Hi,

\*\*I was very curious, so I looked into Docker and QL and got somewhere, quite far, but it's a pain because the QDOS-GCC compiler from the Docker for Windows crashes already when reading the file, so I couldn't use Docker directly from Windows - so I made a temporary folder d:\vmex-tmp and in win wsl2 (first it was necessary to upgrade from wsl1) then another folder for compilation as /home/usrname/*vmex ... in both these folders there is also a src subfolder, which is all copied before compilation from d:\ to /home/.... then it is compiled, then the owner is overwritten (because I log in to this image as root, a pig, sure, I don't know how else) and then the compiled binary is copied from /home/... back to the temp folder on d:/ .... this is the first step, compilation (a professional would make a more generic script out of it, at least for simple things, I'm not good at Linux) .... and the next step is a small detail is QL SuperBasic conversion of that binary to QDOS EXE for which I made bin2exe\_bas after reading a lot of things (which should have been generic but somehow I can't get those string variables into FLEN, FDAT, where there is still "" in front of them ... I don't know how to do it, I didn't solve it and so it's VMEX (binary) and VMEX\_EXE (executable) hardcoded, it just converts this ....... then in QL it's enough to just EXEC dos1\_VMEX\_EXE (I use QPCII emulator where I have host folder mapped to "dos1*" drive...... ufffffff damn work, it's a pain .......... unfortunately it works only with EMB\_TESTS when tests are run directly in main, that interactive monitor doesn't work, ... but if you tried to do something primitive with scanf fgets etc. you can try it .... (just put it in that empty VMEX.c so you don't have to rewrite it, or if you want to create some more generic scripts and put them on the web .... it would probably work somehow, but I don't have the nerves for it)) )))

this doesn't work with win-docker because of that bug in gcc ....

docker run -v D:/\_test:/\_test -w /\_test -u root xora/qdos-devel qdos-gcc -o hello hello.c

so on win: d:\_vmex-tmp\src .... on linux: /home/usrname/\_vmex/src ...

bash batch for linux run in /home/usrname/\_vmex :

#!/bin/bash cp /mnt/d/\_vmex-tmp/src/\* /home/falken/\_vmex/src docker run -v /home/falken/\_vmex:/\_vmex -w /\_vmex -u root xora/qdos-devel qdos-gcc -o VMEX src/VMEX.c sudo chown -R falken:falken ~/\_vmex cp /home/falken/\_vmex/VMEX /mnt/d/\_vmex-tmp/VMEX read -p "Press [Enter] to continue..."

and this is that bin2exe procedure ....

100 DEFine PROCedure BIN2EXE

110 LOCal length,start,datasp

120 length=FLEN(\dos1\_VMEX)

125 datasp=FDAT(\dos1\_VMEX)

130 start=RESPR(length)

140 LBYTES dos1\_VMEX, start

150 POKE start + 1024, 100

160 SEXEC dos1\_VMEX\_EXE, start, length, 16384

170 print "EXE is saved."

200 END DEFine BIN2EXE

yes, if you can, send me a link, but I'm probably giving up on this for now... one test found it, I found out that qdos-gcc doesn't handle stdint, so I pasted a replacement into the source, I still need to clean it up and send it to github

https://github.com/SinclairQL

\*\*otherwise, it was also simple that interactive shell ... I had detection there via target/defines and there was QL missing, and because it's not clear what the identifier should be, and it's all silly, so