R T C - V 2 5

Computer/Controller Technical Manual

Release 0.91
1/28/91

Micromint, Inc.
902 Waterway Place
Longwood, FL 32750
www.micromint.com

DISCLAIMER

While we have attempted to provide accurate and up-to-date information in this manual, Micromint Inc. makes no representations or warranties respecting its contents. We reserve the right to make periodic changes to the text and to issue new editions of this manual without notification.

Occasionally in this manual we refer to other manufacturers' products. Such references do not constitute an endorsement of these products, but are included for the purpose of illustration or clarification. We do not intend such technical information and interface data to supersede information provided by individual manufacturers.

Conditions of Sale and Product Warranty

Micromint Inc. and the Buyer agree to the following terms and conditions of Sale and Purchase:

1. Micromint Inc. extends the following warranty: a factory-manufactured circuit board or assembly carries with it a one-year warranty covering both parts and labor. Any unit which is found to have a defect in materials or workmanship will, at the discretion of Micromint Inc., be repaired or replaced.

2. A minimum inspection fee must be prepaid for the repair of units that are no longer under warranty. Call Micromint Inc. for a current list of fees.

3. Micromint Inc. will not be responsible for the repair or replacement of any unit damaged by user modification, negligence, abuse and mishandling, or improper installation.

4. Micromint Inc. is not responsible to the Buyer for any loss or claim of special or consequential damages.

5. All units returned for repair must first receive prior authorization from Micromint Inc. A return authorization number may be obtained by phone or letter. Please retain a record of the return authorization number since most subsequent correspondence will refer to this authorization. Under no circumstances should any product be returned to Micromint Inc. without such authorization. Micromint Inc. assumes no responsibility for returns unaccompanied by an authorization number. All returns must be shipped prepaid and ought to be insured. Micromint Inc. is not responsible for losses or damage during shipment. Repaired units will be returned with postage and insurance paid.

6. Micromint Inc. reserves the right to alter any feature or specification at any time. This right extends to fees, charges, and any other conditions or warranties contained herein.

Table of Contents

1. Introduction

The Micromint RTC-V25 is a single-board computer/controller featuring a high-integration 16-bit microprocessor which maintains software compatibility with the Intel 8088/8086 while incorporating advanced design features in a single 84-pin PLCC package. Configured primarily for embedded control, the RTC-V25 uses the same vertical stacking I/O expansion bus as all of Micromint's RTC processor boards. All of Micromint's RTC expansion boards are compatible with the RTC-V25.

The optional RTC-V25 ROM monitor provides the system designer with a host of low-level development aids while myriad high-level development software for the IBM PC-series of desktop computers is available for more advanced applications.

(Please note that while the RTC-V25 board is code compatible with the 8088/8086 used in the IBM PC-series of computers, it is not an IBM PC-compatible board. It does not contain video circuitry, keyboard interface circuitry, disk drive interfaces, or an ISA expansion bus, and the serial, parallel, and real-time clock circuits are in no way software compatible with those typically found on an IBM PC compatible.)

RTC-V25 Technical Specifications

* NEC µPD70320 (V25) microprocessor running at 8 MHz (10 MHz optional). Supports a superset of the 8088/8086 instruction set
* Up to 384K bytes total memory on-board (32K/128K EPROM, 32K/128K static RAM, and 32K/128K of either static RAM or EPROM)
* 1024 bits (64 words x 16 bits) EEPROM
* Two asynchronous serial ports (one RS-232, one RS-232 or RS-485)
* Eight-channel, 8-bit or eight-channel, 10-bit analog-to-digital converter
* Four 8-bit parallel I/O ports (32 bits)
* Battery-backed clock/calendar
* Four predecoded expansion board select lines
* On-board memory management unit, 2-channel DMA controller, 2 counter/timers, and 17 interrupt sources
* Requires just +5V (RS-232 voltages generated on-board)
* Measures 3.5" x 5"

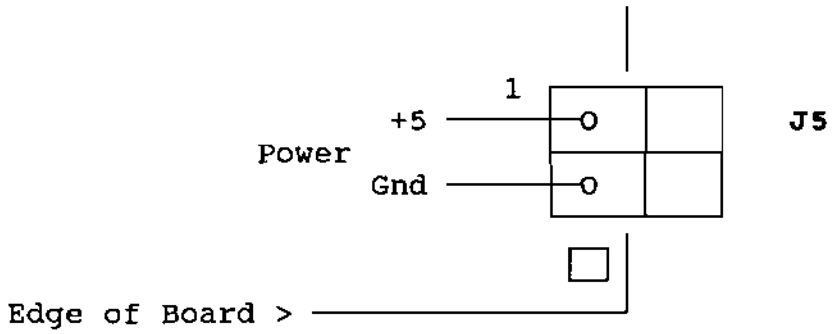
2. Setting Up Your RTC-V25

Very few connections are required to bring up a functional RTC-V25 development system, with the minimum connections being to a terminal and a power supply.

Power

The RTC-V25 typically requires a minimum of 350 mA at +5V ±5% for a fully populated board (all components installed including 384K of memory). Less current is required for a board with fewer components installed (e.g., without the RS-485 driver or less memory). Voltages necessary for RS-232 operation are generated on the board.

Power can be connected to the RTC-V25 in one of two ways: through the power pins on J5 or through the RTC bus. If you purchased the RTC-V25 with quick-disconnect headers, a mating connector and pins are included for J5 connection. If the board has screw terminals, simply insert wires in the top of the connector at J5 and tighten the side screws. The following top view of J5 should be used to properly connect the computer to your power supply. **Be sure to double check your connections before applying power to the RTC-V25! It is very easy to connect the power to J5 backwards. Boards damaged by incorrect power connections are not covered under warranty.**



If you plan to power your RTC-V25 through the RTC bus connectors, be sure to follow the connection instructions supplied with the board to which power will be connected. In no case should power be applied directly to the pins of the RTC bus expansion headers on the RTC-V25 board.

When plugging other RTC boards into the RTC-V25, be sure to line the keys up properly. **Plugging a board into the RTC-V25 backwards may damage the board. A board damaged by plugging it in backwards is not covered under warranty, so exercise appropriate care.**

Terminal

Any standard RS-232 serial terminal can used as a console device for the RTC-V25. Alternatively, a personal computer using communication software and a serial port can be used as a console device.

The RTC-V25 has a 16-pin Berg-type header (J2) used to connect with the console device. An adapter cable which goes from the Berg header to a standard DB-25 is useful for making the connection. Micromint sells such a cable (the VSC-25). If you'd prefer to construct one, please refer to the "Serial I/O" section later in this manual for a detailed description of J2. Be sure to match up pin one of the adapter cable with pin one of J2. You won't get any response from the terminal if the cable is plugged in backwards. With the adapter cable in place, a straight-through DB25-to-DB25 cable can be used to connect a terminal to the RTC-V25.

The same cabling is necessary in most cases when a personal computer is used with terminal emulation software. In some cases, though, a swap or "null modem" cable must be used. The minimum requirements for this cable are that pins 2 and 3 should be swapped and pin 7 should pass straight through.

Please refer to the "Serial I/O" section later in this manual if your terminal requires additional handshaking lines.

When installed, the optional RTC-V25 monitor expects the terminal to be set up for 9600 bps, 8 data bits, 1 stop bit, and no parity.

Powering it up

Once everything has been connected, double check all the connections. Turn on the power to the terminal first and let it warm up. Next, turn on the power to the RTC-V25. If the ROM monitor is installed, the terminal is set up for 9600 bps, and everything is connected properly, the monitor sign-on banner should appear on the terminal screen. Your RTC-V25 system is now working. You should now refer to the RTC-V25 monitor manual for instructions on how to proceed.

If the monitor banner isn't displayed when power is applied, or there is no response to repeated presses of the "Return" key, reset the board by shorting the pins of J1 and try again. If there is still no response, check all connections once again including the power and terminal cables. If you still can't get your RTC-V25 to respond, you may call our technical support staff at (203) 871-6170.

3. RTC-V25 Hardware

The hardware on the RTC-V25 can be broken into three sections: the V25 microprocessor, memory, and I/O. Each of these sections will be discussed in turn.

3.1 V25 Processor

The power of the RTC-V25 is made possible by the NEC V25--a high-performance, 16-bit, single-chip microcomputer with an 8-bit external data bus. It combines the instruction set of the μPD70108 (V20) with many on-chip peripherals.

The V25 processor has software compatibility with the V20 (and subsequently the 8088/8086), faster memory accessing, superior interrupt processing ability, and enhanced control of internal peripherals.

A variety of on-chip components, including 256 bytes of RAM, serial and parallel I/O, timers, and a DMA controller make the V25 a sophisticated microsystem.

Eight banks of registers are mapped into internal RAM below an additional 256-byte special functions register (SFR) area that is used to control on-chip peripherals. Internal RAM and the SFR area are together relocatable to anywhere in the 1M-byte address space. This maintains compatibility with existing system memory maps.

Features of the V25 include:

- o Complete single-chip microcomputer
  - 16-bit ALU
  - 256 bytes of RAM
- o Four-byte instruction prefetch queue
- o 24 parallel I/O lines
- o Two independent DMA channels
- o Two 16-bit timers
- o Programmable time base counter
- o Two full-duplex UARTs
- o Programmable interrupt controller
  - Eight priority levels
  - Five external, 12 internal sources
  - Register bank (eight) context switching
  - Eight macro service function channels
- o DRAM refresh pulse output

- o Two standby modes
  - HALT
  - STOP
- o Internal clock generator
- o Programmable wait state generation
- o Separate address/data bus interface
- o CMOS technology

Note: The V25 User's Manual is available from Micromint for $10 plus shipping.

3.2 Memory

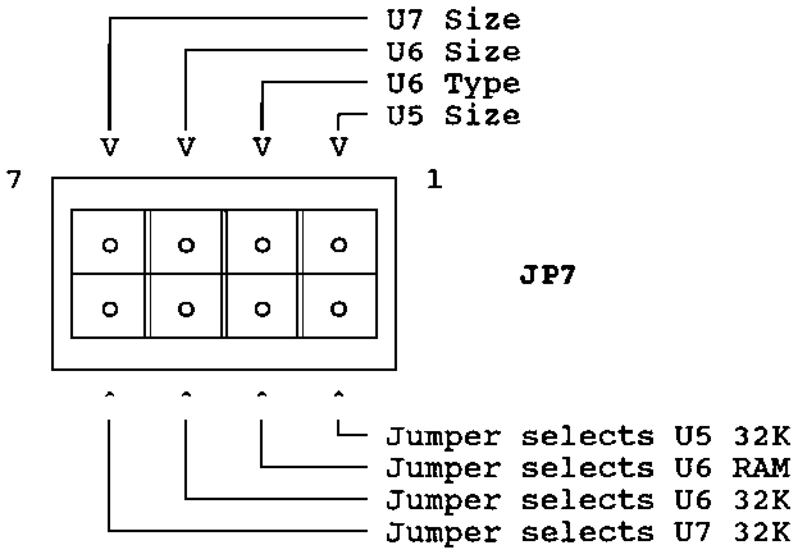
The RTC-V25 allows the use of up to 384K bytes of memory on the main board. In most controller applications, 256K is sufficient, however 384K is available for larger applications. Memory may not be expanded beyond 384K. The RTC expansion bus may only be used for I/O expansion and all system memory must be on the processor board.

EPROM/Static RAM

There are three 32-pin sockets on the RTC-V25 board (U5-U7). Each of the sockets may use either a 32K or a 128K device, depending on what is required by the application. A 27256/27C256 (32K) or 27C010/27C101 (128K) EPROM must be plugged into socket U7 and a 62256 (32K) or 628128 (128K) static RAM chip must be plugged into socket U5. Socket U6 may contain either a static RAM chip or an EPROM.

When plugging 28-pin devices into the 32-pin socket, the part should be lower justified in the socket. That is, when the chip is installed, socket pins 1, 2, 31, and 32 should be empty.

The 4x2 jumper header next to socket U9 (JP7) determines the sizes of all three chips installed and whether an EPROM or a RAM chip is installed in U6. In the case of device size selection, a jumper installed on a header selects a 32K device while removing the jumper selects a 128K part. In the case of device type selection, an installed jumper selects RAM while a removed jumper selects EPROM. The RTC-V25 is shipped from the factory with all four jumpers installed, which sets the board up for a 32K RAM in U5, a 32K RAM in U6, and a 32K EPROM in U7. The following shows how JP7 is set up:



How the memory is mapped into the V25's address space depends on how JP7 is configured. Please refer to Appendix A for a complete description of memory mapping.

Wait States and Refresh

The V25 processor has, among other things, an on-board wait-state generator. It's possible to set anywhere from 0 to 2 wait states for both memory and I/O accesses. Memory wait states are dealt with in this section. See the "Input/Output" section for I/O wait state details.

EPROMs and static RAM chips rated for an access time of 85 ns or better can be used with zero memory wait states on an 8-MHz RTC-V25. Memory chips rated at 150 ns or faster need at least one wait state, while slower chips require two wait states. At no time should chips slower than 250 ns be used with the RTC-V25. Refer to the WTC special function register described in the V25 User's Manual for more details on how to set wait states.

Since the RTC-V25 contains only static memory devices, the V25's on-board DRAM refresh circuits may be disabled to eliminate the overhead of DRAM refresh. The V25's RFM special function register should be initialized to zero by any program responsible for setting up the processor after a reset. See the V25 User's Manual for more details about the RFM register.

EEPROM

The RTC-V25 also sports 1024 bits of Electrically Erasable Programmable Read-Only Memory, configured as 64 words of 16 bits. EEPROM is useful for storing small amounts of information that doesn't change often, such as power-on configuration settings. The EEPROM has a rated life of 10,000 erase/write cycles, which is more than adequate for storing seldom-changed data, but not for general data storage.

The EEPROM interfaces to the V25 processor through bits 0-3 of Port 0. Bit 0 should be set up for input operation and is connected to the data out pin on the EEPROM. Bits 1-3 should be set up as outputs and control the EEPROM data input, clock, and chip select lines, respectively.

In order to set up Port 0 properly to access the EEPROM, the following values should be sent to the corresponding V25 special function registers:

```
00h -> PMC0       ; Set port 0 for port mode
F1h -> PM0        ; Bit 0 input, bits 1-3 output
00h -> P0         ; Chip select off, clock low, data low
```

Appendix B contains pages taken from National Semiconductor's "Nonvolatile Memory Databook" which show the proper timing and commands necessary to access the EEPROM. The sample C code in Appendix C may also help in developing your own application code.

3.3 Input/Output

The RTC-V25 features an I/O space of 65535 (64K) I/O ports accessible through on-board peripherals and through the vertical stacking expansion headers. Expansion is only limited by the number of boards that can be physically stacked on top of the RTC-V25. (Use of more than two or three expansion boards isn't recommended, however.)

The V25 has 65 on-board "special function registers" that are used to control the chip's serial ports, interrupts, timers, and so on. These are mapped into a block of 256 memory locations. The block's location in the memory map is controlled by the IDB register which is fixed at address FFFFFH. The use of these registers is well-documented in the V25 User's Manual, so won't be covered in this manual.

The RTC-V25 also has on-board one 8255 programmable peripheral interface chip that provides three 8-bit parallel I/O ports, a real-time clock/calendar, and an 8-channel 8-bit or 10-bit A/D converter. These are mapped into the V25's I/O space (which is separate from the memory space) with base port addresses fixed at 0000H, 2000H, and 4000H, respectively. See the following sections on parallel I/O, analog input, and the real-time clock for more details. Also see Appendix A for a summary of port locations.

In order to access any peripheral device mapped into the I/O space, the complete 16-bit port address must be specified by any program. The normal V25 "IN AL,port" and "OUT port,AL" instructions cannot be used since they only specify an 8-bit port address. Instead, the "IN AL,DX" and "OUT DX,AL" instructions should be used. These use the 16-bit value found in register DX as the port address and use AL for data.

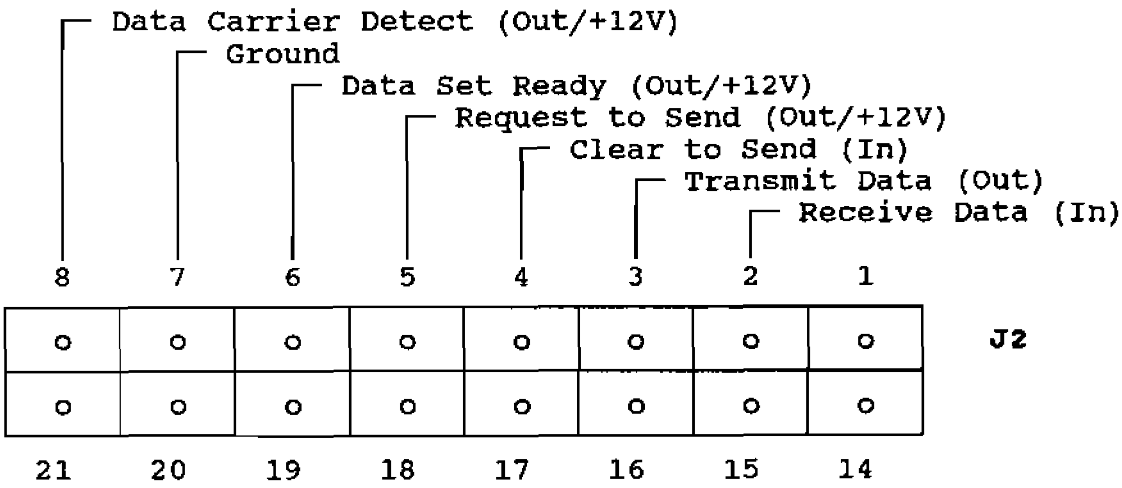
The V25 has an on-board wait-state generator that allows the system to generate anywhere from 0 to 2 I/O wait states. Wait states are used to compensate for a fast processor trying to access a slow peripheral device. It's difficult to predict exactly how many wait states a particular I/O device is going to need. It is safest to set the maximum number of I/O wait states (2), get the device working, then gradually reduce the number of wait states until the device stops working. You may actually want to leave the wait-state generator set to a value one greater than the minimum found by experimenting. This will prevent flakey operation caused by a device being pushed to its limits. Refer to the V25 User's Manual for more details on how to set the on-board wait-state generator (specifically, the WTC register).

3.3.1 Serial I/O

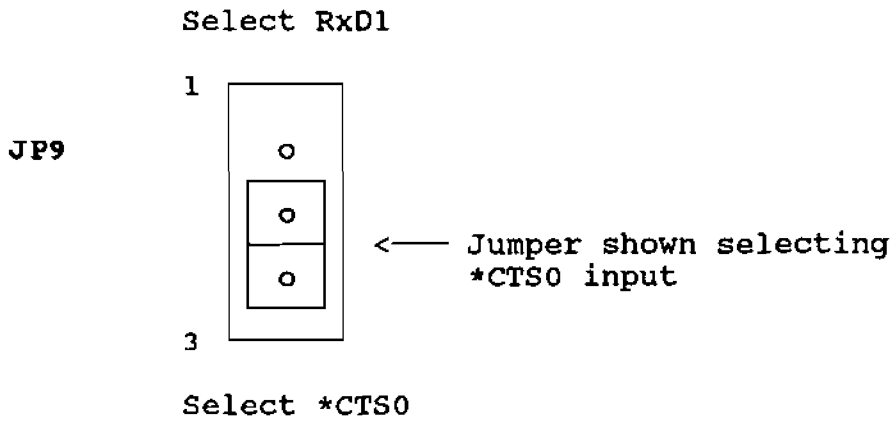
The V25 processor used on the RTC-V25 contains two asynchronous serial ports on the chip which have been brought out to headers on the RTC-V25 so they can be used in applications.

Serial Port 0

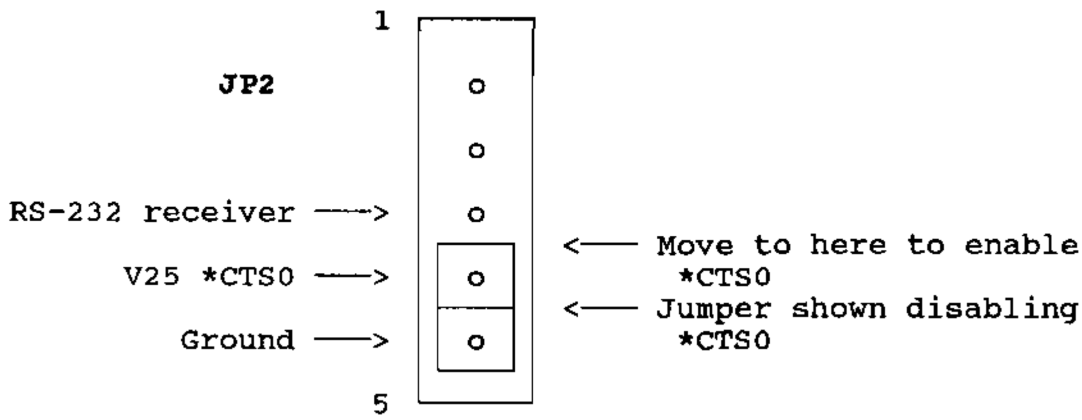
Serial port 0 is normally used as the console port. The serial transmit, receive, and CTS handshake lines are sent through a level shifter to interface with a standard RS-232 terminal and are brought out to a Berg-type 2x8 header (J2) on the RTC-V25 board. Pins 5 (RTS), 6 (DSR), and 8 (DCD) are pulled high by the RTC-V25 to satisfy the requirements of some terminals. The following shows the pinout of J2:



Two jumpers are used in relation to the CTS input on port 0. JP9 is used to share one of the RS-232 receiver circuits between port 0 CTS and port 1 receive data. When port 1 isn't being used or is set for RS-485 operation, port 0 is free to use the receiver circuit for CTS. JP9 should be set as follows to tie in port 0 CTS:

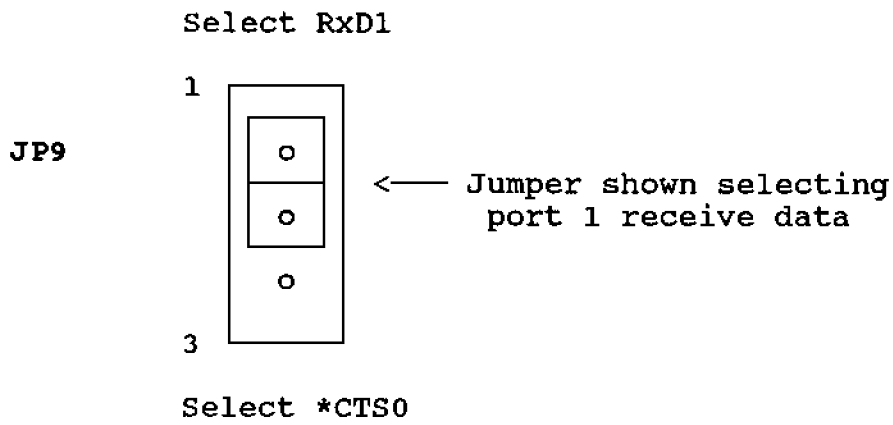


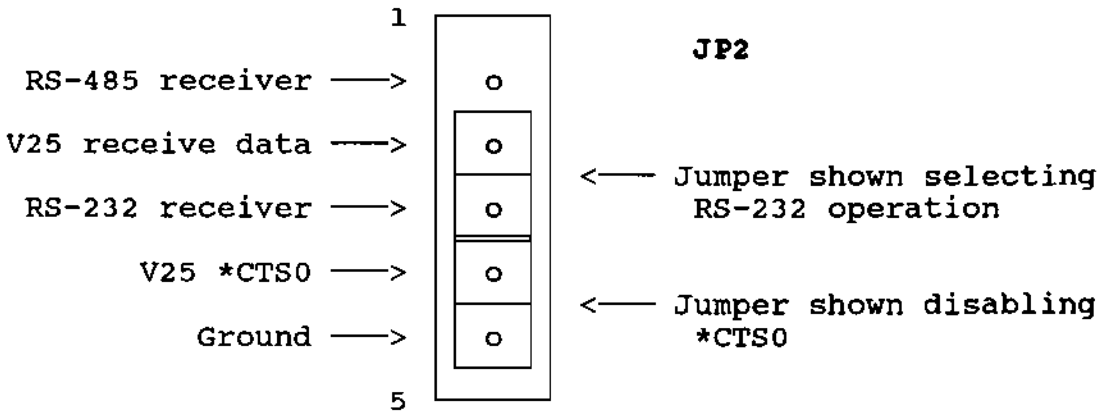
The other jumper that must be set is JP2. When there is a jumper between pins 4 and 5, the port 0 CTS input to the V25 is tied to ground, effectively enabling the line all the time. Moving the jumper to pins 3 and 4 connects CTS through to the processor. The following shows how to set up JP2 to enable or disable port 0 CTS:



Serial Port 1

Serial port 1 is an auxiliary serial port which features both an RS-232 and an RS-485 interface. To use the port with the RS-232 interface, it is necessary to disable port 0's CTS input and also connect port 1's receive data line to the RS-232 interface circuitry. JP2 and JP9 must be set as shown below for RS-232 operation on port 1:





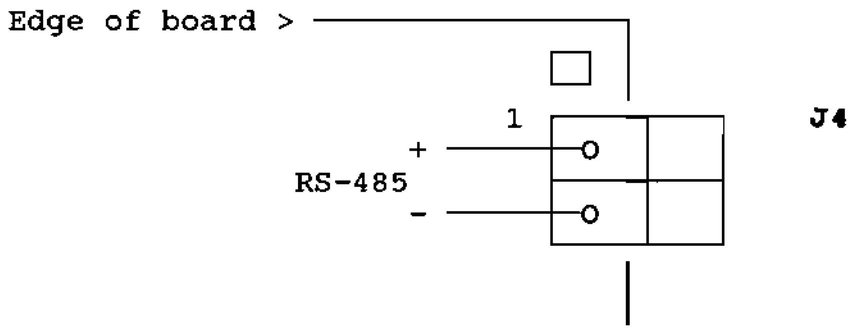
Port 1 may also be used with an RS-485 interface. A standard RS-232 interface is called "single ended." That is, there is a common ground between the devices connected together, and all signals are referenced to that ground. For example, a high signal is +12V referenced to ground and a low signal is -12V referenced to ground.

RS-485 uses what is called "double ended" or "balanced" signaling. In other words, there is no common ground between the devices connected together. Instead, the voltage differential between a pair of wires is used. When that differential voltage is positive, a high signal is denoted. A low signal is represented by a negative differential voltage between the wires.

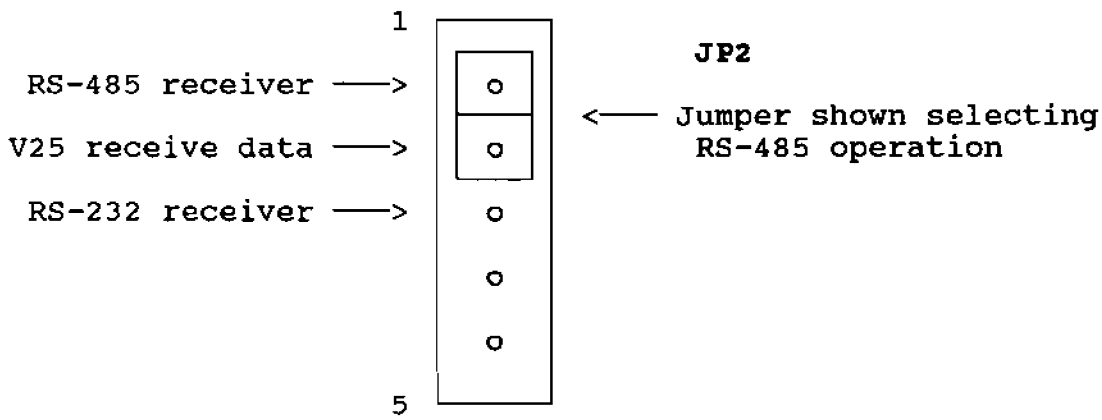
The biggest advantage to using balanced signaling is its increased noise immunity and, hence, increased communication distance. The reason for the increased noise immunity is as follows: when the signal reaches the receiver, the voltage on one wire is subtracted from the voltage on the other. The result is either a positive or negative voltage indicating either a high or a low signal. If some noise is introduced onto the signal on its way from the driver to the receiver, that noise appears on both wires with equal magnitude. When the voltages on the wires are subtracted at the receiver, the noise is canceled out and a clean signal results.

In RS-485 operation, a single twisted pair is used to communicate in both directions. As a result, half-duplex must be used (i.e., data can only flow in one direction at a time). The advantage gained by using RS-485 is that more than two devices may be connected to the same twisted pair, party line fashion. The RS-485 spec states that up to 32 devices may be connected to the same twisted pair. Using this setup, it's possible to implement a low-cost LAN. Protocols used by the devices to arbitrate who talks when are strictly up to the system designer. The EIA RS-422/485 specification states that connections up to 4000 feet using 24-gauge wire may be made at rates under 100 kbps and up to 32 devices may be connected to a single RS-485 twisted pair.

Polarity is important when connecting the twisted pair to the RTC-V25. The positive terminal on each board must be connected to the positive terminal of the other boards on the same twisted pair, and likewise for the negative terminals. The following shows how J4 should be connected:

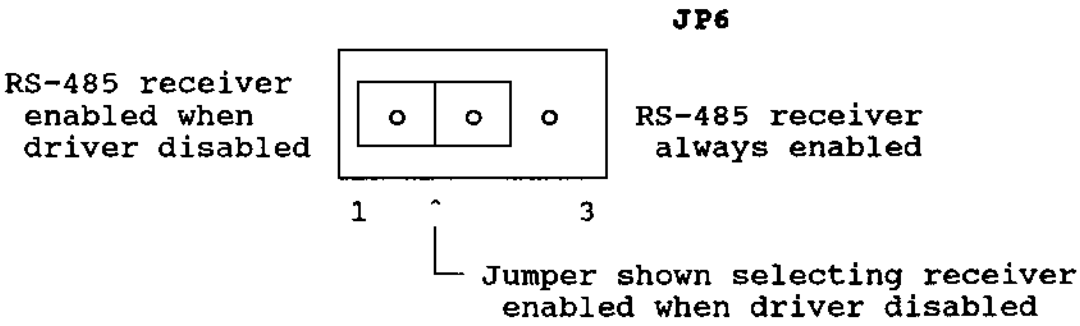


As above, jumper JP2 must be set to properly route the serial signals through the appropriate interface. The setting of JP9 isn't critical since it only affects RS-232 operation. JP2 should be set as follows for RS-485 operation:



Jumper JP6 determines whether the RS-485 receiver interface is always enabled, or is only enabled when the transmitter is disabled. Leaving the receiver enabled all the time allows a program to check its outgoing characters against what it simultaneously received as a simple contention check. On the other hand, immediately receiving each character that is sent out may be too burdensome for a program, so the receiver and transmitter may be enabled alternately.

Install a jumper across pins 2 and 3 on JP6 to enable the receiver all the time. Put the jumper on pins 1 and 2 to enable the receiver only when the transmitter is disabled. The following shows how to set JP6:



A terminating resistor must be installed on the boards at the two ends of the twisted pair cable. Termination is enabled by installing a jumper on JP5. If more than two boards are connected to the same twisted pair, only the boards at the two ends of the cable should be terminated. All boards in between should have termination disabled.

From a software standpoint, the only important item to worry about is enabling and disabling the transmitter. The transmitter should only be enabled when a character is ready to be sent, and should be disabled at all other times. Failure to disable the transmitter blocks all other parties connected to the same twisted pair from transmitting.

To enable the driver circuit, bit 5 of the V25's port 0 should be set high. Before using port 0, though, special function registers PMC0 and PM0 should be initialized so that bit 5 acts as an output. Then it's possible to simply send data to register P0 to turn the RS-485 driver on and off. See the V25 User's Manual for more information.

A short settling delay may be needed after enabling the transmitter to allow the lines to quiet before transmission takes place.

3.3.2 Parallel I/O

Three parallel input/output ports are provided on the RTC-V25 board with an 8255 Programmable Peripheral Interface chip. The 24 I/O bits and ground are available on a 26-pin Berg connector (JP17) located next to the 8255.

The 8255 is a programmable interface device. The 24 I/O lines are divided into three ports--A, B, and C--each separately configurable either as input, output, or handshaking lines under software control. A control register on the chip defines the characteristics of each of the ports on the chip.

The 8255 uses four ports in the RTC-V25's I/O space starting at 0000H and going through 0003H. Port A is at 0000H, port B is at 0001H, port C is at 0002H, and the control port is at 0003H.

The 24 lines of the 8255 are divided into two groups of 12 lines each: group A (port A and the upper half of port C) and group B (port B and the lower half of port C). The functional configuration of each port is set by the system software. The RTC-V25 outputs a control word to the 8255 which contains information such as "mode," "bit set," "bit reset," and so on, that initializes the 8255. The control register is a write-only port (i.e., the control word cannot be read by examining the contents of the control register port).

When the power to the computer is turned on, the 8255 is in an unknown configuration. Before the ports can be used they must be initialized by loading a control word into the control register port. The following is a list of control word values for some typical 8255 port configurations:

Control Word Value	Port A	Port C Low	Port B	Port C High
80H	output	output	output	output
81H	output	output	output	input
82H	output	output	input	output
83H	output	output	input	input
88H	output	input	output	output
89H	output	input	output	input
8AH	output	input	input	output
8BH	output	input	input	input
90H	input	output	output	output
91H	input	output	output	input
92H	input	output	input	output
93H	input	output	input	input
98H	input	input	output	output
99H	input	input	output	input
9AH	input	input	input	output
9BH	input	input	input	input

Signal	Pin	Pin	Signal
Ground	1	2	Ground
PB3	3	4	PB4
PB2	5	6	PB5
PB1	7	8	PB6
PB0	9	10	PB7
PC7	11	12	PA7
PC6	13	14	PA6
PC5	15	16	PA5
PC4	17	18	PA4
PC3	19	20	PA3
PC2	21	22	PA2
PC1	23	24	PA1
PC0	25	26	PA0

JP17

A complete specification of the 8255 is not included in this manual. Should you need to configure the 8255 for a more complicated I/O configuration or require the use of handshaking, refer to an 8255 data sheet published by one of the following manufacturers:

Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

NEC Electronics USA, Inc.
One Natick Executive Park
Natick, MA 01760

National Semiconductor Corporation
2900 Semiconductor Drive
Santa Clara, CA 95051

3.3.3 Analog Input

The RTC-V25 includes on the board either an 8-channel, 8-bit or an 8-channel, 10-bit analog-to-digital converter. Operation of the two are nearly identical, with the only difference being an extra read of the chip required for the 10-bit version.

Voltages presented to the ADC **must** be in the range of 0V to 5V. Anything outside this range will do permanent damage to the ADC chip. It is the responsibility of the system designer to ensure that voltages always remain within range. For the 8-bit ADC, the input voltage is broken into 256 steps of 19.5 mV each, while the 10-bit parts uses 1024 steps of 4.88 mV.

Up to eight separate channels may be connected at any given time, and the processor is free to access any of the eight randomly. Only one conversion may be performed at a time, however. Connection of the analog inputs to the RTC-V25 is through J6 as shown below.

N/C	Ch7	Ch6	Ch5	Ch4	Ch3	Ch2	Ch1	Ch0	N/C
19	17	15	13	11	9	7	5	3	1
o	o	o	o	o	o	o	o	o	o
o	o	o	o	o	o	o	o	o	o
20	18	16	14	12	10	8	6	4	2
Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd

J6

Before any conversions may be done, however, the ADC chip requires a clock input. On the RTC-V25, this is provided by the output of the V25's on-board timer 0, so the timer must be intialized before conversions may be done. The following special function registers must be initialized to generate the clock:

```
00h -> MD0      ; Set maximum speed
ACh -> TMC0     ; Start count, enable TOUT, interval timer
20h -> PMC1     ; Enable TOUT
```

The resulting clock runs at 666.7 kHz and results in a conversion time of about 100 microseconds (35 microseconds for the 10-bit part).

Once the clock is running, it's time to talk to the ADC. It is addressed starting at port 4000H, with channel 0 being at 4000H and progressing up to channel 7 at 4007H. To start a conversion, simply write any value to the port corresponding to the channel of interest. When the conversion is complete, read the same port once for the 8-bit ADC, or twice for the 10-bit ADC (to get all 10 bits, two reads are required). With the 10-bit device, the first read results in the high-order eight bits and the second read results in the low-order two bits, left justified as follows:

High Byte	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
Low Byte	2^{-9}	2^{-10}	0	0	0	0	0	0

Refer to the sample C code found in Appendix C for an example of how to initialize the timer and access the ADC.

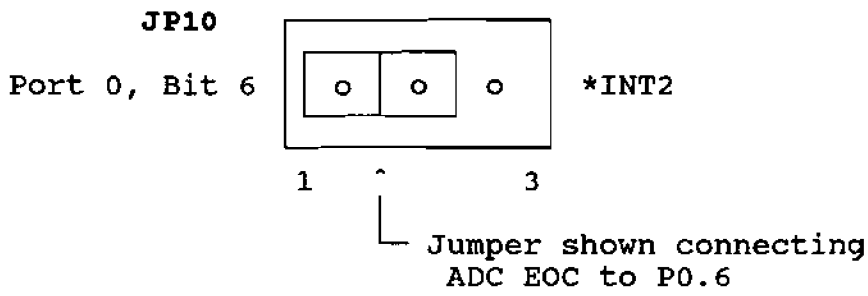
Knowing when a conversion has completed can be handled in one of several ways. The easiest is to simply insert a fixed delay of at least 100 μs (35 μs for 10-bit ADC) between starting the conversion and reading the result. Two other techniques involve using the EOC signal from the ADC chip.

Depending on the setting of JP10, EOC is connected to either the *INT2 interrupt input to the V25 or to bit 6 of the V25's port 0. The EOC signal is normally high; when a conversion is started, EOC will go low sometime between 0 and 14 microseconds after the start, then will go high at the end of the conversion.

To use EOC when connected to *INT2, be sure there is a jumper between pins 2 and 3 of JP10. Next, set up the V25's port 1 bit 3 for interrupt operation using special function registers PMC1 and PM1. Then set up *INT2 to interrupt on the rising edge using special function register INTM, and enable the interrupt using the EXIC2 register.

To manually poll EOC rather than use an interrupt, place a jumper between pins 1 and 2 of JP10. Set up bit 6 of port 0 for input operation using special function registers PMC0 and PM0, and EOC may be read by reading port 0 and masking bit 6.

The following shows how to configure JP10:



3.3.4 Real-Time Clock/Calendar

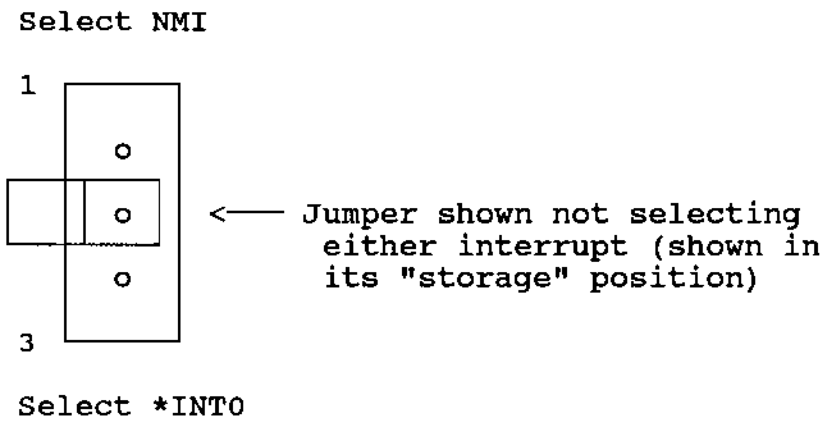
Time-of-day and day-of-week information may be stored and retained through power failures using the MSM6264B Real-Time Clock/Calendar chip found on the RTC-V25. Periodic interrupts may also be generated by the clock chip. The MSM6264B is decoded starting at I/O address 2000H and occupies 16 consecutive I/O ports as shown in the following table:

I/O Port Address	Register Name	Description
2000	S1	1-second digit
2001	S10	10-second digit
2002	MI1	1-minute digit
2003	MI10	10-minute digit
2004	H1	1-hour digit
2005	H10	10-hour digit, AM/PM
2006	D1	1-day digit
2007	D10	10-day digit
2008	MO1	1-month digit
2009	MO10	10-month digit
200A	Y1	1-year digit
200B	Y10	10-year digit
200C	W	Week
200D	CD	Control Register D
200E	CE	Control Register E
200F	CF	Control Register F

Please refer to Appendix B for complete information on accessing and using the MSM6264B. Appendix C also contains sample C code for dealing with the MSM6264B on the RTC-V25.

In order to use the interrupt generation capability of the clock/calendar chip, JP11 must be set up to direct the interrupt to either *INT0 or NMI on the V25. In both cases, special function registers PMC1 and PM1 must be set up to select interrupt operation for port 1, register INTM must be configured to select the falling edge for the interrupt, and, in the case of *INT0, the interrupt must be enabled using register EXIC1. Refer to Appendix B for details about setting the MSM6264B's interrupt mode.

The following shows how to set up JP11:



3.3.5 LED Indicator

The RTC-V25 contains a single LED which may be used as a general-purpose indicator. The LED is connected to the the V25's port 1 bit 4. Setting the bit turns the LED on and clearing the bit turns the LED off.

To use the LED, the following special function registers must be initialized or accessed:

```
00h -> PMC1     ; Set P1.4 for port operation
00h -> PM1      ; Set P1.4 for output
10h -> P1       ; Turn LED on
00h -> P1       ; Turn LED off
```

Refer to the sample code in Appendix C for an example of accessing the LED from C.