

NEW NMI MENU FOR MB02+ SYSTEM

```
Device:disk    128K Page: 16      22:25:48
@ 42  $ 0      Master:  Slave:WP

I- AY,FDC,DMA,floppy      O- change disk
E- 128K reset             P- change dir
R- 48K reset              C- cat
B- warm start             Z- save .280 snap
N- CLEAR #5FFF:NEW        X- load .280 snap
S- show screen            A- setup
D- device (tape/disk)     U- run file
M- mrs monitor            V- run from path
L- run devastace+         W- save mem.chunk
0-g- 128K pages           Y- load to memory
F- load .SNA snap         H- CD player
G- save .SNA snap         J- WP master
Q- quit                   K- WP slave

.....
```

Last version: 1.33h

Date of release: 31.03.2008

What's new:

- added EasyHdd write protect
- CLS of the screen after .280 snap is loaded (previous versions had chaos on the screen after loading)

Author: Hood, email: hood@znojman.cz

<http://hood.speccy.cz>

Tel: +420-777-192-191



CONTENTS

<i>Foreword.....</i>	<i>3</i>
<i>What's in package, instalation, setup.....</i>	<i>4</i>
<i>Functions- basic overview.....</i>	<i>5</i>
<i>Commentary to some functions</i>	<i>6</i>
I reset AY, FDC, DMA, floppy	6
B warm BASIC start	6
N CLEAR #5fff: NEW	6
M jump to MRS debugger	6
0-9 switch among 128k banks	6
S displays and saves VRAM	6
L devastace+ debugger (VRAM version).....	6
F,G load/save .SNA snap 48k/128k (see key A).....	6
A setup.....	7
U run file from a current path	7
V run file from a default path	7
ss displays the date.....	7
J,K EasyHdd write protect master/slave	7
<i>History</i>	<i>8</i>
<i>Acknowledgements</i>	<i>10</i>

FOREWORD

Dear followers and users of MB02+ disk system. In your hands I am humbly giving my creation- nmi menu for this system. First of all, I must say I am very proud for that creation, mainly because I am not considering myself as a programmer and mathematical way of thinking is something I am not familiar with. I created the program, and hopefully continue working on it, from my love, if I may call it so, towards ZX Spectrum. Programming in the machine code is immensely creative activity and in this sense also fills up masculine creative needs.

Origination to the new nmi menu gave one Omega's diskette forgotten on the table at Shucon. There was a .SNA and .Z80 runner downloaded from somewhere on the internet. But the program did not work and it was on that time I put myself a question why MB02+ cannot work with snaps, despite being such an outstanding disk interface. I also remembered a standard nmi menu which was great but because of a lack of space in BSROM could not be extended anymore. And so I thought for myself I wanted to be able to work with snaps, in order to finish my favourite one Where Time Stood Still 128:).

Need to point out, that as for functionally, I started with Busy's standard nmi menu which you may find in BSROM118, mainly because users were already used to it and I also used his existing routines from BSROM118. Everything is coloured In chapter "Functions-basic overview". In black are basically Busy's work, even descriptions of functions are taken from Busy's manual for BSROM118. My nmi menu is self-contained system, it does not need ROM for the work. At the moment, it occupies SRAM pages number 99 and 100, but I may need another one in the future.

The beginning of 2008 was paramount, I was to return to ZX Spectrum for the second time. Or better to say, I never abandoned it, so I rather set myself from passive to active mode:). I knew the last version I created (ver. 1.31n released in April 2003) was good but I still owed it several new functions, improvements and fixes. By the end of 2007 I set heart upon restoring to work on my programme. The source code was written in Prometheus assembler and I stored it on my hdd (attached to MB02+). By the end 2007 I returned to the scene and wanted to continue working on it. I found out, though, the hdd did not respond. After unavailing attempts to resuscitate the dead hdd (hdd electronic exchange, professional companies would save the data which would however cost me thousands of crowns) I decided to restore the source code based on my last saved source code (ver. 1.31a), thus I had to overcome 13 saves from 1.31a to final and restored 1.31n. Fortunately, the changes took place only in load/save snaps routines, thus the difference between 1.31a and 1.31n was not vast. Finally, everything went well and after couple of days of discontinuous work, fully restored 1.31n version was here (of course, it was restoration without commentaries). And because I have not forgotten what I wanted to improve on nmi menu, I started with RTC support. Many thanks go to Velesoft for help by the 1.31n restoration.

WHAT'S IN PACKAGE, INSTALATION, SETUP

In the package you will find:

- this pdf file in Czech and English
- MBD image for Real Spectrum emulator for PC
- TAP file for direct usage on ZXS
- source code for Prometheus assembler. IN CASE YOU ARE GOING TO CHANGE SOMETHING IN MY NMI MENU, LET ME KNOW, PLEASE!!!

Installation:

- directory 0 on .MBD image. Install using NEW command for file "nmi_1.33h".
- you will find "nmi_1.33h" also as a .TAP file for direct use on real ZXS. Install in the same way.
- default path (see key V) is @3\$0. If this path does not suit you, you may poke your own before installation. Before installation load the code file into memory, then POKE 27184, disc and 27185, directory, and then save the file back to disk with the same length and starting address. This ensures each installation to have your own path for file running.

FUNCTIONS- BASIC OVERVIEW

Keys:

I	reset AY, FDC, DMA, floppy
E	128k reset
R	48k reset
Q	quit nmi menu
B	warm BASIC start. Initialisation of all BASIC variables except for PROG, VARS, ELINE (i.e. Basic program as well as Basic variables remain preserved)
N	executes CLEAR #5fff: NEW. I.e. the memory above #6000 remains intact
M	jump to MRS debugger (must be present in the memory). All registers are set like they were in the moment when nmi button was pressed. (PC and SP are correctly set only when SP was not changed to #5800).
0-9	switch among 128k banks (8, 9= displays the second VRAM)
S	displays VRAM, pressing ENTER will save the screen to disk (extension 3, i.e. the saved file can be loaded directly from Basic).
D	tape/disc device selection
L	runs debugger devastace+ (VRAM version) with all registers, SP and PC set as they were in the moment of pressing the nmi button. SS+Q to return back to nmi menu.
F	load .SNA snap 48k/128k (see also key A)
G	save .SNA snap 48k/128k (see also key A)
O	disc change
P	directory change
C	disc catalogue
Z	.Z80 snap save (inactive for now)
X	.Z80 snap load (128k version only)
A	setup, snap type selection, and default path selection (see key V)
U	run file (enter file number, runs from a current path)
V	run file (enter file number, runs from a default path set in setup- established for frequently used and run files)
W,Y	load/save of memory chunk
H	CD player (not functional)
ss	displays date, clock is running in right upper corner
J,K	EasyHdd write protect master/slave

black)	author Busy soft- original nmi menu (sparkling red border)
blue)	contained in Busy's menu, but modified by Hood
red)	utterly new functions- author Hood

COMMENTARY TO SOME FUNCTIONS

I reset AY, FDC, DMA, floppy

Resets the following peripherals: AY, FDC, DMA and a diskette drive

B warm BASIC start

Initialisation of all BASIC variables except for PROG, VARS, ELINE (i.e. Basic program as well as Basic variables remain preserved)

N CLEAR #5fff: NEW

executes CLEAR #5fff: NEW. I.e. the memory above #6000 remains intact

M jump to MRS debugger

(must be present in the memory). All registers are set like they were in the moment when nmi button was pressed. (PC and SP are correctly set only when SP was not changed to #5800).

0-9 switch among 128k banks

(8, 9= displays the second VRAM), after nmi button is pressed, the program detects currently attached bank and displays the its number.

S displays and saves VRAM

pressing ENTER will save the screen to disk (extension 3, i.e. the saved file can be loaded directly from Basic).

L devastace+ debugger (VRAM version)

jumps into devastace+ debugger with all registers, SP and PC set as they were in the moment of pressing the nmi button. SS+Q returns you back to nmi menu. Does not corrupt a single byte form RAM, a user can trace programs, browse in RAM, etc.

F,G load/save .SNA snap 48k/128k (see key A)

128k .SNA load/save- working well on speccy but a very few snaps from PC does not work on ZXS and the problem is not on my side, I think. IMPORTANT!!! If you want to transfer .SNA from ZX to PC, most probably you will have to increment twice the double-byte on offset 23. There is stack pointer which is on 128k version on PC twice incremented. On the other hand, if you take snaps .SNA from PC to speccy, you do not have to bother with no such changes, nmi program can handle this fact.

- save may not work 100%- need to be fixed

X load .Z80 snap (128k version only)

.Z80 load- key X- does not work 100% even though most .Z80 snaps will run. No 48k type support!

A setup

setup- for the purpose of several functions I introduced a small setup, where the user can set several parameters: snap type selection, path selection (see also key V)

- key Z- .Z80 snap type for save (version 128k or 48k). Actually, does not affect anything as the .Z80 save function is not activated.
- key G- .SNA type (128k/48k) for save. Default is 128k.
- key O- default drive and directory for running files function.
- key Q- back to main menu.

I think, introducing the setup menu is not very user friendly. We may wait for reactions, maybe I will solve the setup parameters without setup menu.

U run file from a current path

runs the file from a current path (thanks to Shrek and Sweet)

V run file from a default path

same as key U, but the files are run from the default path set by user in setup. It means, that the user can save his most favourite and run programs on the preferred disc and directory and run them easily using this function.

ss displays the date

displays RTC date and time. Clock is running in the upper right corner. Pressing SS will display date- works same exactly as in MB-Commander. If you are still not using MBC, start now:)

J,K EasyHdd write protect master/slave

EasyHdd write protect for master and slave IDE devices together with simple IDE detection. Nmi menu gives you information about currently connected IDE devices and if it is HDD/CF card. If it is another IDE device, information is not displayed. If you have CD/HDD connected, on the screen you will see MASTER, SLAVE texts in white colour. Furthermore, if you have EasyHdd by LMN (most probably 99,9% of all users operating IDE on their MB02+ machines, visit www.8bc.com/download), the texts are either in green (write protect disabled) or in red with "WP" (write protect enabled). However, also there can be green Master and white Slave meaning you have installed EasyHdd for master and that Slave was only connected after EasyHdd installation. Thus you may operate EasyHdd on master and anything else on slave (supposed that someone has written other operating system).

HISTORY

ver. 1.33h (released 31.03.2008)

- keys J,K- added EasyHdd write protect for master and slave together with simple IDE device detection
- CLS of the screen after .Z80 snap is loaded (previous versions had chaos on the screen after loading)

ver. 1.31x (released 23.03.2008)

- displays RTC date and time.
- written ldir routines for comfortable assembly of all nmi menu blocks together into one CODE file (not substantial for a user)

ver. 1.31n (released April 2003)

- 128k page detection
- "raining" effect removed- I register filled with 0 (for ZXS 128+ machines, thanks to POKE)
- user setup
- running the files from current path
- running the files from default path
- load/save of .SNA format- 128k version
- load .Z80 snaps 128k version
- because of better clarity some lines on the screen were inverted if a function is chosen.
- during character deleting some chaos was left on the line- corrected
- key routine- corrected wait key delay- the keyboard is working much better compared to ver.1.27.

ver. 1.27 (released 2002)

- save of a memory chunk to disc
- load of a file of any type into memory (i.e. any extension type)
- jump into MRS monitor works now 100%

ver. 1.14 - (released 17.4.2002- first release of the nmi menu)

The first release of nmi menu has the same functions as Busy's menu but has even some add-ons. For user it is the same, no keys were changed, only the key S is saving in a way, that only when S is pressed the screen will appear, and pressing the ENTER then saves the screen to disc (the extension is 3, i.e. you are then able to load the file normally from BASIC). Any other key will return you back to menu. On the screen accrued new info about device (TAPE/DISK), about 128k page and a current @ and \$.

- disc catalogue
- disc and directory selection
- jump to devastace+ monitor

Known problems in ver. 1.14:

1) menu stability. This is a big problem. I explain why. The aim was to improve the menu, therefore I use my own stack which is placed in ROM (otherwise I had to corrupt VRAM or RAM). Thus I write-enabled ROM. On the original stack only return address and AF register is kept. So, after nmi button is pressed, the program still in ROM stores registers and SP and then jumps into menu itself (in this version it is page 99, in next versions the user can choose to which page nmi menu will be installed).

Also version 1.14 is stored only in one SRAM page. In next versions it will be two pages (VRAM, devastace+, CD player, work with snaps- it all cannot be hold in 16kB page).

Back to stability- to keep it short, the 99 page content is overwriting very often and so the menu must be installed fairly often

2) jump into MRS monitor- works only partially

3) boot screen- (a process right after turning on computer- a moving diskette on the boot screen). If you press nmi button to enter the menu and then leave the menu with Q, you are not back at the booting process, because entering the nmi menu the service GETCST touches the current @ and \$ and resets the boot drive.

ACKNOWLEDGEMENTS

I would like to thank the people who helped me with works on nmi menu a lot with their advices, ideas or routines (in random order)- Tritol, Busy, Sweet, Shrek, Logout, Velesoft, Poke (hope, I did not miss anybody) – without them the program would not be so far as it is today.