

Mouse Systems Corporation
M-1 Mouse
Technical Reference Manual

© March 1983 Mouse Systems Corporation

Warning

The mouse generates and uses radio frequency energy. Although we know of no possible configuration of the mouse which would cause it to interfere with radio or television reception, if interference is suspected (which can be determined by turning the mouse off and on), we suggest you consult your dealer or an experienced radio/television technician for advice. The mouse has been certified to comply with the limits for a Class B 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 in a residential installation.

Warranty

Your mouse is warranted against defects in material and workmanship for a period of one year from the date of purchase. The obligation of this warranty shall be limited to repairing or replacing any part of the product which, in the opinion of MSC, shall be proved defective in materials or workmanship under normal use and service during the one-year period commencing with the date of purchase. Contact the factory for a return authorization number. Return postage pre-paid to Mouse Systems Corporation, 2336H Walsh Avenue, Santa Clara, CA 95051.

This one-year warranty is in lieu of all other express warranties, obligations or liabilities. Any implied warranties, obligations or liabilities, including but not limited to any implied warranty of merchantability, shall be limited in duration to the one-year duration of this written limited warranty. Any action for breach of any warranty hereunder, including but not limited to, any implied warranty of merchantability, must be brought within a period of 18 months from the date of purchase. Some states do not allow limitations on how long an implied warranty lasts, so the above limitation may not apply to you. No agent, representative, dealer or employee of MSC has the authority to increase or alter the obligations of this warranty. This warranty shall not apply to any product which, in the opinion of MSC, has been modified, repaired, or altered in any way without the express written consent of MSC. This warranty shall not apply to the felt feet which are expendable. In no case shall MSC be liable for any consequential damages for breach of this or any other warranty expressed or implied whatsoever. Some states do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

1. Introduction

This document explains all about the MSC M-1 optical mouse. MSC has been shipping the M-1 unit since 1982 and this document is constantly updated with feedback from our customers. If you have any questions, comments, or problems, feel free to contact us.

MSC is constantly improving the M-1 mouse. Our new M-2 mouse will use almost half as many components as the M-1 and many potential problems with the M-1 will be non-existent in the M-2. However, new products take time.

Based on field experience at hundreds of installations, the M-1 mouse has proven to be extremely reliable computer peripheral. The contents of this manual should answer all your questions in the very remote possibility that something goes wrong. If it doesn't, don't hesitate to call or write us.

2. Normal use

The mouse is a fairly simple device. Basically all you do is plug it into your system and go.

When the mouse is first powered up, it goes through a calibration sequence:

MOVE THE MOUSE IN *WIDE* CIRCLES (E.G., 6 INCH DIAMETER) AT A COMFORTABLE SPEED UNTIL THE CURSOR ON THE SCREEN MOVES. This should only take a few seconds.

The remainder of this document is for users who must attach the mouse to a system or who want to know more.

3. Connecting the mouse to your system

Connecting the mouse is straightforward. It will either plug into an existing mouse port in your computer, or into the RS-232 interface box. If you are using the RS-232 interface box, be sure the power supply is plugged into the wall and the mouse is plugged into the interface box. Plug the provided 25-pin connector into your computer and the other end into the interface box. The two RJ11C connectors on the interface box are interchangeable.

Orient the pad in the proper direction (see DIP switch options).

As with most computer devices, power to the mouse should be applied rapidly (e.g., by turning on a switch or quickly plugging in a connector). Generally this is done by a wall or power-strip switch so that it is not necessary to plug/unplug the mouse cables and/or power supply.

4. Mouse use

The mouse works best when it is within $\pm 45^\circ$ from its nominal (vertical with respect to the selected pad) orientation. Most mouse users place their wrist on the surface and grasp the mouse with their thumb and little finger. Because the mouse can be lifted and re-positioned, the screen cursor can (and often is) *always* be positioned anywhere on the screen without having to move any muscles above the wrist.

One finger may be used for all three buttons, or the mouse buttons may be used one finger per button.

There are about as many different styles of mouse usage and preferences for mouse shape as there are people. The MSC M-1 design attempts to satisfy the broadest range of people.

5. DIP switch options

Various options may be set by changing the DIP switches inside the mouse. A paper clip is a good tool to use for this purpose. Mice are normally shipped so the pad is horizontal.

<u>Switch</u>	<u>OFF</u>	<u>ON</u>
1	1200	300 baud
2	non-rotatable	rotatable protocol
3	normal	self-test
4	horiz	vert blue lines
5	normal	noiseless
6	RS-232	TTL

The pad orientation can be adjusted to suit your desk space or screen aspect ratio preferences. The mouse is roughly rotation insensitive to $\pm 90^\circ$ from nominal (straight up and down).

If the serial protocol is selected, switch 6 controls whether the data is inverted or not. In the TTL mode, a start bit is a "low" TTL voltage.

The disadvantage of selecting the rotatable protocol as standard is that the maximum acceleration of the mouse is reduced.

Self-test mode is 3 ON; 1 OFF.

LED range test is 3 ON; 1 ON.

For processors which cannot tolerate a continuous stream of information when the mouse may not be moving (i.e., when the mouse is on a line boundary), "noiseless" mode may be selected at the expense of single count motion (motion of positive 1 unit will be delayed). Note however, that the mouse will rarely transmit continuously in its normal mode so that this switch is rarely used.

Mice with a parallel interface (i.e., not RJ11C or RS-232), *must be operated with all switches OFF*. However, the self-test switches will still work.

6. Opening up the mouse

The mouse must be opened to change the DIP switch settings. Remove the two screws on the bottom of the mouse. Then place the mouse on a table as if you were going to use it. Then remove the top of the mouse.

There are only four things inside the mouse; 2 lenses, a PC board, and a rubber light shield. Please note the placement of these components when you open the mouse. DO NOT touch the mirror; it is easily scratched. DO NOT touch the LEDs; they have been bent in that funny position in order to provide the proper amount of illumination. When assembling the mouse, be sure the rubber light shield is centered (see the dot on the plastic case).

7. Power-cycling the mouse

If it is desired to power-cycle the mouse, wait at least 5 seconds before plugging the mouse back in. This allows sufficient time for the microprocessor reset capacitor to discharge. Pre-mature re-plugging could cause unpredictable results, but will not harm any of the mouse components. As mentioned in section 3, power should be applied quickly.

8. Calibration sequence

Each time the mouse is powered on, the integrity of the microprocessor and the mouse circuitry are verified. Any problems appear as error codes (section 22).

Next, the mouse tries to sequentially calibrate each LED it needs for the mode selected by the internal DIP switches. Only the currently calibrating LED is turned on. Since the IR LEDs are invisible, pressing a button will cause both red LEDs to turn on (you will have to turn the mouse over to see this). This indicates that the mouse is receiving power and calibration is proceeding. However, **DO NOT** press any of the buttons while calibrating. This test is *only* to help you diagnose any problems.

The mouse must be moved at a relatively comfortable speed in order for the LEDs to calibrate. The IR LEDs only detect crossings of green lines and the red LEDs only detect crossings of blue lines. It is sufficient to move the mouse in wide circles to enable calibration in both directions.

The mouse will only calibrate on surfaces of sufficiently high contrast ratio, such as the mouse pad provided.

CALIBRATION IS FINISHED when at least one LED is on and pressing a switch seems to make no difference on which LEDs are on. Of course, in practice it is not necessary to make this test since when the mouse is calibrated, the cursor on the screen will start moving. If the mouse is in non-rotatable mode, only one red LED will be on. If the mouse is in rotatable mode, both LEDs should be on.

The mouse adjusts to whatever surface it is placed on. There is some time constant associated with this (about 7 seconds). For fastest calibration, the mouse should be placed on the pad when it is powered up and left on the pad until calibration is finished. There is no time constant after calibration. Once the mouse is calibrated, it remains calibrated so long as power is supplied.

If the mouse usually takes more than 15 seconds to calibrate, you may be moving in circles that are too small, moving too fast or too slow, or pressing a switch. If this is not the case, see section 21.

9. What to do if the mouse doesn't work

We've never had any mouse we've built fail; even after being (accidentally) dropped six feet onto concrete. However, we are not naive enough to think that this will always be the case. So all sorts of error detection/error recovery software is built into the mouse.

When the mouse is powered on, pressing a switch should cause the two red LEDs to light. If this doesn't happen, it probably means the mouse isn't plugged in or there is a bad cord somewhere. Try wiggling the mouse and power supply cords around especially near a connector.

If it seems to take endless time to get out of the calibration sequence (e.g., you've been moving in large circles at a normal speed for 30 seconds) turn the mouse over. If it is flashing in a regular sequence, see section 22.

If the mouse does not calibrate and no error code is flashing, the cause could be some obstruction of the optical path (cracked lens, mispositioned light shield, mirror slippage), a mistake by the calibration software in thinking that a side which is taped up is good (most likely caused by a poor taping job; see section 23), an LED which has become very weak (see section 21), or a bad mouse pad.

If the mouse completes the calibration sequence (section 8) but shows no screen motion, the problem could be in your computer tracking software. If you are using the RS-232 interface, you may have ordered the

wrong cable. Try reversing pins 2 and 3 (or purchase a "null modem") if you suspect this is the case. The mouse does not set DSR, CTS, etc.

Finally, if the mouse has been dropped a great distance, an LED may be knocked out of position. Try repositioning the LED and then check (Section 21).

10. Self-test mode

Self-test mode provides a quick way for the user to determine whether his mouse may be in need of repair.

First, unplug the mouse cable. Then open up the mouse case and turn switch #1 OFF and switch #3 ON. Plug the cable back in. All LEDs should be OFF when the mouse is exposed to a sufficient amount of room light. When any mouse button is pressed, two LEDs should turn ON. When all three buttons are pressed, the two LEDs will flash. If all this happens, reset the switches the way they were, re-assemble the mouse, and press all three buttons at once to restart the mouse. It is not necessary to power-cycle the mouse.

Any deviation from the above behavior means your mouse either isn't getting power (or getting insufficient power) or isn't working. Try switching with a friend before you call MSC.

11. Curing poor mouse performance

The mouse should not mis-track in normal use. At high speeds and when the mouse is lifted from the surface while moving at high speeds, occasional mis-tracking may occur.

Erratic tracking of the mouse under normal conditions could be due to not moving in large circles at smooth speed during calibration, pressing a switch during calibration, an LED knocked out of position, improper DIP switch settings (e.g., baud rate or pad orientation), dust in the optical path, a misaligned light shield, a misaligned LED, a loose piece rattling around, a large change in temperature, or a very large power spike.

Power-cycling the mouse will usually cure the problem.

See also section 21 and section 23.

12. Care of the mouse pad

The mouse pad is made out of aluminum and coated with a very hard organic coating. However, the pad is susceptible to scratches and dents. You can wash it and window glass cleaner on it.

Some mouse pads may not have a logo on them. Without a logo, your eye doesn't have anything to focus on when it looks at the pad. However, since the pad is reflective, the effect is ameliorated.

13. Replacement parts

We have included replacement felt feet for your mouse. Be careful not to tear the felt when removing the backing from the felt.

14. Redundancy

If the mouse is used in non-rotational mode, only one set of LEDs is used. Should one of the LEDs fail, the mouse will automatically use the second set of LEDs the next time the mouse is powered on.

15. Additional features

All mice have a timeout feature. After 1 hour, the rate at which the LEDs are sampled is decreased to 10 Hz. The rate is restored to normal at the sign of first motion. This feature is transparent to the user, but results in ridiculously high MTBFs and low power consumption.

16. Unusual applications

The mouse can be used as a digitizer if a transparency of the item to be digitized is placed over the CRT screen.

17. For owners of early mice (Serial number under 4000)

Transparent inks, such as used on the mouse pad, are somewhat susceptible to UV radiation. The mouse pad should not be exposed to direct sunlight (i.e., taken outdoors). The prototype mouse pad (without UV absorber) was rendered helpless after 5 weeks of direct sunlight (equivalent to about 15 years of constant exposure to room level fluorescent lighting). Please note that it is perfectly safe to operate the mouse indoors in direct sunlight since window glass absorbs UV.

18. For more information

The mouse is covered by and described in U.S. Patent 4,364,035.

Mouse repair

19. Introduction

MSC mice have been designed to be trouble-free. At the time of this writing, we have never had any mouse failures. However, even the best designs can sometimes fail. This section will help you narrow down the problem.

20. Self-test mode

The self-test mode is used for testing the mouse boards at the factory before they are inserted in the plastic case.

Self-test requires the PC board only, case and cable are not required. The PC board should be exposed to room light. Flip the switch(es) to self-test (section 5) and apply power.

If LED1 goes on, it means not all the integrators got reset; the problem could be caused by U1, U2, or U3.

LED3 will remain on whenever the detector is not receiving enough room light. If LED3 remains on when the light level is increased, the problem is in the detector, U1, U2, or U3, or the room isn't light enough.

If all has gone well, the LEDs should be off. Whenever a button is pressed, both red LEDs will light. This happens no matter what other errors have occurred. If only one LED lights, the other LED is probably broken. If both LEDs light when only some switches are pressed, then the other switches are broken. If both LEDs are always on, one of the switches is stuck in the "closed" position or you aren't in self-test mode.

Pressing all three switches at the same time will restart the processor at location zero. This will also cause a noticeable "blinking" of the LEDs.

Grounding U3 pin 1, 2, or 14 should cause LED1 to go on. Shorting C1, C2, or C3 should cause LED3 to come on.

At the end of the self-test period, reset the DIP switches, then press all three buttons at once. The processor be out of self-test mode (pressing all three switches won't do anything special).

21. LED range check

To verify that all LEDs are providing sufficient light output, set DIP switches 1,2, and 3 ON and power cycle the mouse. Then go through the standard calibration sequence. If the mouse successfully calibrates (both red LEDs are ON), everything is ok. If the mouse is flashing an error code, see section 22. Otherwise, the mouse has either finished calibration because it thinks that one LED set has a broken LED(s), or is stuck trying to calibrate a weak LED. Since the calibration sequence is 4,3,2,1, the stuck LED should be easy to locate. Usually just re-positioning the LED should cause it to function effectively. Reset the DIP switches and power-cycle the mouse.

In rare cases, it is possible that the mouse may perform poorly even though it passes the range test. This is caused by uneven illumination of the imaging point on the mouse pad. Re-adjust the LED so the beam is unobstructed and try the range check again.

22. Mouse error codes

There are 10 error codes which can be flashed. 8 of the codes are user-serviceable with no test equipment. The last two error codes mean the mouse has discovered an internal failure.

If the mouse flashes one or two LEDs one or more times with about 3 seconds between flashes, the mouse has discovered an error inside itself.

When the mouse is in LED range check mode, flash codes 1 thru 8 are enabled. Flash codes 1 thru 4 means that the given LED is too bright. The mouse should be opened and the flagged LED should be adjusted away from the hole. Flash codes 5 thru 8 are analogous to codes 1 thru 4 except the meaning is that the LED is not bright enough. The problem LED is obtained by subtracting 4 from the error code. The LED should be adjusted closer to the hole. The LED numbers are marked on the PC board inside the mouse.

Please observe the position of the LEDs before making any attempt to bend them should this be necessary. The approximate position of the LEDs is 30° from vertical. The LEDs should be positioned so that: the hole is half obscured when looking vertically down; the axis is 30° from vertical; the two LEDs are centered over the hole; the top is a couple of mm from the PC board.

Error codes 9 and 10 are not user serviceable; it indicates there is an error in the microprocessor (9) or in the external circuitry (10).

Error codes can be recovered from by pressing all three buttons at once. This is exactly like power-cycling the mouse (this three button restart is, of course, only enabled when an error code is flashing).

Please contact the factory if you get an error code; they should never happen.

23. Fault simulation

The redundancy feature (section 14) may be used to help isolate a performance problem. If the lens opening on the bottom of side of the mouse of the current LED set is taped up with black tape, insufficient light will reach the detector during calibration (for very bright LEDs, it might be necessary to open the mouse and tape one of the holes on the PC board). This will simulate a fault condition and the mouse will automatically take corrective action.

OEM interfacing

24. OEM Software interfacing

For the IBM PC, MSC provides a variety of software support packages. For other machines, we will provide code samples and suggested high level interfaces on request. If you would like to market your mouse software through MSC, we would be delighted to talk with you.

There are two types of protocols: rotatable and non-rotatable. In the rotatable mode, the position of two sensors are transmitted. This makes it possible for the computer to determine the rotational and translational orientation of the mouse. However, most applications only care about how the mouse has been translated so that the non-rotatable protocol should be used. We encourage you to use this mode since it allows the mouse to be fully redundant in the case of an LED failure.

The protocol is as follows: five byte data blocks are sent when and only when there is a change of mouse state. The start of a data block is indicated by a sync word whose upper 5 bits are 10000. The next 3 bits are the debounced state of the switches (0 means depressed). If the rotatable protocol is selected, the next four bytes give the change in the position of the right sensor (viewed from above) of the mouse since the last transmission, and the change in the position of the left sensor since the last transmission. Each byte is a two's complement packed 8-bit binary number. The sensors are located 100 counts apart 1 inch below the center of the mouse. Rotation information is typically obtained by updating unnormalized direction cosine registers (e.g., adding $\Delta x_2 - \Delta x_1$ to the x-cosine-register).

Mice are shipped with the non-rotatable protocol enabled. This protocol sends two updates of the x and y registers in each five byte block: switches, $\Delta x_1, \Delta y_1, \Delta x_1, \Delta y_1$ (note that Δx is the distance moved since the last transmission of Δx relative to the coordinate system of the pad). The following table summarizes this (the rotatable protocol is in parentheses):

Byte	Contents
1	1 0 0 0 0 $L M R$
2	Δx_1
3	Δy_1
4	Δx_1 (Δx_2)
5	Δy_1 (Δy_2)

The mouse only sends out data when it thinks it has changed state (a switch transition or movement). In rare cases where the mouse is positioned exactly on the boundary of a line, the mouse may (more or less) continuously send out data (this will not happen if noiseless mode is selected). If the mouse moves more than one pixel on the screen when it is stationary, it probably means the host computer is missing characters.

Mouse tracking software should read bytes until a byte with a 10000 high-order bit pattern is read. The switches are the lower 3 bits of this byte (0 means pressed). Next, read two sets of two bytes. The interpretation of these bytes depends on the protocol the user has selected. Mice are normally shipped with all DIP switches off meaning non-rotatable protocol at 1200 baud. After a total of 5 bytes have been read, the software should ignore bytes until the "start" byte is recognized. This ensures synchronization. Since extraneous bytes are not sent in the M-1 mouse, testing for extraneous bytes is a good way to debug your tracking software.

After initializing the UART to 8 data bits, no parity, 1200 baud operation, the basic algorithm is as follows:

1. Read a byte
2. If upper bits are 10000, then save this in SWITCH-STATUS. Otherwise, goto 1 (extraneous byte received). XOR'ing the new SWITCH-STATUS and old SWITCH-STATUS will indicate whether a button has changed state.
3. Add the next byte to the X-REGISTER (be sure to do an 8086 CBW instruction first if you are on a 16-bit machine).
4. Add the next byte to the Y-REGISTER.
5. Add the next byte to the X-REGISTER.
6. Add the next byte to the Y-REGISTER.
7. Go To 1.

Now, asynchronously with this operation, the computer may LOAD-AND-ZERO these registers. Depending on the program controlling the mouse cursor, these values may or may not be added to the mouse cursor position register (e.g., some software may restrict the mouse cursor to certain areas of the screen at certain times).

Some systems may update the mouse cursor after reading the first two-byte set. Other systems may wait longer depending on how long it takes to move the mouse cursor on the screen. If you update x and y screen registers at the same time the screen tracking will look smoother than if you don't. How much smoother is anyone's guess.

Even if you only update the screen cursor at 30 Hz, DO NOT throw away the second set of bytes if you are using the non-rotatable protocol. Those bytes really are significant and you could cause serious mistracking if you ignore them. You will also get (on the average) half the resolution meaning you have to move the mouse twice as far. Remember, you don't have to redraw the cursor every time the mouse position is updated.

Be sure not to let the mouse cursor go off-screen.

Many systems use a 2:1 screen-to-mouse ratio. That is, 1 inch of mouse motion gives 2 inches of screen motion. This usually requires that the mouse position information be multiplied by 2 before being displayed. Hence, every other dot on the screen is addressable. This has never turned out to be a problem; single "pixels" should normally be addressed with the screen "magnified" so that a single pixel is magnified into a square array of pixels. This is because human pointing has a limited dynamic range; it is inefficient to require someone to point with better than 10 mil accuracy.

We strongly urge you to try 2X magnification. Most software engineers are reluctant to do so, but after they try it, they find the feeling of control and speed far outweighs the inability to choose single pixels. Also be careful on magnification since some screens are not square. The general rule is that if you draw a circle with the mouse, it should look like a circle on the screen.

There has been some confusion about the non-rotatable protocol: switches, $\Delta x_1, \Delta y_1, \Delta x_1, \Delta y_1$. The second and third bytes are not the same as the fourth and fifth bytes. They are the same register which is sent and cleared *twice* in the five byte packet. This is because it is a more efficient use of the available bandwidth to send the position information more often than the switch status. The mouse NEVER sends high and low bytes of a sixteen bit word. In other words, treat the second and third bytes just like the fourth and fifth bytes.

When using the RS-232C interface, the mouse does not set DSR, CTS, etc. These signals can usually be ignored by software. In some cases, user software must contain code which specifically tells the UART to ignore these signals. Before the mouse is calibrated, it will send a start bit continuously. Detecting this (via the UART BREAK detection) will indicate either an uncalibrated or non-working mouse. You may need to reset the UART after the receipt of the BREAK.

The mouse sends 8 bit bytes with no parity. All eight bits contain significant data. Operating system features that change a bit (e.g., always clear high order bit), swallow nulls, respond to ↑S or ↑Q, and/or replicate DELETE should be disabled.

16-bit machines should remember to propagate the sign bit (all quantities are two's complement). Use the 8086 instruction CBW (Convert Byte to Word) to do this.

Some users have asked for a poll mode where the mouse can be queried for its position. This is unnecessary overhead for both the mouse and the host system: the host must be capable of updating the screen on every refresh for good human factors. Hence, polling is redundant.

If you don't expect to use the rotation feature of the mouse (most systems don't), DO NOT have the user select the rotatable format and ignore the last two bytes. This is because the maximum acceleration is reduced by a factor of 4 in this case. However, it turns out that even if the rotatable protocol is selected, if one of the LEDs fails (or is deliberately taped up), the maximum acceleration is the same as for the non-rotatable case, but rotation detection is lost since the second and third bytes are always the same as the fourth and fifth bytes (that way, software using the rotatable protocol is still usable in the event of an LED failure since the mouse "looks" like a rotatable mouse that is never rotated).

Since 5 bytes are transmitted at 1200 baud, the maximum sustained velocity of the mouse should be under 61 in/sec (30 in/sec for rotatable protocol) for real-time tracking (internally, the mouse uses a 15 bit buffer so the maximum sustained velocity is a lot faster than anyone is ever going to achieve). The mouse position is available for refresh 48 (24 for rotatable) times/sec. Both these numbers are reasonable, however. Velocities seldom exceed 30 in/sec and 24 frames/sec is the frame rate for motion pictures. The mouse can transmit at higher baud rates but the performance advantage is negligible compared with the added burden on the mainframe.

Since the mouse performs poorly at 90° and it is uncomfortable for the hand to rotate more than 90° and because the cord gets in the way, the mouse should probably be used as a *relative* rotation device too (just as it is a relative position device), e.g., the difference $\Delta x_2 - \Delta x_1$ gives the amount to move the tip of the rotatable screen vector to the right. The user is free to lift the mouse and rotate it and put it back down. Then, for example, the user can move a "left" pointing arrow around on the screen while leaving his mouse in the much more comfortable nominal orientation.

Since the sensors are located 100 counts apart 1 inch below the center of the mouse (x_1 is to the right of the centerline), it is strictly necessary to do some simple coordinate transformations (contact the factory on this) though in practice, probably no one will ever know. Be aware that the counters are NOT snap-shotted since the microprocessor doesn't have a lot of memory. Hence, only at the very end of a motion will the coordinates be strictly correct. The same is true for the non-rotatable protocol.

If the software interface is correct, the mouse will track reliably and accurately under normal conditions. It will never drift on the screen. Under the non-rotatable protocol at 1200 baud, no time-delay will be apparent to the user, even at high speeds.

For examples of innovative mouse software, we refer you to Apple's Lisa, Xerox's Star, VisiCorp's VisiOn, or any of the products from Lisp Machines, Inc. or Symbolics (both in Cambridge, MA).

25. OEM Hardware interfacing

If you go direct into a UART be sure to set the data output polarity switch correctly. Don't forget the 1K pull-up resistor. The mouse requires 5V @ 150mA. The RJ11C connector pinout is GND (1), +5 (2), OUTPUT to UART(3). To determine which is pin 1, unplug the line cord (not the handset cord) from your telephone. Grasp the connector with the "pins" facing you and the cord hanging straight down (or towards you). Then pin 1 is on your left. Note that the connector is a six position connector with four pins (1-4).

The levels sent on the RS-232 interface are RS-232 compatible, although they do not meet RS-232 specifications. Specifically, the upper level is +5 V with some ripple and the lower voltage is -5 V (regulated). We used this scheme for increased reliability and low cost: the RS-232 level converter is nothing more than a 1K pull-up resistor.

26. Specifications

- Requires 5V @ 150mA.
- RJ11C connector pinout: GND (1), +5 (2), OUTPUT to UART(3)
- Mouse OUTPUT line requires a 1K pull-up resistor at UART input
- Size: 98 x 66 x 28 (L x W x H in mm)
- 0.25 mm resolution (100 counts/inch)
- Typical acceleration: 5 G's (50 M/sec-sec) non-rotatable; 1.25 G's rotatable
- Depth of field >1 mm