APPENDIX E:

USER'S MANUAL

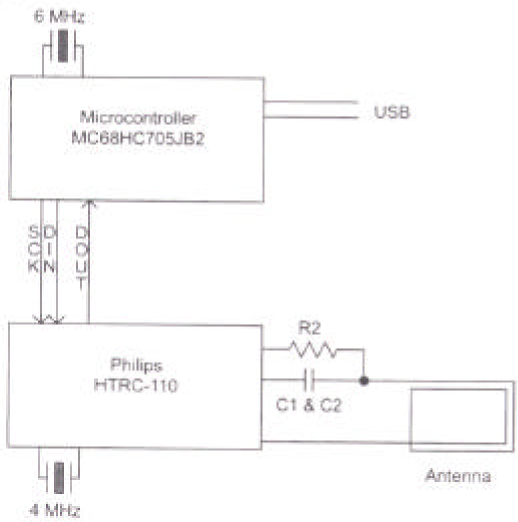
Preliminary User's Manual

1. You should have Windows 98 installed on your computer
2. Run HIDVIEW v 3.0
3. Remove and reinsert the USB device
4. The Program should recognize available USB device
5. Run **"get data from device"**
6. Run **"continuous"**
7. Place transponder on mousepad

USB P1 Mouse Pad Theory of Operation

Jeff Mock
December 15, 1998

This document describes the operation of the P1 USB mouse pad. A block diagram of the mouse pad is shown in the figure below:



Other useful documents for understanding mouse pad operation are listed below. These documents are on my website at https://www.mock.com/affinity:

- Motorola MC68HC705JB2 General Release Specification
- Philips HTRC-110 Datasheet
- Philips HTRC-110 Application Note
- Intel USB 1.1 Specification
- P1 mouse pad schematic
- P1 PCB layout
- P1 Bill of Materials
- JB2 microcode source files

USB

The Motorola MC68HC705JB2 provides a pretty complete USB interface. The JB2 generates a hard reset when the USB signals are wiggled by the host to generate a reset according to the USB spec. A USB reset will happen when the device is plugged into the host, during operating system startup, or when some upstream device (like a hub) needs to be reset.

On reset, the microcontroller initializes communication with the HTRC-110 and puts the HTRC-110 in a low power state. Most of the power consumed by the mouse pad is used to drive the antenna coil. In order to follow the USB spec the device must consume less than 10mA until a USB negotiation is done with the host and the device is given permission to consume additional power.

All USB configuration communication is done with an interrupt handler in the JB2 microcode. A relatively complex sequence of communication with the host begins after the device is hot-plugged. A variety of data structures are read and once the host computer has determined that the device can be used by the host, the device is put in the 'configured' state by host software. The JB2 responds by powering-up the HTRC-110 and attempting to read tags near the antenna.

If the device is 'unconfigured' by the host, the HTRC-110 is put back in a low power state and tags are no longer read.

All configuration information is communicated through USB endpoint-0 and special configuration cycles. Once the mouse pad is configured, tag information is sent to the host on endpoint-1. The tag communication does not require any interrupt level support.

The JB2 microcontroller has 2k bytes of program memory and 128 bytes of RAM. About 1k bytes of program memory is used for USB configuration, the remaining 1k is used for tag code.

Antenna Control

The Philips HTRC-110 is used for controlling the antenna used for powering and communicating with tags. The device is used in the manner described in the Philips HTRC-110 datasheet and application note. These documents need to be read to fully understand this part.

The HTRC-110 communicates with a microcontroller through a 3-wire interface similar to I2C. All communication is done serially through this 3-wire interface. For 125kHz tag operation the device needs a clock of 4, 8, or 12MHz. Unfortunately, the USB microcontroller needs a 6MHz clock for proper operation so it's not possible to use a single crystal for both the microcontroller and HTRC-100. As a result, there are two ceramic resonators on the board.

The HTRC-110 drives a serial resonate circuit formed by the antenna coil and C1/C2. This circuit needs to be tuned to resonate at 125kHz. There are two capacitors in parallel to ease component selection for choosing component values to make the circuit resonate at 125kHz. Different PCB fabs, thickness or material changes might necessitate different component values for C1/C2 to achieve resonance.

When the HTRC-110 is running, the voltage across the antenna might be as high 50-volts peak-to-peak sine wave at 125kHz. The DC resistance of the antenna is about 25 ohms. The current through the antenna is about 135mA (RMS). The total power consumption of the mouse pad when the antenna is powered is about 150mA. The microcontroller informs the host that it requires 200mA, the host will only configure the device if it determines that it can provide 200mA to the device, otherwise an error message is printed to the user. This should not be a problem except in rare cases where the mouse pad is plugged into a bus-powered hub that is also powering a number of high power devices.

Tag Reading

In normal operation the microcontroller instructs the HTRC-110 to supply a constant 125kHz drive to the antenna. This will power any nearby tag and the tag will begin transmitting some information. In normal operation the HTRC-110 is put in a mode to demodulate any data received from the antenna and place the data on the DOUT pin of the HTRC-110. The HTRC-110 makes no attempt to decode this data or recover any timing information, this is done by firmware on the JB2. When investigating unknown tags it's often useful to monitor the DOUT pin of the HTRC-110 with a logic analyser to determine the type of data being sent from a tag.

An exception to this normal state is that the HTRC-110 demodulator occasionally needs to be recalibrated to determine the optimal sampling point on the antenna signal for received data. In the absence of a tag, the JB2 will recalibrate the HTRC-110 every 27 seconds. This will usually have no affect on tag reading. The pad is recalibrated in the way described in the Philips applications note.

The JB2 firmware looks for a stream of 64T manchester encoded date. Pulses that are too long or too short reset the state machine. When a tag is not present the HTRC-110 generates noise that is usually easy to recognize and avoid false tag reading. On occasion the JB2 will falsely read a tag from the noise coming from the HTRC-110. The JB2 does not check parity information from tags because it doesn't have enough space in its tiny program memory for this function. So, on occasion the mouse pad will report a false tag to the host and the host must check the parity information and discard the packet if appropriate.

Encoded data from tags are measured in T units. A T unit is a cycle of the antenna frequency, 125kHz or 8uS. A 64T bit-time means data cells that are 512uS. The JB2 is simultaneously looking for two types manchester encoded streams corresponding to Marin V4050 tags or H4100 (Hitag-2) tags. The Marin V4050 tags are easiest to recognize because packets begin with a particular sequence of pulses not found in normal Manchester data. H4100 tags are a little more difficult to recognize because a packet is indicated with a stream of 9 manchester 1's. There is a bit of an ambiguity in that 9 1's will never appear in a H4100 packet except in the header, but might occur in a V4050 tag. The ambiguity is avoided by designing the state machine to conditionally accept H4100 encoding unless a special V4050 header sequence is seen in the subsequent data stream.

H4100 tags are simple in that they continually repeat a sequence of bits laser trimmed onto the tag. V4050 tags are more complex because once the tag is recognized, the JB2 must modulate the antenna with special signaling containing a command to read the serial number and other data from the tag

Miscellaneous Bits

The mouse pad is a bus powered device, it gets its power from the upstream USB device. As a result, the mouse pad may run on a Vdd as low as 4.2 volts considering voltage drop of the cables, etc. It is difficult to construct a case where Vdd will be this low, but it is possible according to USB rules. The HTRC-110 is specified to run on 5v - 10%. Experimentation has shown good performance down to 2.7v, and Philips will verbally agree to low voltage operation, but there is a small risk for a future problem.

Most RFID applications use a single crystal for both the HTRC-110 and the microcontroller. This allows precise timing of bit cell times against the antenna frequency. This application uses two separate resonators that may differ as much as 0.5% from specified values. This has presented no problems so far, but there might be some implication in the future.

Most of the power in the mouse pad is consumed by current in the antenna. When signaling to V4050 tags the field is switched on and off. The dynamic current changes make the 10uF tantalum decoupling capacitor necessary to prevent large changes on Vdd.

The 3.3v output of the JB2 (pin-16) had 100mV of 1MHz noise, probably from the internal switching supply used to generate the 3.3v reference. C8 was added to bypass the 3.3v supply and clean-up its output. Without the decoupling capacitor the noise from the 3.3v pin cause unreliable USB operation. Motorola also agreed that a decoupling capacitor was necessary here.

The resistor R4 is used to lower the Q of the antenna tuned circuit. With a PCB antenna, the resistance of the coil is about 25 ohm which was sufficient to lower the Q of the circuit to a recommended value (9 on prototype boards). As a result, R4 is a 0-ohm jumper. On boards using wire antennas it might be necessary to increase the value of R4 to 10-ohms or so.

The board is layed out for both SOIC and DIP versions of the JB2. This is just for component availability issues.

R6 is used to start the ceramic resonator on the JB2 but experimentation has shown that it's not necessary, the internal resistance on the JB2 is low enough for reliable operation so this part is not stuffed.