mpf\_utils\_rac - A Test/Utility ROM for the Multitech Micro-Professor

by Robert Coward

--------------------------------------------------------------

This utility ROM has been specifically developed for the Multitech Micro Professor MPF-1A or MPF-1B, and offers a selection of useful test routines to fully check out the integrity of the major

components on the board, as well as some interesting demonstration

routines. Since the full source code and listing file are provided,

this can be used as a basis for developing your own software.

The supplied EPROM should be fitted into the Expansion ROM

socket (U7), located between the Monitor ROM (U6) and RAM (U8).

No modification from the base configuration of the board should be required. However, if the U7 socket was previously occupied by a RAM chip or TMS2732 (which is not compatible with a normal 2732), some jumper adjustments may be required - see section 4.1 (page 35) of the MPF-1 User Manual.

If you need to program an EPROM from the supplied mpf\_utils\_rac.hex file, you should select a compatible device. If your board has a 24 pin socket for U7, the EPROM may be a 2716 or 2732 device; if it has a 28 pin socket for U7, you may also use a 27C64 device. In all cases, you should configure your programmer to fill all unused locations with 00 hex. This will ensure that the checksum reported by the relevant test should match the checksum reported by the EPROM programmer software. Note that the hex file is offset from 2000H, since this is the start of the utility ROM in memory; most programmer software should allow you to specify a negative offset to load the hexfile into the device memory starting from zero.

IMPORTANT! This utility ROM makes extensive use of the Micro-Professor MPF-1 Monitor functions, and is ONLY compatible with the standard MPF-1A or MPF-1B monitor. If you have a non-standard monitor EPROM fitted, this utility ROM may not function correctly.

The utility ROM provides the following routines for the user to access. You can execute them from the keyboard, using the address codes specified below:

2080 : BaseMonitorRomChecksum - This performs a computation of the 16 bit checksum for the base Monitor EPROM (address range 0000 - 07FF), and with the standard monitor fitted, should display "5C00".

2084 : TopMonitorRomChecksum - This performs a computation of the 16 bit checksum for the top part of the Monitor EPROM (address range 0800 - 0FFF), and with the standard MPF-1B Basic monitor fitted, should display "AFCD". If you have an MPF-1A (without Basic), or some other non-standard monitor, a different value will be shown.

2088 : FullMonitorRomChecksum - This performs a computation of the 16 bit checksum for the full range of the Monitor EPROM (address range 0000 - 0FFF), and with the standard MPF-1B Basic monitor fitted, should display "0BCD". If you have an MPF-1A (without Basic), or some other non-standard monitor, a different value will be shown.

208C - ExpansionRomChecksum - This performs a computation of the 16 bit checksum for the Utility EPROM (address range 2000 - 27FF), and the displayed value should be "E331". This assumes that unused locations not specified in the hex image are programmed to "00".

2090 - RamAddressTest - This test checks the integrity of the

main RAM (U8) address lines, by writing an incrementing value

to each address boundary, effectively walking a 1 across the address

lines; a value of zero is written to the first location. Once all locations have been written, they are read back and verified for correctness. If the test passes, then PASS is displayed; otherwise the offending address is displayed in the address field with the data display alternating between the read data and expected data (DP lit). Since stuck address lines will alias back to zero, it is likely that the first address location will contain the number corresponding to the failed address line. If two address lines are tied together, then the alias will occur at the higher address boundary and a non-zero address location will be reported.

2094 - RamPatternTest - The test operates like the RAM test described in the Micro Professor user guide (section 6 page 59), non-destructively testing ALL RAM locations, but will display clearer indications. If the test passes, then PASS is displayed, otherwise the failed address is displayed and it then cycles between data displays showing the actual and expected data (DP lit). The test will check that all bits of all RAM locations can store either 1 or 0, but will NOT detect address line faults and may not detect obscure faults that only show up with certain patterns. Note that since the RAM test is non-destructive, it can safely be run on all locations, including the stack and monitor area.

2098 - KeyCodeDisplay - This is a comprehensive keyboard test that

checks that all keys on the keyboard are working except the bottom two left hand keys (INTR and USER KEY). The data display shows a hex value of the key that was last pressed. The address display counts the keys pressed, so you can detect any keys that are bouncing; if the counter increments by more than 1 each time a key is pressed, it is bouncing excessively. When the test is started, the data display should show "--", indicating no key press detected so far. This test can also detect keys which are bouncing closed without being pressed - gently shocking the board will cause such keys to register as pressed keys, updating the display.

209C - KeyCodeDisplayPos - This is a comprehensive keyboard test that checks that ALL keys on the keyboard are working including the left most column of keys (RESET, MONI, INTR and USER KEY). The data display shows a hex value of the position code of key that was last pressed. The address display counts the keys pressed, so you can detect any keys that are bouncing; if the counter increments by more than 1 each time a key is pressed, it is bouncing excessively. When the test is started, the data display should show "--", indicating no key press detected so far. The USER KEY will cause a "UK." display to be shown whenever it is pressed or held down. The MONI and INTR keys will result in the Monitor being entered; pressing the GO key should resume execution of the test. The RESET key will of course reset the Micro-Professor back to the monitor.

20A0 - TestAllDisplaySegments This test directly drives the displays

to illuminate every single display segment and decimal point. As well as checking that they are all working, it can show up any variation in

brightness across the display, for example a weak LED segment.

20A4 - TestEachDisplaySegment- This test directly drives the displays

to illuminate each display segment and DP in turn; only one segment or

DP on a single digit is lit at any one time, so any display scanning faults can be detected.

20A8 - IntDrivenClock - This is based on the interrupt driven clock from Experiment 15 in the MPF-1 Experiment Manual, but has been significantly enhanced. On startup, displays all zeros, and allows the user to set it by pressing the number keys - this shifts in the starting count across the display Once the GO key is pressed, the clock starts running. Note that there is no protection against entering "invalid" hours, minutes or seconds, but the system will cope with these, and they will self-correct eventually. Also note that it is not particularly accurate, but still serves as a useful demonstration. Note that the Z80 CTC must be fitted in socket U11 on the board for the clock to operate.

20AC - IntDrivenCountDown - Countdown timer, with initial time setting from the keyboard. On startup, displays all zeros, and allows the user to set it by pressing the number keys - this shifts in the starting count across the display Once the GO key is pressed, the timer starts running. Note that there is no protection against entering "invalid" hours, minutes or seconds, but the system will cope with these, and they will self-correct eventually. Once the countdown has reached zero, will emit repeated beeping tones. Note that if the countdown timer is started and GO immediately pressed, the beeping will start immediately. Note that the Z80 CTC must be fitted in socket U11 on the board for the countdown timer to operate.

20B0 - TelRingTone - A really noddy demonstration copied from Experiment 16 in the MPF-1 Experiment Manual, producing a speaker tone similar to an international telephone ring tone.

20B4 - TelRingUKTone - A really noddy demonstration based on Experiment 16 in the MPF-1 Experiment Manual, producing a speaker tone similar to a UK telephone ring tone.

20B8 - TelRingBusyTone - A really noddy demonstration based on Experiment 16 in the MPF-1 Experiment Manual, producing a speaker tone similar to a telephone busy tone. This happens to be the same tone as produced by the countdown timer when it reaches zero.

20BC - ElectronicOrgan - Another fairly noddy demonstration program copied from Experiment 17 in the MPF-1 Experiment Manual. Note that all 16 keys on the main keyboard are implemented (only the far left hand column keys do not generate tones). This has been enhanced to briefly display the key code whenever a key is pressed (hardware limitation preclude the display being refreshed while the tone is being generated).

20C0 - MusicBoxJingleBells - Another fairly noddy demonstration program copied from Experiment 18 in the MPF-1 Experiment Manual. This plays a very short version of "Jingle Bells", fairly badly!

20C4 - MusicBoxGreenSleeves - Another fairly noddy demonstration program copied from Experiment 18 in the MPF-1 Experiment Manual. This plays a longer version of "GreenSleeves", not quite so badly.