therefore only 2 channels are available on the DIO2 Header.

The **TS-7200** and **TS-7250** support an optional 8 channel, 12-bit A/D converter (ADC) with a conversion time of 12  $\mu$ S through the MAX 197 chip. This will allow up to 60,000 samples per second. The dedicated MAX197 A/D Header brings out all the 8 A/D channels.



## 5 CONNECTORS AND HEADERS

---

### 5.1 10/100 Base-T Ethernet Connector

---

The EP9302 Ethernet LAN controller incorporates all the logic needed to interface directly to any MII compatible Ethernet PHY chip. A low-power Micrel KS8721 chip is used to implement the Ethernet PHY function and an integrated RJ-45 connector with built-in 10/100 transformer and LED indicators completes the Ethernet sub-system.

The **TS-7260** has both a LINK/ACTIVITY LED and a 10/100 speed LED built into each RJ-45 connector that indicates the current Ethernet status. The LINK LED (left side of connector, green) is active when a valid Ethernet link is detected. This LED should be ON whenever the **TS-7260** is powered and properly connected to a 10/100BaseT Ethernet network. The LINK/ACTIVITY LED will blink to indicate network activity for either inbound or outbound data. The SPEED LED (right side of connector, amber) will be on when a 100Mb network is detected and off for a 10Mb network. Both of these LEDs are controlled by the KS8721 and do not require any overhead by the processor.

The Ethernet PHY chip can be powered down, under software control, to save approximately 90 mA of current consumption. This is controlled by the EP9302 Digital output on Port H, bit 2. A logic zero will power down the KS8721 PHY interface.

**Note**

TS-Linux provides all the software support to use the EP9302 10/100 Ethernet core. For more details, find the TCP/IP configuration instructions on the TS-Linux documentation.

### 5.2 USB Connector

---

The USB Connector on the **TS-7260** provide two USB interfaces for the user. These are directly connected to the EP9302 processor, which integrates an USB dual-port Open Host Controller Interface (Open HCI), providing full-speed serial communications ports at a baud rate of 12 Mbits/sec. Up to 127 USB devices (printer, mouse, camera, keyboard, etc.) and USB hubs can be connected to the USB host in the USB “tiered-star” topology. This includes the following features:

- ✓ USB 2.0 compatible
- ✓ Open HCI Rev 1.0 compliant
- ✓ USB device connections support at both low-speed (1.5 Mbps) and full-speed (12 Mbps)
- ✓ Root HUB integrated with 2 downstream USB ports
- ✓ Transceiver buffers integrated, over-current protection on ports
- ✓ Supports power management
- ✓ Operates as a master on the bus

**Note**

TS-Kernel implements all the necessary driver support to enable the USB OHCI. Also, a wide variety of USB drivers for devices such as mouse, keyboard and flash memory are available. Refer to the Linux for TS-ARM User's Guide or contact us for further information on how to integrate an USB device and an USB Linux driver with your **TS-7260**.

### 5.3 SD Card Connector – CPLD

The SD Card socket (ALPS connector) at the back side of the **TS-7260** enables SD Cards to be plugged to the SBC. The hardware core implemented by Technologic Systems is integrated inside the on-board CPLD. Technologic Systems has written a binary Linux driver module and a set of generic, OS-independent read/write routines for accessing the SD flash inside of an ARM object (.o) file. The format of the SD card must be in EXT2 format for proper operation with Linux as a root file system.


**Note**

On the **TS-7300**, the SD Card is standard bootable, but it is not on the **TS-7260** by default. If you either need a bootable SD Card socket on the **TS-7260** or don't want the NAND flash support on **TS-7260**, contact Technologic Systems.

### 5.4 COM1 Connector

The COM1 RS-232 port uses a DB-9 male connector on the **TS-7260**.

**Table: COM1 DB9 Pin-Outs**

DB9 Pin	Signal	Description
<b>1</b>	DCD	Data Carrier Detect
<b>2</b>	RXD	Receive Data
<b>3</b>	TXD	Transmit Data
<b>4</b>	DTR	Data Terminal Ready
<b>5</b>	GND	Ground
<b>6</b>	DSR	Data Set Ready
<b>7</b>	RTS	Request to Send
<b>8</b>	CTS	Clear to Send
<b>9</b>	--	--

The base address of COM1 appears in the physical address space at **0x808C\_0000**. Technologic Systems provides software drivers to access this port. This full complement of RS-232 signals (all except Ring Detect) allows COM1 to interface to almost any serial RS-232 device.

## 5.5 COM2 Header

The base address of COM2 appears in the physical address space at **0x808D\_0000**. Technologic Systems provides software drivers to access this port. COM2 has RS-232 support only for the TXD and RXD signals. This is sufficient to interface with serial devices that only require transmit and receive data lines. COM2 is accessible on a 10-pin header labeled COM2.

**Table: TS-7260 COM2 Serial Port Header Pin Out**

TX-	NC	NC	RX-	NC
<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
TX+	RXD	TXD	RX+	GND



### Note

The COM2 serial port header uses a non-standard numbering scheme. This was done so the header pins would have the same numbering as the corresponding DB-9 pin; i.e. pin 3 (TXD) on the header connects to pin 3 on the DB-9.

**Table: COM2 Serial Port Pin Configuration of Connectors**

COM2 Header	DB9 Connector	RS-232	Half Duplex RS-485	Full Duplex RS-485
<b>1</b>	<b>1</b>		TX+ / RX+	TX+
<b>2</b>	<b>2</b>	RXD		
<b>3</b>	<b>3</b>	TXD		
<b>4</b>	<b>7</b>			RX+
<b>5</b>	<b>5</b>	GND	GND	GND
<b>6</b>	<b>6</b>		TX- / RX-	TX-
<b>7</b>	<b>7</b>			
<b>8</b>	<b>8</b>			
<b>9</b>	<b>9</b>			RX-
<b>10</b>	-	-	-	-

Adapter cables are available to convert COM2 to a standard 9-pin SubD male connector. In addition, COM2 has optional support for full or half-duplex RS-485



### Note

The COM2 serial port 10-pin header has both RS-232 and RS-485 signals. Only one set may be used for a given application – it is not possible to use both at the same time.

### Optional RS-485 Support

The COM2 Port supports RS-485 half duplex or full duplex operation. RS-485 drivers allow communications between multiple nodes up to 4000 feet (1200 meters) via twisted pair cable. Half-duplex RS-485 requires one twisted pair plus a Ground connection, while full duplex requires two twisted pair plus a Ground. For half-duplex operation, a single twisted pair is used for transmitting and receiving.

The `ioctl` Linux system call is used to initialize the automatic RS-485 control logic. For full-duplex operation, two twisted pairs are used and the transmit driver is typically enabled all the time. Fail-safe bias resistors are used to bias the TX+, TX- and RX+, RX- lines to the correct state when these lines are not being actively driven. In a typical RS-485 installation, the RS-485 drivers are frequently “tri-stated.” If fail-safe bias resistors are not present, the RS-485 bus may “float” and small amounts of noise can cause spurious characters at the receivers. 4.7K Ohm resistors are used to pull the TX+ and RX+ signals to 5V and also to bias the TX- and RX- signals to ground.

Termination resistors in many installations are not required when cable lengths are less than 50 meters and the baud rate is 9600 or less. Termination resistors may be required for reliable operation when running longer distances or at higher baud rates. Termination resistors (typically 100-150 ohms) should only be installed at each end of an RS-485 transmission line. In a multi-drop application where there are several drivers and/or receivers attached, only the devices at each end of the transmission line pair should have termination resistors.

A read at I/O location **0x2240\_0000** bit 1 will return a “1” when the RS-485 option is installed.

**Note**

A special `ioctl` command has been added to the TS-Kernel's serial code to turn off and on the automatic RS-485 feature. Refer to the Linux for TS-ARM User's Guide for additional software information on RS-485 support.

***RS-485 Quick-Start Procedure***

Perform the following steps:

1. Check that the RS-485 option is installed in the **TS-7260** (required).
2. Attach the RS-485 cable to the 10-pin header. (Adapter cables to SubDB9 connector are available from Technologic Systems).
3. Set the COM2 UART serial parameters (baud rate, data, parity, and stop bits, interrupts, etc) in your source code.
4. Use the `TIOC_SBC's485 ioctl()` to enable the automatic RS-485 feature.
5. Simply `read()` and `write()` data to the COM2 UART – the automatic RS-485 logic controls TX enable.

***Automatic RS-485 TX Enable***

The **TS-7260** supports fully automatic TX enable control under hardware control. This simplifies the design of half-duplex systems since turning off the transmitter and enabling the receiver in software is typically difficult to implement. If this is implemented in software, then the COM2 UART transmit holding register and the transmit shift register both must be polled until empty before deasserting the TX enable signal to the RS-485 driver. The design gets more complicated when using the TX FIFO or when using a multi-tasking OS such as Linux.

The implementation on the **TS-7260** uses hardware to automatically control when the RS-485 drivers and receivers are selected. The automatic RS-485 logic eliminates the need for any software polling and supervision. This circuit automatically turns on/off the RS-485 transceiver and receiver at the correct times. This only requires the COM2 Mode registers to be initialized once based on baud rate and data format.

**Table: COM2 Mode Register**

Value	MSB	MID	LSB	Mode
0x00	0	0	0	RS-232
0x01	0	0	1	Full-Duplex RS-485
0x02	0	1	0	-- Reserved
0x03	0	1	1	-- Reserved
0x04	1	0	0	Half-Duplex 9600 baud
0x05	1	0	1	Half-Duplex 19.2Kbaud
0x06	1	1	0	Half-Duplex 57.6Kbaud
0x07	1	1	1	Half-Duplex 115.2Kbaud

COM2 will default to RS-232 mode at reset (COM2 mode register = Hex 00). In order to switch COM2 into Full duplex RS-485 mode, the COM2 Mode register at **0x22C0\_0000** must be set to Hex 01. In this mode, the TX drivers are always asserted.

In order to switch COM2 into half-duplex RS-485, then the COM2 mode register at 0x22C0\_0000 must be set to Hex 04, 05, 06 or 07 depending upon the baud rate required. A second COM2 Mode register at **0x2300\_0000** must be set to Hex 01 if the COM2 is in half-duplex mode and the data format is using 8 data bits with parity or 8 data bits with 2 stop bits.


**Note**

COM2 in half-duplex RS-485 supports baud rates of 9600, 19.2K, 57.6K and 115.2K. If you need to operate at any other baud rate, then please contact Technologic Systems

## 5.6 COM3 Header

In the default **TS-7260** configuration (OP-2TTLCOM not selected), the COM3 port has RS-232 signal levels, utilizes RTS and CTS signals and is accessible on a 10-pin header labeled COM3:

**Table: TS-7260 COM3 Serial Port Header Pin Out**

NC	RTS	CTS	NC	NC
<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
NC	RXD	TXD	NC	GND

The COM3 serial port is implemented at address **0x12400000**. It has a 5-byte receive FIFO and a 1-byte transmit FIFO. A TS-Kernel serial driver, "tsuart" is available which allows manipulation of this hardware in Linux using the device node /dev/ttyTS0. The 5-byte receive FIFO is enough to prevent buffer overflow at 115200 baud with the Linux driver; however, it is recommended that user designs first utilize using COM1 or COM2 ports with 16-byte FIFOs at these high speeds

**Table: COM3 Registers Description**

Addr	Register	Bits Description
0x0	STAT register	Bit 7-5: MODE, baud rate (RW) Bit 4: RTS, UART request-to-send (RW) Bit 3: CTS, UART clear-to-send (RO) Bit 2: OERR, Overflow error (RO) Bit 1: DR, Receive data ready (RO) Bit 0: TBRE, Transmit buffer empty (RO)
0x1	Read Address: RXDAT register	Bits represent RX data
0x2	Write Address: TXDAT register	Bits represent TX data

**Table: COM3 STAT-Mode Register – Bits 7-5**

Bit7	Bit6	Bit5	Mode
0	0	0	115200 8N1
0	0	1	57600 8N1
0	1	0	38400 8N1
0	1	1	19200 8N1
1	0	0	9600 8N1
1	0	1	4800 8N1
1	1	0	2400 8N1
1	1	1	UART off, IRQ disabled

The UART will interrupt the CPU while there is data in the RX fifo and on the rising edge of the TBRE bit. (full -> empty transition). The IRQ is cleared on read of the STAT register.

If the **TS-7260** is ordered with the OP-2TTLCOM Option, the COM3 port assumes TTL level signals and only the RXD and TXD signals will be used. It is sufficient to interface with serial devices that only require transmit and receive data lines. With the OP-2TTLCOM selected, the COM3 signals are available at the DIO2 header, as shown in the following table.

**Table: COM3 TTL on DIO2**

DIO2 Pin	Signal
1	COM3 TXD
3	COM3 RXD

## 5.7 DIO1 Header

The DIO1 Header provides +3.3V, GND, and 9 digital I/O lines that may be used to interface the **TS-7260** with a wide range of external devices. DIO lines DIO\_0 thru DIO\_7 are a byte-wide port using Port B on the EP9302 and are accessed via the data register at address location **0x8084\_0004**. The DDR for this port is at address location **0x8084\_0014**.

**Table: DIO1 Header Pin Configuration**

DIO1 Pin	Default Signal	TS-7200
1	DIO_0	
2	GND	
3	DIO_1	
4	Port_C0	ADC0
5	DIO_2	
6	SPI_Frame	ADC4
7	DIO_3	
8	DIO_8	
9	DIO_4	
10	SPI_MISO	
11	DIO_5	
12	SPI_MOSI	
13	DIO_6	
14	SPI_CLK	
15	DIO_7	
16	+3.3 V	

DIO\_8 is accessed via bit 1 of Port F in the EP9302. The Port F data register is at address location **0x8084\_0030**. The DDR address for this port is location **0x8084\_0034**.

The Pin 4 of the DIO1 Header, in the default configuration, is accessed via bit 0 of Port C in the EP9302. The address location **0x8084\_0008** is Port C Data Register and **0x8084\_0018** is Port C Directon Register.

When accessing these registers, it is important not to change the other bit positions in these Port F registers. Other DIO1 Port functionality, used for dedicated **TS-7260** functions, utilize these same control registers. All accesses to these registers should use read-modify-write cycles.

**Warning**  
 All pins on the DIO header use 0-3.3V logic levels. Do not drive these lines to 5V.

When the DIO pins are configured as outputs, they can “source” 4 mA or “sink” 8 mA and have logic swings between GND and 3.3V. When configured as inputs, they have standard TTL level thresholds and must not be driven below 0 Volts or above 3.3 Volts. DIO lines DIO\_0 thru DIO\_3 have 4.7K Ohm “pull-up” resistors to 3.3V biasing these signals to a logic “1”. The other DIO pins have 100K Ohm bias resistors biasing these inputs to a logic “1”.

**SPI Interface**

The EP9302 Synchronous Serial Port is available on the DIO1 header. This port can implement either a master or slave interface to peripheral devices that have either Motorola SPI, or National Semiconductor Microwire serial interfaces.

The transmit and receive data paths are buffered with internal FIFO memories allowing up to eight 16-bit values to be stored for both transmit and receive modes. The clock rate is programmable up to 3.7 MHz and has programmable phase and polarity. The data frame size is programmable from 4 to 16 bits.

By using some of the DIO1 Header pins as peripheral Chip Select signals, a complete interface is available for addressing up to 9 SPI peripherals. The SPI bus pins are defined in the table below:

**Table: DIO1 Header Pin-Outs for SPI**

DIO1 Pin	Name	Function
10	SPI_MISO	Master In/Slave Out
12	SPI_MOSI	Master Out/Slave In
14	SPI_CLK	Clock
6	SPI_Frame	SPI Frame pin

A fourth SPI bus function [SPI\_Frame] is available by adding a 10 ohm resistor in the position labeled R1 on **TS-7260**. This signal is not required for many SPI peripherals but it may prove useful in some applications.

**Warning**  
 The SPI bus pins use 0-3.3V logic levels. Do not drive these lines to 5V.

Refer to the [EP9301 User's Guide](#) and see Chapter 19 for more details on using SPI bus.

### Matrix Keypad

DIO signals DIO\_0 through DIO\_7 are physically arranged to allow a 16-pin (2x8) ribbon cable to directly connect a **4x4 matrix keypad**. Sample code is available for the Matrix Keypad. Contact Technologic Systems for further information.

## 5.8 DIO2 Header

The DIO2 Header provides +3.3V, GND, and 10 digital I/O lines that may be used to interface the **TS-7260** with a wide range of external devices.

When the DIO2 pins are configured as outputs, they can “source” 4 mA or “sink” 8 mA and have logic swings between GND and 3.3V. When configured as inputs, they have standard TTL level thresholds and must not be driven below 0 Volts or above 3.3 Volts.

**Table: DIO2 Header General Pin Configuration**

Pin	Signal	TS-7260 Primary Function	Alternate Signal	Alternate Function
1	XDIO_0	XDIO_0	TXD	COM3
2	GND			
3	XDIO_1	XDIO_1	RXD	COM3
4	Ext. Reset#			
5	XDIO_2	XDIO_2	TXD	COM4
6	Shutdown#			
7	XDIO_3	XDIO_3	RXD	COM4
8	ADC_2	ADC2		
9	XDIO_4	XDIO_4 – PWM/ One Shot Output – Quadrature Index Count	TXD	COM5
10	ADC_4	ADC4		
11	XDIO_5	XDIO_5 - Quadrature A	RXD	COM5
12	COM3_RTS	COM3_RTS		
13	XDIO_6	XDIO_6 - Quadrature B	TX Frame	COM4
14	GPIO	GPIO		
15	XDIO_7	XDIO_7 - Edge Counter/ Frequency/Quadrature	TX Frame	COM5
16	3.3 V			



When the additional 4th and 5th UART ports option is not selected, the DIO2 Header pins dedicated to COM4 and COM5 are GPIO pins, which implement basic digital input/output function. This means that a standard test cable can still be used to test the pins. There is a Digital IO capability implemented at **0x12C0\_0000**. The following register map describes the behavior of this feature:

**Table: Simple DIO feature on DIO2 Header Registers**

Addr	Register	Bits
<b>0x0</b>	Data Register	<b>1</b> - Pin at logic high <b>0</b> - logic low
<b>0x1</b>	Data Direction Register	<b>1</b> – output <b>0</b> - input

Bits X of each register corresponds to pin (X\*2+1) on the DIO2 header.

Pin 14 on the DIO2 header is another GPIO pin implemented in the CPLD. The data direction register for this GPIO pin is bit 1 and the data register for this GPIO pin is bit 2 at location **0x2140\_0000**.

**Warning**  
 All pins on the DIO2 header use 0-3.3V logic levels. Do not drive these lines to 5V.

### **Optional COM4 and COM5**

With the incorporation of Technologic Systems option OP-2TTLCOM, two additional on-board serial communication ports are available. However, the **TS-7260** standard TS-XDIO features are negated.

Note that the interfacing signal levels are TTL compatible, not RS-232. COM4 and COM5 TX Frame signals provide an envelope representing serial data transmitted on the respective port from the **TS-7260**.

If the board was built with the OP-2TTLCOM option, the 4th and 5th COM ports appear at addresses **0x1300\_0000** (COM4) and **0x1340\_0000** (COM5). They use the DIO2 header pins as shown below:

**Table: DIO2 COM4 and COM5**

DIO2 Pin	Signal
<b>5</b>	COM4 TXD
<b>7</b>	COM4 RXD
<b>9</b>	COM5 TXD
<b>11</b>	COM5 RXD
<b>13</b>	COM4 TX Frame
<b>15</b>	COM5 TX Frame

### **Optional SD-Card Socket**

When the **TS-7260** is built with the OP-SDSOCKET option, the DIO2 header signals will become the SD Card core signals. Therefore, both the TS-XDIO and the extra COM4/COM5 functions are lost. This is done because there is not enough physical space available in the on-board CPLD for the different cores to be programmed simultaneously.

### **Cirrus 5 Channel A/D Converter**

The Cirrus EP9302 features a 5 channel, 12-bit Analog to Digital Converter with an analog multiplexer, having an input range of 0 to 3.3 V. The Cirrus A/D converter can do a maximum of 925 samples per second, and requires a settling time of 2 milliseconds between channel switches.

On the **TS-7260**, three of the five available inputs are utilized to monitor board voltages, under software control. The remaining two channels are available for user signals. The monitored voltages are shown in the table below:

**Table: On-board Monitored Signals**

ADC Channel	On-board Signal	Channel Address
0	Power In	0x0000_0608
1	1.8 VDC Core Voltage	0x0000_0680
2	User Input 0 ADC #2	0x0000_0620
3	5.0 VDC Input Voltage	0x0000_0640
4	User Input 1 ADC #4	0x0000_0610

Software may access these monitored signals by executing an A/D conversion on the specified channel. The DIO2 Port pins are shown in the next table. They are connected to two of the EP9302 12-bit A/D converter inputs. These can be used to measure analog signals in the range of 0 to 3.3V.

**Table: DIO2 Port ADC Input Channels**

DIO2 Pin	ADC Input Channel
8	User Input 0
10	User Input 1

To maintain 12-bit accuracy, the analog signal being measured must have a low source impedance (less than 10 ohms). Otherwise, an operational amplifier may need to be added to buffer the A/D input. For detailed information, please see the Cirrus **EP9301 User's Guide**, page 518.

The following steps outline the software execution to use the Cirrus A/D converter:

1. Unlock the software lock before setting the TSEN bit in the ADCCLK register by writing 0xAA to the ADCSWLock register (0x8090\_00C0). "OR" in the TSEN bit (bit 31) to the ADCCLKDiv register (0x8093\_0090)
2. Unlock the software lock (again) before OR'ing in the ADCEN (ADC clock enable, bit 31) to 0x8093\_0080
3. Clear bit 2, the ADCPD (ADC Power Down) bit, at 0x8093\_0080. This bit MUST be set to 0 (see page 91 of the EP9301 User's Guide)
4. After unlocking the software lock, write the channel's magic value (see Cirrus EP9301 User's Guide, table 20-2) to the ADCSwitch register (0x8090\_0018) to select that channel for the next data acquisition
5. Poll the ADC Result register (0x8090\_0008) until bit 31 is not set
6. Using a 32 bit read operation, read the result from 0x8090\_0008, masking off the upper 16 bits

### **Interpreting Cirrus A/D Converter**

The Cirrus on-chip A/D converter is a successive approximation A/D converter. Each A/D channel is calibrated on the **TS-7260** and these 16-bit values are stored in non-volatile EEPROM. These calibration values minimize the offset errors and gain errors in the EP9302 A/D. It is important for the user program to use these values as per our sample code, which can be found either on our website or in the CD included in the Developer's Kit. Two reference points, 0 and 2.5 Volts, with the corresponding reference values stored in EEPROM. Bytes 0x07EB through 0x07FE of the EEPROM hold a two dimensional array:

[channel number][0V ref. point, 2.5V ref. Point]

The reference points are stored as a 16 bit value, and should be used to correlate the values returned by the Cirrus A/D converter to voltage.

**External Reset/Shutdown Control**

Two signals are available on the DIO2 Header to control the **TS-7260** Shutdown and Reset. These signals are TTL compatible and use negative logic. The reset process using the DIO2 Header causes a full-system reset, exactly the same as if the power were cycled.

**Table: TS-7260 External Control**

DIO2 Pin	Signal
4	Ext_Reset#
6	Shutdown#

**TS-XDIO**

The following is a description of the TS-XDIO port on the **TS-7260**. This core is designed to implement common controller tasks that are difficult, costly (CPU intensive), or impossible to accomplish in software with regular DIO / GPIO hardware and the facilities of the 200Mhz ARM processor. The TS-XDIO core appears at address locations **0x12c00000-0x12c00003** on the **TS-7260** boards and uses ARM IRQ 32 (IRQ 6). The 2 available clock sources are the 14.7456 Mhz fast clock (68ns of period) and the 32.7680 Khz slow clock (30518ns of period). The following is the pin-out for the TS-XDIO port on DIO2 header:

**Table: TS-7260 DIO2 Pin-out (TS-XDIO)**

GND	RST	SHD	ADC2	ADC4	RTS	GPIO	3.3V
<b>2</b>	<b>4</b>	<b>6</b>	<b>8</b>	<b>10</b>	<b>12</b>	<b>14</b>	<b>16</b>
<b>1</b>	<b>3</b>	<b>5</b>	<b>7</b>	<b>9</b>	<b>11</b>	<b>13</b>	<b>15</b>
XD_0	XD_1	XD_2	XD_3	XD_4	XD_5	XD_6	XD_7

TS-XDIO port features includes:

- ✓ 8-pins, individually programmable as schmitt-trigger inputs, outputs, or special functions (3.3V levels)
- ✓ glitch-detector (all pins) to catch edges/pulses as short as one fast clock period without continuous software polling
- ✓ edge-counter/frequency meter (pin 7) max incoming clock period of one-half the core clock frequency
- ✓ quadrature counter (pins 6,7) with max increment / decrement rate of one-half the maximum edge-counter frequency
- ✓ Pin 5 can optionally be used as an index pulse for resetting the edge counter or quadrature counter to 0
- ✓ PWM (pulse width modulation) output on pin 4 with period between two fast clock periods and 8190 slow clock periods, with 0.02% duty cycle precision
- ✓ One-shot programmable high or low output pulse (pin 5) of duration between one fast clock period and 4095 slow clock periods
- ✓ pulse / period timer (any pin) can measure high/low pulse time or full period lengths at fast clock period resolution

The TS-XDIO functionality can be controlled through 4 control/status registers that appear at physical memory. The register 0 is used to set the XDIO mode and general configuration:

**Table: TS-XDIO Register #0**

Bits	Function	Binary Values and Description
<b>Bits 7-6</b>	MODE (selects what registers 1-3 are)	<b>00</b> - pin direction, data register, IRQ/DRQ control <b>01</b> - edge / quadrature counter and glitch monitor <b>10</b> - input pulse timer and pin select <b>11</b> - PWM / pulse high / low-time set
<b>Bits 5-4</b>	Pulse timer mode and polarity	<b>00</b> - negedge to posedge (low pulse time) <b>01</b> - posedge to negedge (high pulse time) <b>10</b> - negedge to negedge (period time) <b>11</b> - posedge to posedge (period time)
<b>Bit 3</b>		pulse timer reset (W) or waiting for trailing edge (R)
<b>Bit 2</b>	Quadrature/Edge	quadrature enable / edge counter disable
<b>Bit 1</b>	PWM	PWM enable / pulse start
<b>Bit 0</b>	Clock select	clock select (0 – fast clock, 1 – slow clock)

Registers 1-3 are dependent upon the mode set on Bits 7-6 of Register 0. The table below describes the remaining XDIO registers:

**Table: TS-XDIO Registers 1-3 with MODE 0**

Register	Bits	Function and Description
<b>Reg #1</b>		DIO direction register ( <b>1</b> output, <b>0</b> input)
<b>Reg #2</b>		DIO data register
<b>Reg #3</b>	<b>bit 0</b>	irq/drq on quadrature change direction
	<b>bit 1</b>	irq/drq on quadrature / edge overflow / underflow
	<b>bit 2</b>	irq/drq on pin glitch (pins 0-3)
	<b>bit 3</b>	irq/drq on pin glitch (pins 4-7)
	<b>bit 4</b>	irq/drq on pulse timer trailing edge
	<b>bit 5</b>	irq/drq on quadrature / edge counter index pulse (if enabled)
	<b>bit 6</b>	reserved (may be used in customer specific TS-XDIO cores)
	<b>bit 7</b>	drq select ( <b>1</b> drq, <b>0</b> irq)

**Table: TS-XDIO Registers 1-3 with MODE 1**

Register	Bits	Function and Description
<b>Reg #1</b>		Glitch monitor, read or write resets all to 0
	<b>bit 7</b>	enable index pulse on pin 5
	<b>bits 6-0</b>	1 - glitch detected on pin 0 - inactive since last read
<b>Reg #2</b>	<b>bits 7-0</b>	bits 7-0 of edge / quadrature counter
<b>Reg #3</b>	<b>bits 7-0</b>	bits 15-8 of edge / quadrature counter

**Table: TS-XDIO Registers 1-3 with MODE 2**

Register	Bits	Function and Description
<b>Reg #1</b>		Pin select / timer
	<b>bit 7</b>	pulse time accumulate - keeps on running adding pulse times
	<b>bits 6-4</b>	input pin number (0-7)
	<b>bits 3-0</b>	bits 19-16 of pulse timer
<b>Reg #2</b>	<b>bits 7-0</b>	bits 7-0 of pulse timer
<b>Reg #3</b>	<b>bits 7-0</b>	bits 15-8 of pulse timer

**Table: TS-XDIO Registers 1-3 with MODE 3**

Register	Bits	Function and Description
<b>Reg #1</b>	<b>bits 7-4</b>	bits 11-8 of high time
	<b>bits 3-0</b>	bits 11-8 of low time
<b>Reg #2</b>	<b>bits 7-0</b>	bits 7-0 of high time
<b>Reg #3</b>	<b>bits 7-0</b>	bits 7-0 of low time, one-shot mode enabled if all zeros

### **TS-XDIO Programming Considerations**

- ✓ If quadrature index pulse is enabled, counter is reset on positive edge of index pulse if DIO Data Register, bit 5 is set to 0, negative edge if set to 1.
- ✓ Pulse width modulation (PWM) time is 2 + high / low time clock periods of 14.7456 MHz or 32 KHz.
- ✓ If using Pulse Width Modulation (PWM) in "One shot" mode (single pulse), Type is high pulse (low-high-low) if DIO Data Register, bit 4 is set to 0, low pulse (high-low-high) if set to 1.
- ✓ One could use the 20-bit Pulse Timer as a free-running timer by setting the Accumulate bit and setting pin select to PWM output (pin 5). 14.7456 MHz counter can be enabled by "watching periods," slower free-running counters can be had by clever use of high-time and low-time PWM and accumulating pulse times. (e.g. high-time 1, low-time 224 creates a 65.536Khz free-running timer that overflows once every 16 seconds).
- ✓ Edge counter counts both edges of a waveform; divide by 2 for number of periods / frequency. To count X edges and then interrupt, pre-load counter value with  $2^{16} - X$  and enable IRQ on overflow.
- ✓ Max/min quadrature can be counted by enabling IRQ / DRQ on quadrature change of direction. Max / Min is value +/- 1 as long as IRQ / DRQ latency is less than 1 quadrature cycle time.
- ✓ The XDIO Core does not reset -- registers and state remain if board undergoes a reset.
- ✓ Upon power-up, DIO Data Direction Register is all 0's (inputs), but other register contents are undefined


**Note**

Technologic Systems provides C API, source code and examples for TS-XDIO programming. Visit the [TS-XDIO webpage](#) for more details.

## 5.9 LCD Header

The LCD Port can be used to interface to a standard alphanumeric LCD display or as 11 additional digital I/O lines. The header has been arranged to allow a 14-pin ribbon cable to directly connect to industry standard LCD displays.

**Table: LCD Header Pin Configuration**

GND	Bias	LCD_WR	LCD_0	LCD_2	LCD_4	LCD_6
<b>2</b>	<b>4</b>	<b>6</b>	<b>8</b>	<b>10</b>	<b>12</b>	<b>14</b>
<b>1</b>	<b>3</b>	<b>5</b>	<b>7</b>	<b>9</b>	<b>11</b>	<b>13</b>
5V	LCD_RS	LCD_EN	LCD_1	LCD_3	LCD_5	LCD_7

Technologic Systems has a 2x24 LCD display available with software drivers for rapid development.

DIO lines LCD\_0 thru LCD\_7 are a byte-wide port using Port A on the EP9302 and are accessed via the data register at physical address location **0x8084\_0000**. The DDR for this port is at **0x8084\_0010**. Because this port is interfacing to a 5V LCD, 1.0K Ohm resistors have been added in series between the EP9302 and the LCD\_0 thru LCD\_7 pins. This is required since the LCD data bus could be driving these lines above 3.3V. The series resistors prevent the LCD from overdriving the EP9302 Port A pins.

If using these pins for general purpose DIO, the current sourcing and sinking capability of these DIO pins is limited by the 1.0K Ohm resistors.

LCD\_EN, LCD\_RS, and LCD\_WR are DIO pins using EP9302 Port H bits 3 through 5 respectively and are accessed via the data register at physical address location **0x8084\_0040**. The DDR for this port is at **0x8084\_0044**. When these DIO pins are configured as outputs, they can source 4 mA or sink 8 mA and have logic swings between 3.3V and ground. When configured as inputs, they have standard TTL level thresholds and must not be driven below 0 Volts or above 3.3 Volts. These DIO pins have 100K Ohm bias resistors biasing these inputs to a logic "1". It is important not to change the other bit positions in these Port H registers since the other DIO pins are being used on the **TS-7260**. All accesses to these registers should use read-modify-write cycles.

Pin 4 on this header (labeled Bias) is a 620 Ohm resistor to ground for LCD contrast biasing. The 5V power on the LCD header has a 750 mA Poly-Fuse to limit the current.



### Warning

LCD pins 3, 5, and 6 use 0-3.3V logic levels. Do not drive these lines to 5V.

### LCD Interface

A 14-pin LCD connector is provided for interfacing with standard alphanumeric LCD displays. These displays use a common controller, the Hitachi HD44780 or equivalent. While software written for the HD44780 works with all displays using the controller, the cable needed is dependent on the display used.

For most displays, a straight-through type ribbon cable can be used. The connector on the LCD display is typically mounted on the backside of the display.

**Table: LCD Header Signals**

PIN	Function	Comments
1	LCD 5V	LCD Power
2	LCD_GND	
3	LCD_RS	Register select
4	Bias	620 Ohm to GND
5	LCD_EN	Active high enable
6	LCD_WR#	Active low write
7	LCD_D1	D0 - D7: Buffered bi-directional data bus
8	LCD_D0	
9	LCD_D3	
10	LCD_D2	
11	LCD_D5	
12	LCD_D4	
13	LCD_D7	
14	LCD_D6	

Note that the pin-outs in the above table are not the standard given for LCD displays. However, these pin-outs allow a standard ribbon cable to be used when the ribbon cable is attached to the backside of the LCD.

**Warning**  
 ! Using an incorrect cable or mounting the LCD connector on the front-side can result in a reverse power polarity and can damage the LCD display. Please refer to your LCD data sheets for in-depth information.

Sample code is available for the LCD. Contact Technologic Systems for more information.

### 5.10 JTAG Header

The JTAG header can be used to program the on-board CPLD using special software and hardware support tools. However, it is not available for application debug purposes since it has no connection to the EP9302 JTAG interface. The following table shows which pins of the JTAG header are used for the JTAG interface signals:

**Table: JTAG signals at JTAG Header**

PIN	Signal	Description
9	TCK	Test Clock
11	TDI	Test Data In
13	TMS	Test Mode Select
14	TDO	Test Data Out
1, 3, 5	3.3 VCC	Power Supply
10, 12	GND	Ground

The JTAG header is also utilized for jumper configuration. See the Jumpers section of this manual for more details.

Technologic Systems has made the design choice to save on board real-estate and not bring out the JTAG header. If you need access to the JTAG pins, a skilled technician can solder wires to the pins.

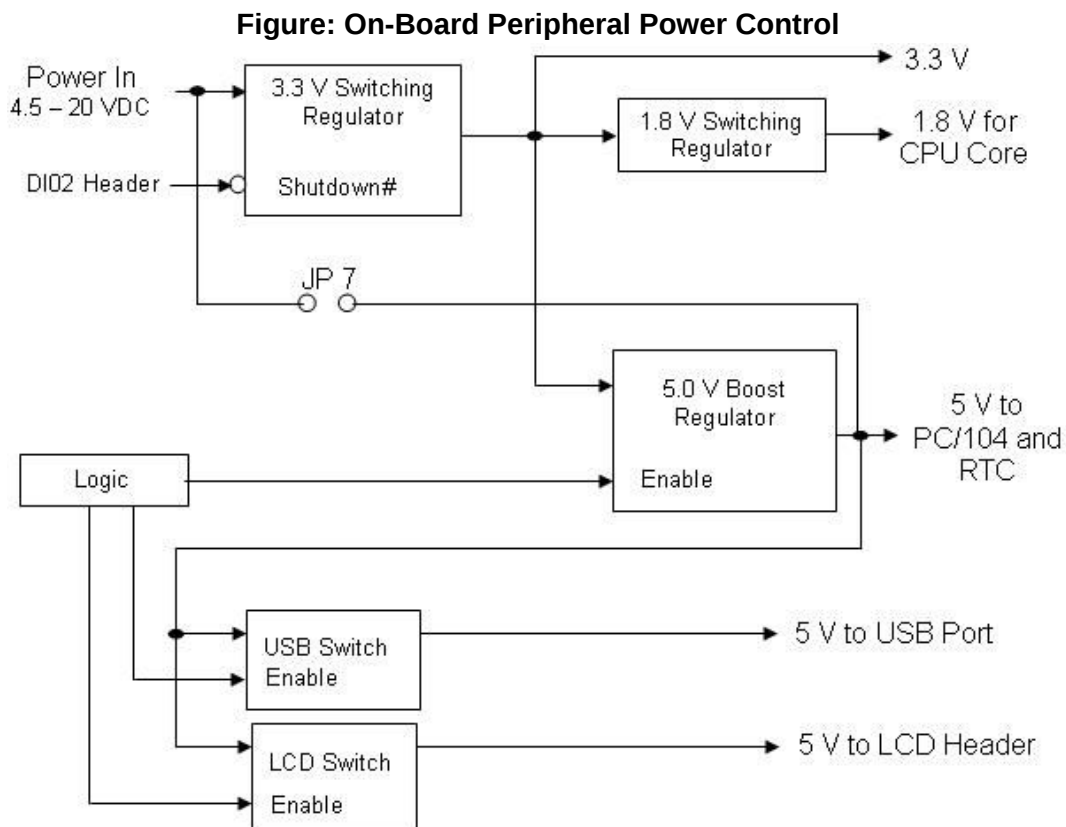
### 5.11 Power Supply Header

A quick release screw-down terminal block for the input power and GND connections are provided for easy connection to an external power supply.

An unregulated or regulated 4.5 - 20 VDC power supply, preferably with current limiting to less than 4 Amps may be used. A current limited supply is recommended.

#### Power Management

The **TS-7260** incorporates software controlled power features making it useful in applications requiring battery power or where power consumption is an issue. Figure 5 is a block diagram illustrating this control.



**Note**

JP 7 can only be installed if the input power is regulated 5 VDC.

The software control of these peripherals and other power savings are defined below through the register definition of the **TS-7260** specific 8-bit register at address, **0x1200\_0000**:



**Table: Power Management Register Description**

Bit	Description
<b>Bit 0</b>	RS232 level converter power (1 - on, 0 - off)
<b>Bit 1</b>	USB 5V power
<b>Bit 2</b>	LCD header 5V power
<b>Bit 3</b>	5V switcher enable (only setable if JP7 OFF)
<b>Bit 4</b>	PC104 clock disable (1 - clock disabled)
<b>Bit 5</b>	PC104 fast strobes (1 - enabled)
<b>Bit 6</b>	TTL uart enable (1 - COM3 switched to TTL on DIO3 header)
<b>Bit 7</b>	Scratch register (retains state during reboots)

The 5V switcher will default to on, '1,' if the board is powered up with JP7 off. If the board is powered up with JP7 on, this bit is '0'. Changing this bit during normal operation is not recommended.

The Scratch Register is intended to be used by Watchdog software to determine if last reboot was the result of watchdog expiry (value '1'), or normal system reboot/power-on (value '0').

The PC104 Fast Strokes Bit allows for non-ISA compatibility, but much faster bus cycle timing for custom boards. Contact Technologic Systems for more information. This bit should remain '0' for normal PC104 bus operation.

When the TTL UART Enable bit is set, COM3 RS232 signals RX and TX on the COM3 header are ignored and instead appear on pins 1 (TX) and 3 (RX) on the DIO2 header as 3.3V TTL signals. See section COM3 header for more information. This will override any settings for those pins in the DIO registers.

The least significant three bits of Model Register address, 0x22000000 are '010' (decimal 2 – **TS-7260** unique code).

Bit 0 of the Options Register refers to the state of OP-2TTLCOM (5 UART option) is in-place with (1), or TS-XDIO option (0).

## 6 PC/104 BUS EXPANSION

The PC/104 is a compact implementation of the PC/AT ISA bus ideal for embedded applications. Designers benefit from an established industry standard bus that already has well-developed documentation and many compatible peripherals available in the market place. The presence of a compact form-factor PC compatible standard has encouraged the development of a broad array of off-the-shelf products, allowing a very quick time to market for new products.

The electrical specification for the PC/104 expansion bus is identical to the PC ISA bus. The mechanical specification allows for the very compact implementation of the ISA bus tailor made for embedded systems. The full PC/104 specification is available from the IEEE Standards Office, No. IEEE P996.1.

This bus allows multiple peripheral boards to be added in a self-stacking bus. Since the electrical specs are identical (except for drive levels) to a standard PC ISA bus, standard peripherals such as COM ports, Digital I/O, Ethernet ports, and LCD drivers may be easily added.



### Note

The **TS-7260** implements a sub-set of the full PC/104 bus. This allows the support of many common I/O peripheral boards. Some of the PC/104 signals are not supported; for example, the DMA signals. These pins are used on the **TS-7260** to support data lines D8- D15. This means that a full 16-bit data bus implementation of the PC/104 bus is possible by only using the 64-pin connector. The supplemental 40-pin connector is only required for legacy boards – any new design can implement a full 16-bit data bus with only the 64-pin connector.

PC/104 peripherals will appear in the **TS-7260's** physical address space in one of four address regions depending upon whether it is emulating an x86 Memory cycle or I/O cycle and whether it needs to be a 8-bit cycle or a 16-bit cycle. Each region is a full 1MB in size, even though the I/O region will typically only use a 1 Kbyte region for legacy products.

**Table: PC/104 Memory Map**

Physical Address Region	Emulates x86 cycle
11E0_0000 thru 11E0_03FF	8-bit I/O cycles
21E0_0000 thru 21E0_03FE	16-bit I/O cycles
11A0_0000 thru 11AF_FFFF	8-bit Memory cycles
21A0_0000 thru 21AF_FFFE	16-bit Memory cycles

I/O cycles on the PC/104 expansion bus strobe either IOR# or IOW#, while Memory cycles strobe the MEMR# or MEMW# signals. For example, a TS-SER1 peripheral board can be jumper-selected as COM3, which would correspond to a PC I/O base address of 0x3E8. Since this is an 8-bit peripheral, this COM port must be accessed at the physical base address of 0x11E0\_03E8.

**Table: Signals on the 8-Bit Bus**

Pin	Signal	Pin	Signal
A1	BHE#*	B1	GND
A2	Data 7	B2	RESET
A3	Data 6	B3	+5 V
A4	Data 5	B4	Data 8*
A5	Data 4	B5	Reserved
A6	Data 3	B6	RTC_CS*
A7	Data 2	B7	Reserved
A8	Data 1	B8	RTC_ALE*
A9	Data 0	B9	Reserved
A10	IOCHRDY	B10	GND
A11	Address 20*	B11	MEMW#
A12	Address 19	B12	MEMR#
A13	Address 18	B13	IOW#
A14	Address 17	B14	IOR#
A15	Address 16	B15	Reserved
A16	Address 15	B16	Reserved
A17	Address 14	B17	Data 9*
A18	Address 13	B18	Data 10*
A19	Address 12	B19	Address 21*
A20	Address 11	B20	Data 12*
A21	Address 10	B21	IRQ7
A22	Address 09	B22	IRQ6
A23	Address 08	B23	IRQ5
A24	Address 07	B24	GND*
A25	Address 06	B25	Data 11*
A26	Address 05	B26	Data 13*
A27	Address 04	B27	Data 14*
A28	Address 03	B28	Data 15*
A29	Address 02	B29	+ 5 V
A30	Address 01	B30	Osc (14.3 MHz))
A31	Address 00	B31	GND
A32	GND	B32	GND

\* These signals have a non-standard usage.

## 6.1 Adding Serial Ports

If your project requires more than the included **TS-7260's** serial ports, additional ports may be added via the PC/104 expansion bus. Technologic Systems offers three different **peripheral boards** (TS-SER1, TS-SER2, and TS-SER4) that add 1, 2, or 4 extra COM ports respectively. These can be configured using PC designations COM1, COM2, or higher.

These ports will appear in the PC/104 8-bit IO address space (**0x11E0\_0000** base address). For example, a TS-SER1 peripheral board can be jumper selected as COM4, which would correspond to a PC I/O base address of 0x2E8. On the **TS-7260**, this COM port will appear at the physical base address of 0x11E0\_02E8. The PC/104 Expansion Bus has IRQ5, IRQ6, and IRQ7 available for additional serial port interrupts.

Typically each serial port has a dedicated interrupt, but the TS-SER4 peripheral board allows all four extra serial ports to share a single interrupt. This is very helpful in systems with a large number of serial ports since there are a limited number of IRQ lines available.

Drivers are available in the TS-Linux distribution to support these extra COM ports.

## 6.2 Adding Ethernet Ports

---

Additional ethernet ports can be added to the **TS-7260** using appropriate PC/104 hardware provided by Technologic Systems, allowing the implementation of hardware and software applications such as network firewalls, gateways or routers.

The **TS-ETH10** is a 10Mb ethernet PC/104 card based on the CS8900 ethernet chip that is perfectly suited for a WAN uplink. Up to 4 TS-ETH10 cards can be stacked on the PC/104 connector since the add-on board has 4 separate address decodes.

TS-ETH0 driver is available for TS-Kernel. Documentation and support is provided by Technologic Systems, as well.

## 6.3 Adding CAN Bus

---

Controller Area Network (CAN) is a serial network primarily used in embedded systems that was originally designed for the automotive industry, but has also become a popular bus in industrial automation and other applications. The addition of a CAN bus to the **TS-7260** is easily accomplished by adding the optional TS-CAN1 peripheral board through the PC/104 bus.

The **TS-CAN1** provides one single channel CAN interface and enables a PC/104 embedded systems to be connected to CAN1.1 and CAN2.0a networks. It uses a Philips SJA1000 CAN controller which is opto-isolated from the physical CAN network, increasing the system security. Up to four boards can be installed in any single system. The Philips SJA1000 CAN controller can also be found at Technologic Systems **TS-7KV** multi-function video board.

Technologic Systems provides complete SJA1000 Linux driver support for TS-CAN1 and TS-7KV, including CANOpen high level protocol. Documentation and support for CAN application development are also provided.

## 6.4 Adding Video Support

---

The **TS-7KV** multi-function PC/104 peripheral board can be added to the **TS-7260** if your application needs a video solution, . The TS-7KV implements the TS-VIDCORE using an FPGA solution to deliver 16-bit color, 640X480 resolution video, 8MB dedicated video memory running at 95Mhz, and a simple fast video accelerator. The TS-7KV includes a standard DB15 VGA connector and a 10 pin header. Regarding software support, an accelerated Linux framebuffer driver is available for TS-Kernel. In addition, the QT/Embedded graphical library can be used with the video solution and compiled packages can be found at Technologic Systems website.

The TS-VIDCORE included on the FPGA is a unique video core specifically designed in tandem with the Linux framebuffer driver. Hardware accelerations were implemented and designed to mesh well with the capabilities of the Linux 2.4 frame buffer device driver API. Technologic Systems engineers carefully chose which functionality was implemented in hardware, which functionality was left out for software and which functionality was hard-coded in the bitstream. The result is a higher speed design that is technically elegant and simple in good embedded systems fashion. The TS-7KV has dedicated video memory which allows CPU throughput and real-time response to remain fast and predictable, a clear advantage over devices which share system memory.



### Note

When running TS-Linux and using a TS-7KV video board, a USB mouse and a USB keyboard, it is possible to use the **TS-7260** embedded system as a desktop-like system, independent of the host machine. TS-Linux provides the complete solution for automatic configuration during boot time and console output at the plugged monitor. Contact Technologic Systems for further information.

## 7 LEDS, JUMPERS AND BUTTONS

### 7.1 Status LEDs

The **TS-7260** has two LEDs (one Red and one Green) available for user software. These LEDs may be used for diagnostics, status messages, and simple output. When power is first supplied to the **TS-7260**, both LEDs are immediately turned on under hardware control. Once the processor begins execution, the LEDs are turned off, and then flashed on and off again briefly. After booting is complete, these LEDs can be used for user applications.

The RED and Green LEDs can be controlled at physical address location **0x8084\_0020**. Bit 1 is the RED LED and bit 0 is the Green LED. A Logic "1" turns the LED on.

### 7.2 Jumpers

The **TS-7260** contains six default jumpers for specific board configuration:

**Table: TS-7260 Jumpers Description**

Jumper	Description
<b>JP1</b>	Boot to Serial Port COM1. This is a recovery mechanism that allows the <b>TS-7260</b> to boot using COM1 instead of the on-board Flash chip.
<b>JP2</b>	Enable Serial Console. COM1 is used as the Console. (If JP4 is not installed).
<b>JP3</b>	Write Enable Flash. Remove this jumper to write protect the Flash drive.
<b>JP4</b>	Console swapped to COM2 (Requires JP2 installed to enable console).
<b>JP5</b>	User Jumper.
<b>JP6</b>	User Jumper.

The **TS-7260** model contains three additional jumpers for specific board configuration on power management:

**Table: TS-7260 Jumpers Description**

Jumper	Description
<b>JP7</b>	Power In to + 5.0 VDC Bus. This Jumper should only be added if the board is using a regulated 5.0 VDC power supply and the application calls for the bussing of the internal and external power supplies.
<b>JP8</b>	Power In to +12 VDC PC/104 Power. This jumper should only be added if the power in to the board is a regulated +12 VDC power supply and the application calls for "passing" the power to the PC/104 Expansion Bus
<b>JP9</b>	Power to the Ethernet LED.

Jumpers 2, 3, 4, and 5 can be read at physical address location 0x1080\_0000. The status of all jumpers is available to software through the physical memory as describes the table below:

**Table: Memory Map of Jumpers**

Address	JP and Bit	On/Off
0x1080_0000	JP2 is bit 0	Logic "1" = Jumper ON
0x1080_0000	JP3 is bit 1	Logic "1" = Jumper ON
0x1080_0000	JP4 is bit 3	Logic "0" = Jumper ON
0x1080_0000	JP5 is bit 4	Logic "0" = Jumper ON
0x2280_0000	JP6 is bit 0	Logic "1" = Jumper ON

### 7.3 Buttons

---

On the **TS-7260** model, the reset push-button is replaced by pins 4 and 6 of the DIO2 Header. Refer to the DIO2 Header section to further information about the reset and shutdown process.

## 8 SPECIFICATIONS

---

To ensure optimum product operation you must maintain the operational environmental specifications listed in the table below.

**Table: Environmental Specification for TS-7260**

Environmental Specification	Standard Temp	Extended Temp
<b>Ambient Temperature</b>	-20° to +70° C The internal temperature must not exceed +70° C.	-40° to +85° C Extended temperature range is also standard in our TS-72XX products  <b>Note:</b> Extended Temp requires lower CPU speed (<=166Mhz) at higher temperatures  <b>Note:</b> Refer to your product manual, or contact Technologic Systems if the environmental temperature of the location is in doubt.
<b>Relative Humidity</b>	0 to 90% relative humidity. Not to exceed 90% non-condensing.	Not to exceed 90% non-condensing.

## 9 FURTHER REFERENCES

---

- ✓ **Getting Started with TS-Linux**  
(<http://www.embeddedarm.com/documentation/software/arm-linux-ts72xx.pdf>)
- ✓ **Linux for TS-ARM User's Guide**  
(<http://www.embeddedarm.com/documentation/software/arm-tslinux-ts72xx.pdf>)
- ✓ **TS-7200 Data Sheet** (<http://www.embeddedarm.com/documentation/ts-7200-datasheet.pdf>)
- ✓ **TS-7250 Data Sheet** (<http://www.embeddedarm.com/documentation/ts-7250-datasheet.pdf>)
- ✓ **TS-7260 Data Sheet** (<http://www.embeddedarm.com/documentation/ts-7260-datasheet.pdf>)
- ✓ **EP9301 User's Guide**  
([http://www.embeddedarm.com/documentation/third-party/ts-7000\\_ep9301-ug.pdf](http://www.embeddedarm.com/documentation/third-party/ts-7000_ep9301-ug.pdf))
- ✓ **EP9301 Data Sheet**  
([http://www.embeddedarm.com/documentation/third-party/ts-7000\\_ep9302-ds.pdf](http://www.embeddedarm.com/documentation/third-party/ts-7000_ep9302-ds.pdf))
- ✓ **TS-7000 Yahoo Users' Group** (<http://groups.yahoo.com/group/TS-7000/>)



**APPENDIX A: DOCUMENT HISTORY**


---

<i>Date of Issue/Revision</i>	<i>Revision Number</i>	<i>Comments</i>
January 27, 2006	Preliminary Draft	Preliminary release for first customer ship - not complete
March 03, 2006	1.0	Initial release
May 18, 2006	1.1	Minor corrections + SD Card section added
Jul 20, 2006	1.2	COM2/COM3 ports pin-out fixed + further info about CPLD configs + minor corrections and improvements
Jun 26, 2007	1.3	Minor corrections, changed series name to TS-72XX, updated product matrix and links section
Nov 21, 2007	1.4	Minor corrections
July 2, 2008	1.5	Fixed broken web links
May 29, 2009	1.6	Updated mailing address
January 26, 2010	1.7	Changed JP5 description
May 25, 2010	1.8	Fixed DIO2 blurb in "Common Interfaces"

**APPENDIX B: MEMORY AND REGISTER MAP**

Address Region	Function
0xF000_0000 - 0xFFFF_FFFF	nCS0 (not used)
0xE000_0000 - 0xEFFF_FFFF	SDRAM (TS-7250)
0xD000_0000 - 0xDFFF_FFFF	SDRAM (not used)
0xC000_0000 - 0xCFFF_FFFF	SDRAM (not used)
0x8084_0000 - 0x8084_00C8	GPIO control registers
0x8000_0000 - 0x800F_FFFF	AHB mapped registers
0x71C0_0000 - 0x71FF_FFFF	CS7 PC/104 8/16 bit I/O (user selectable timing)
0x7180_0000 - 0x71BF_FFFF	CS7 PC/104 8/16 bit Memory (selectable timing)
0x7000_0000 - 0x70FF_FFFF	TS-9420 Flash
0x7000_0000 - 0x7FFF_FFFF	CS7 (bit bus cycles)
0x6000_0000 - 0x60FF_FFFF	on-board Flash (TS-7200)
0x6000_0000 - 0x6FFF_FFFF	CS6 (Flash)
0x3000_0000 - 0x3FFF_FFFF	CS3 (not used)
0x21C0_0000 - 0x21FF_FFFE	PC/104 16-bit I/O
0x2180_0000 - 0x21BF_FFFF	PC/104 16-bit Memory
0x2000_0000 - 0x2FFF_FFFF	CS2 (16-bit bus cycles)
0x11C0_0000 - 0x11FF_FFFF	PC/104 8-bit I/O
0x1180_0000 - 0x11BF_FFFF	PC/104 8-bit Memory
0x1000_0000 - 0x1FFF_FFFF	CS1 (8-bit bus cycles)
0x0001_0000 - 0x0000_FFFF	SDRAM region

Register Address	Function
0x8090_0020	Cirrus A/D lock register
0x8090_0018	Cirrus A/D channel select register
0x8090_0008	Cirrus A/D result register (RO)
0x808D_0000 - 0x808D_FFFF	UART2 control registers
0x808C_0000 - 0x808C_FFFF	UART1 control registers
0x808A_0000 - 0x808A_FFFF	SPI control registers
0x8084_0044	LCD_EN, LCD_RS, LCD_WR direction reg.(bits3-5)
0x8084_0040	LCD_EN, LCD_RS, LCD_WR data reg.(bits3-5)
0x8084_0034	DIO_8 direction register (bit 1)
0x8084_0030	DIO_8 data register (bit 1)
0x8084_0020	On-board LEDs register (bits 0, 1)
0x8084_0018	Port C direction register (TS-7300) LCD_7 direction register (bit 7)
0x8084_0014	DIO_0 thru DIO_7 direction register (R/W)
0x8084_0010	LCD_0 thru LCD_7 direction register (R/W)
0x8084_0008	Port C data register (TS-7250) 1000 mA driver output on DIO (bit 0) (TS-7300) LCD_7 data register (bit 7)
0x8084_0004	DIO_0 thru DIO_7 data register (R/W)
0x8084_0000	LCD_0 thru LCD_7 data register (R/W)
0x8081_0000 - 0x8081_FFFF	Timer Control registers
0x8080_0000 - 0x8FFF_FFFF	APB mapped registers
0x800B_0000 - 0x800B_FFFF	VIC 0 registers
0x8006_0000 - 0x8006_FFFF	SDRAM control registers
0x8002_0000 - 0x8002_FFFF	USB registers
0x8001_0000 - 0x8001_FFFF	Ethernet MAC registers
0x7220_0000 - 0x729F_FFFF	(TS-7300) FPGA SDRAM region
0x7210_0000 - 0x7200_FFFF	(TS-7300) FPGA MAC core 32-bit registers

Register Address	Function
0x7200_0044 - 0x7200_0047	(TS-7300) FPGA DIO2 TS-XDIO #2 registers
0x7200_0040 - 0x7200_0043	(TS-7300) FPGA DIO2 TS-XDIO #1 registers
0x7200_0030 - 0x7200_003A	(TS-7300) FPGA Video core 16-bit registers
0x7200_0020 - 0x7200_0027	(TS-7300) FPGA SD Card core 16-bit registers
0x7200_0000 - 0x7200_001F	(TS-7300) FPGA COM3-4-5-6-7-8 16-bit registers
0x6000_0000	(TS-7250) NAND Flash data register
0x6040_0000	(TS-7250) NAND Flash control register (bits 0-2)
0x6080_0000	(TS-7250) NAND Flash Busy status (bit 5)
0x23C0_0000	WDT Feed register (bits 0-2)
0x2380_0000	WDT Control register (bits 0-2)
0x2340_0000	PLD version (bits 0-2)
0x2300_0000	COM2 RS-485 control register
0x22C0_0000	COM2 RS-485 control register (bits 0-2)
0x2280_0000	JP6 (bit 0)
0x2280_0000	Booting from TS-9420 (bit 1)
0x2280_0000	TS-9420 present (bit 2)
0x2240_0000	MAX197 A/D option present (bit 0)
0x2240_0000	COM2 RS-485 option present (bit 1)
0x2200_0000	Model Number (bits 0-2)
0x21E0_0000 - 0x21E0_03FE	PC/104 16-bit I/O (legacy support)
0x2100_0000	CF IDE 16-bit register
0x13C0_0000 - 0x13C0_0001	(TS-7300) FPGA loader registers
0x1340_0000 - 0x1340_0002	(TS-7260) Additional COM5 registers
0x1300_0000 - 0x1300_0002	(TS-7260) Additional COM4 registers
0x12C0_0000 - 0x12C0_0003	(TS-7260) DIO2 Header TS-XDIO registers
0x12C0_0000 - 0x12C0_0001	(TS-7260) DIO2 Header Basic DIO registers
0x1240_0000 - 0x1240_0002	(TS-7260) COM3 serial port registers
0x1200_0000	(TS-7260) Power Management register
0x11E0_0000 - 0x11E0_03FF	PC/104 8-bit I/O (legacy support)
0x11A0_0000 - 0x11AF_FFFF	PC/104 8-bit Memory (legacy support)
0x1170_0000	RTC R/W data register
0x1100_0001 - 0x1100_0007	CF IDE 8-bit registers
0x10F0_0000 - 0x10F0_0001	MAX197 A/D registers
0x1080_0000	MAX197 A/D busy bit (bit 7) (RO)
0x1080_0000	JP2-JP5 (bits 0,1,3,4) (RO)
0x1080_0000	COM1 DCD (bit 6) (RO)
0x1080_0000	Write Only index register
0x1040_0006 - 0x1040_0007	CF AUX IDE 8-bit registers

**APPENDIX C: DOWNLOADS - SCHEMATICS AND MECHANICAL DRAWING**

---

- ✓ **TS-7200 schematic**  
(<http://www.embeddedarm.com/documentation/ts-7200-schematic.pdf>)
- ✓ **TS-7200 mechanical drawing**  
(<http://www.embeddedarm.com/documentation/ts-7200-mechanical.pdf>)
- ✓ **TS-7200's download section** (<http://www.embeddedarm.com/epc/ts7200-spec-d.html>)
  
- ✓ **TS-7250 schematic**  
(<http://www.embeddedarm.com/documentation/ts-7250-schematic.pdf>)
- ✓ **TS-7250 mechanical drawing**  
(<http://www.embeddedarm.com/documentation/ts-7250-mechanical.pdf>)
- ✓ **TS-7250's download section** (<http://www.embeddedarm.com/epc/ts7250-spec-d.html>)
  
- ✓ **TS-7260 schematic**  
(<http://www.embeddedarm.com/documentation/ts-7260-schematic.pdf>)
- ✓ **TS-7260 mechanical drawing**  
(<http://www.embeddedarm.com/documentation/ts-7260-mechanical.pdf>)
- ✓ **TS-7260's download section** (<http://www.embeddedarm.com/epc/ts7260-spec-d.htm>)

**APPENDIX D: TS-ARM SBC FEATURE MATRIX**

<b>Product</b>	<b>TS-7200</b>	<b>TS-7250</b>	<b>TS-7260</b>	<b>TS-7300</b>	<b>TS-7400</b>
<b>CPU</b>	200 Mhz AMR920T	200 Mhz AMR920T	200 Mhz AMR920T	200 Mhz AMR920T	200 Mhz AMR920T
<b>PC/104 connector</b>	Yes	Yes	Yes	Yes	No
<b>On-board FPGA</b>	No	No	No	Yes	No
<b>RAM</b>	32 MB	32 MB	32 MB	32 MB	32 MB
<b>Optional RAM</b>	64 MB	64 MB 128 MB	64 MB 128 MB	64 MB 128 MB	64 MB 128 MB
<b>On-board Flash</b>	8 MB	32 MB	32 MB	No	32 MB
<b>Optional on-board Flash</b>	16 MB	64 MB 128 MB 256 MB	64 MB 128 MB 256 MB	Possible Contact us	64 MB 128 MB 256 MB
<b>Standard A/D</b>	Yes - 2 ch	Yes	Yes - 2 ch	Yes - 1 ch	Yes - 4 ch
<b>Optional A/D</b>	8 ch	8 ch	No	No	No
<b>Ethernet</b>	1x 10/100	1x 10/100	1x 10/100	2x 10/100	1x 10/100
<b>USB1.1 ports</b>	Yes - 2x	Yes - 2x	Yes - 2x	Yes - 2x	Yes - 2x
<b>VGA Video Out</b>	No	No	No	Yes	Yes
<b>IDE Compact Flash</b>	Yes	No	No	No	No
<b>SD Card Interface</b>	No	No	Yes - 1x on RevB	Yes - 2x	Yes - 1x
<b>Digital I/O</b>	20	20	30	55	20
<b>TS-XDIO</b>	No	No	Yes - 1	Yes - 2	No
<b>RS-485</b>	Opt. full/half	Opt. full/half	Opt. full/half	Opt. full/half	Opt. full/half
<b>Standard COM Ports</b>	2	2	3	10	3 TTL
<b>Optional COM Ports</b>	No	No	2	Yes	No
<b>RS-232 Console</b>	Yes	Yes	Yes	Yes	No
<b>RTC</b>	Opt.	Opt.	Opt	Opt.	Opt.
<b>LCD Interface</b>	Yes	Yes	Yes	Yes	No
<b>Keypad Interface</b>	Yes	Yes	Yes	Yes	No
<b>Linux2.4</b>	Yes	Yes	Yes	Yes	Yes
<b>Bootloader</b>	Redboot	Redboot	Redboot	Linux	Linux
<b>Real-Time RTAI</b>	Yes	Yes	Yes	Yes	Yes
<b>Java</b>	Yes	Yes	Yes	Yes	Yes
<b>USB WiFi Support</b>	Opt.	Opt.	Opt.	Opt.	Opt.
<b>USB Flash Support</b>	Opt.	Opt.	Opt.	Opt.	Opt.
<b>Extended Temperature</b>	Yes	Yes	Yes	Yes	Yes
<b>Switching-Mode Power Supply</b>	No	No	Yes	No	Opt.
<b>Quantity 1 Pricing</b>	\$149.00	\$149.00	\$179.00	\$219.00	\$129.00
<b>Quantity 100 Pricing</b>	\$119.00	\$119.00	\$149.00	\$189.00	\$99.00

**APPENDIX E: CONTACT TECHNOLOGIC SYSTEMS**

---



**16525 East Laser Drive  
Fountain Hills, AZ 85268  
TEL 1.480.837.5200  
FAX 1.480.837.5300**

**[www.embeddedARM.com](http://www.embeddedARM.com)  
[support@embeddedARM.com](mailto:support@embeddedARM.com)**

Call us Monday-Friday, **from 9 am to 5 pm**, Arizona-USA time;  
or email us at any time.

Our engineers answer tech support calls and are more than happy to talk to you  
about your needs and help you find the best solution for your project.