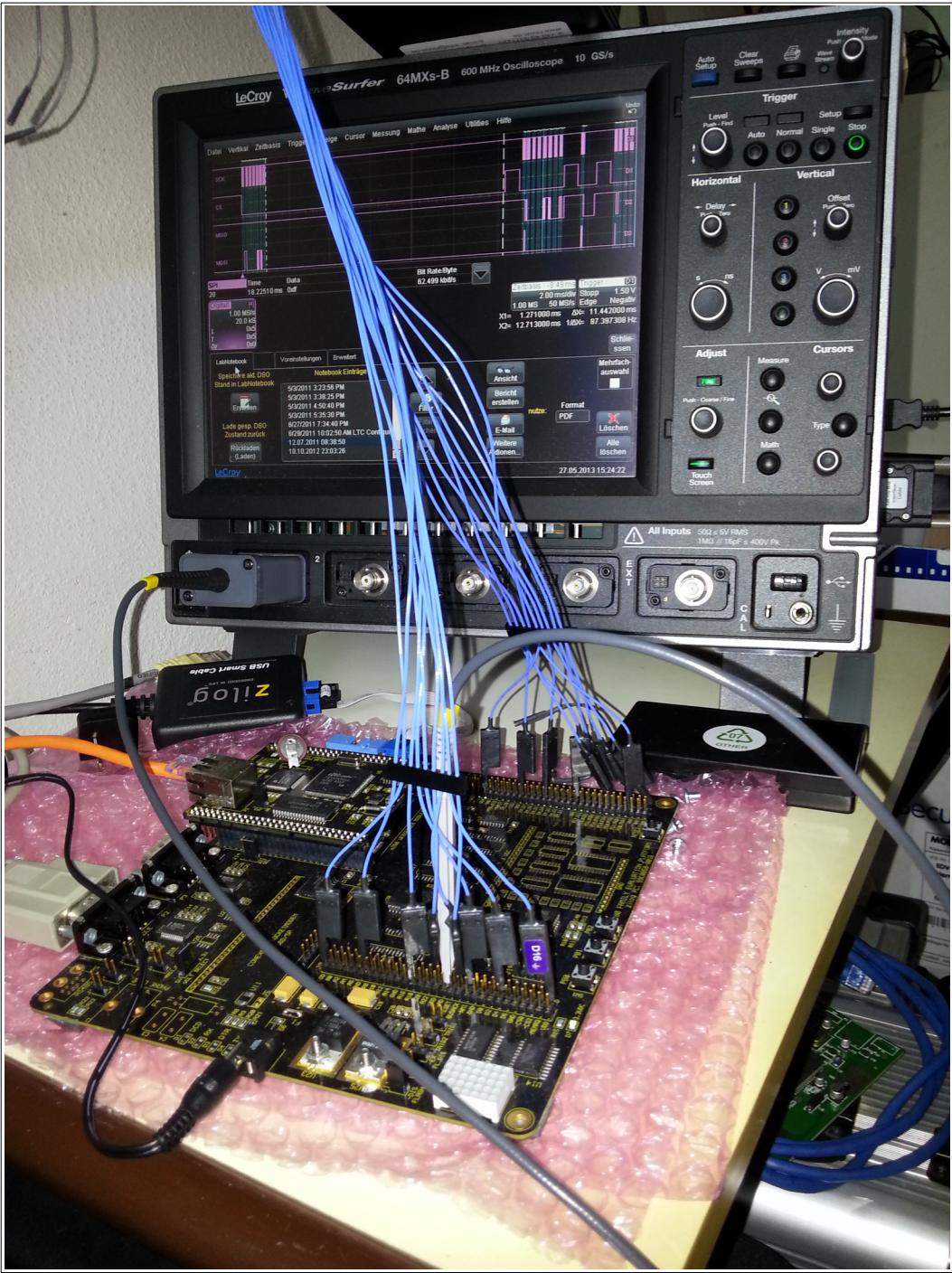


## 1.8.2023: Add this build instruction.

- 1) git clone --recurse-submodules [git@github.com-FreeRTOS:notwendig/FreeRTOS-eZ80F91.git](https://github.com-FreeRTOS/notwendig/FreeRTOS-eZ80F91.git)
- 2) cd FreeRTOS-eZ80F91
- 3) cd kernel; patch -p1 < patches/0001-ZiLOG-Acclaim-fixes.patch  
cd network; patch -p1 < patches/0001-ZiLOG-Acclaim-fixes.patch  
cd uzlib; patch -p1 < patches/0001-ZiLOG-Acclaim-fixes.patch
- 4) Run virtual windows-guest on your Linux-host. If you are using an USB-Debugge's then redirect ther USB to the guest  
Install the ZDSII v5.3.5 on your guest if not already done. And on the host connect a terminal (Putty) on the serial-port  
( for example /dev/ttyS4 ) and setup the connection to 115200 8N1 RTS/CTS.
- 5) Now you can build the libraries.  
Load the project uzlib, network and kernel one after the other into the ZDSII and compile them.
- 6) Connect the Zilog Debugger and the serial-port to your Build-PC
- 7) Load the application EZ80F910300ZCOG into the ZDSII and build and run it.
- 8) Open a Linux-Console (132x40) and enter "nc -u <the IP shown on putty> 4060". Trigger the target by enter RETURN-Key.

FreeRTOS ez80F91 Acclaim! Port - Copyright (C) 2016-2023 by NadiSoft All rights reserved

This file is part of the FreeRTOS port for ZiLOG's EZ80F91 Module.  
Copyright (C) 2016 by Juergen Sievers <[JSievers@NadiSoft.de](mailto:JSievers@NadiSoft.de)>



The Port was made and rudimentary tested on ZiLOG's EZ80F910300ZC0G Developer Kit using ZDSII Acclaim 5.3.5 Developer

Environment and comes WITHOUT ANY WARRANTY to you!

Developer:

SIE Juergen Sievers <JSievers@NadiSoft.de>

Repository directories:

- doc some documentation.
- kernel sub-module the real time kernel source code.
- network sub-module FreeRTOS-Plus-TCP/source the real TCP/IP source code.
- EZ80F910300ZC0C The Demo's header and source files.

- `uzlib` sub-module The compress/un-compress library

See <http://www.freertos.org.html> for full details of the FreeRTOS directory structure and information on locating the files you require.

## My Build- and Test-environment:

I use a Fedora Linux Workstation as host system. On this host a Windows 10 is running as guest under QEMU. On Windows I mount a samba share as drive Z: with at least the following directories.

1. **Z:\ZDSII\_eZ80Acclaim!\_5.3.5**  
ZiLOG Developer Studio II-eZ80Acclaim!® installation.  
Free download from [www.zilog.com](http://www.zilog.com).
2. **Z:\workspace\FreeRTOS-eZ80f91** Windows Project-root an SMB Mount from host  
~/ZiLOG

The Target is connected over its Serial and Ethernet port on the Linux-PC

The Debug-output uses target's serial-port 0 115200,8,1,n,RTS/CTS. I use PuTTY on the Linux-PC to display such information.

```

A 11
B 6

UT 0:1:32 332ms
UT 0:1:32 284ms

eZ80 tty
11223344->44332211->3351057p
prvIPTask started
Link .....EMAC 100MBPS, FULL_DUPLEX, Link up
prvInitialiseDHCP: start after 318767354 ticks
Socket 4040 -> 0.0.0.0:0 State eCLOSED->eTCP_LISTEN
vDHCPProcess: discover
vDHCPProcess: discover
vDHCPProcess: timeout 10000 ticks
vDHCPProcess: discover
vDHCPProcess: timeout 20000 ticks
vDHCPProcess: offer 192.168.77.24
vDHCPProcess: reply 192.168.77.24
vDHCPProcess: offer 192.168.77.24
vDHCPProcess: acked 192.168.77.24
NetworkEventHook eNetworkEvent:0
IP Address: 192.168.77.24
Subnet Mask: 255.255.255.0
Gateway IP Address: 192.168.77.1
DNS server IP Address: 192.168.77.1
Socket 4050 -> 0.0.0.0:0 State eCLOSED->eTCP_LISTEN
Looking up NTP-Server 'ptbtime1.ptb.de'
prvProcessDNSCache: add: 'ptbtime1.ptb.de' @ 192.53.103.108
DNS[0x0000]: The answer to 'ptbtime1.ptb.de' (192.53.103.108) will be stored

```

On the Linux-PC also runs a DHCP-Server which will answer to the target's DHCP Requests.

At last the ZiLOG USBSmartCable is connect between the target and Linux-PC. This USB -Device is routed to the Windows guest. It will be used by ZiLOG's Developer Studio for download, flashing and debugging.

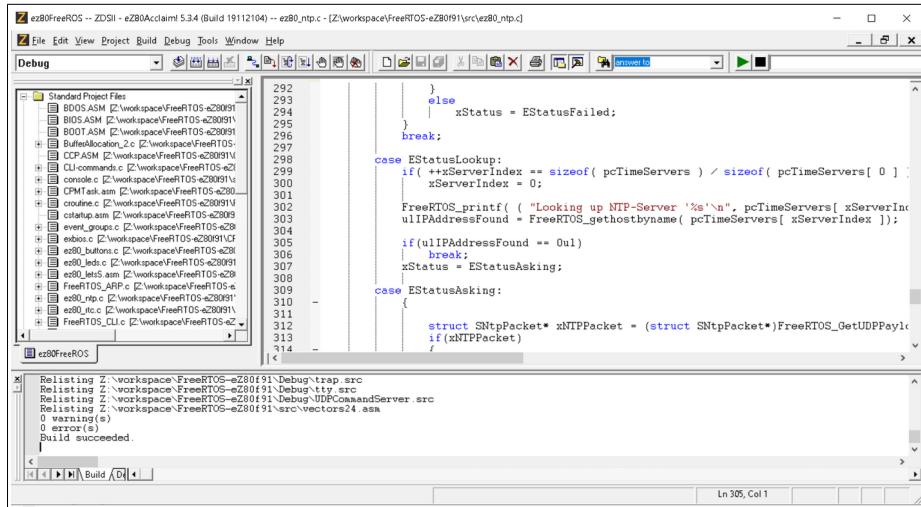
Changing to directory `Z:\workspace` (or on the linux to the shared) and clone the project <https://github.com/notwendig/FreeRTOS-eZ80f91.git> including submodules.

Apply the patch to zulib.

On the Windows-guest run the ZiLOG Developer Studio. Open the Project-file Z:\workspace\FreeRTOS-eZ8091\uzib.zdsproj and build the uzlib[d].lib.

On a linux terminal change to directory uzlib and type make to build the packer/unpacker. Change to CPM directory and type make to build the CP/M 2.2 boot-tracks and the Disk-Image as packed C-Array.

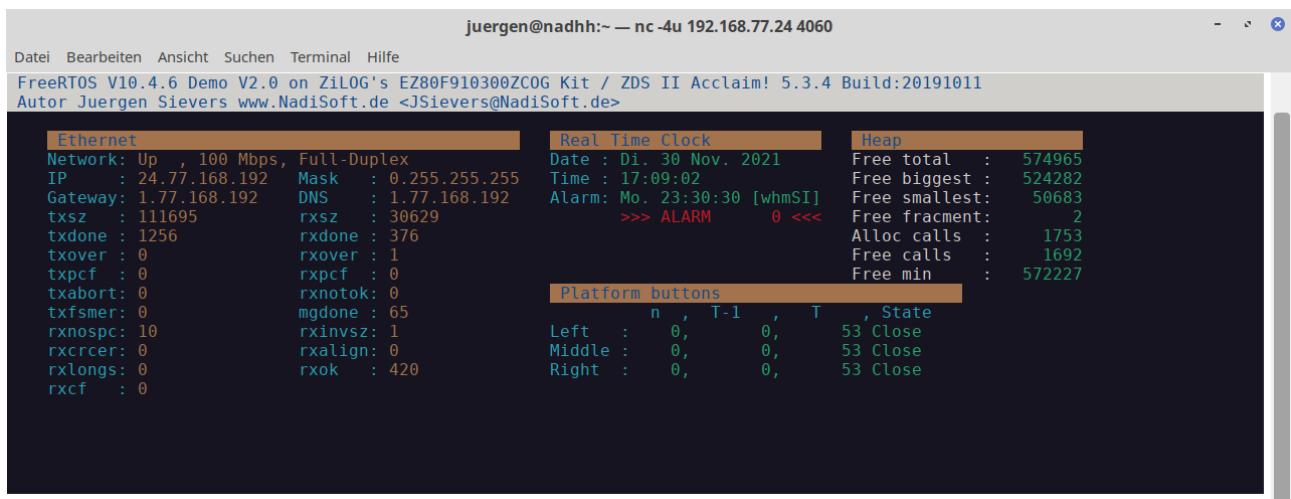
Now you are ready to open and build the z80FreeROS.zdsproj, and build it.



The screenshot shows the ZiLOG Developer Studio interface with the project 'z80FreeROS' selected. The left pane displays the file tree under 'Standard Project Files'. The main pane shows a portion of the code for 'ez80\_ntp.c'. The status bar at the bottom right indicates 'In 305, Col 1'.

Before you download and run the Demo to the Target, a terminal (PuTTY) should be running on the target's serial-port connection.

After the target has gotten it's IP you may start several other connection to the target.



The PuTTY terminal window shows the following output:

```
juergen@nadh:~ — nc -4u 192.168.77.24 4060
Datei Bearbeiten Ansicht Suchen Terminal Hilfe
FreeRTOS V10.4.6 Demo V2.0 on ZiLOG's EZ80F910300ZCOG Kit / ZDS II Acclaim! 5.3.4 Build:20191011
Autor Juergen Sievers www.NadiSoft.de <JSievers@NadiSoft.de>

Ethernet
Network: Up , 100 Mbps, Full-Duplex
IP : 24.77.168.192 Mask : 0.255.255.255
Gateway: 1.77.168.192 DNS : 1.77.168.192
txsz : 111695 rxsz : 30629
txdone : 1256 rxdone : 376
txover : 0 rxover : 1
txpcf : 0 rxpcf : 0
txabort: 0 rxnotok: 0
txfsmr: 0 mgdone : 65
rxnospc: 10 rxinvsz: 1
rxrcr: 0 rxalign: 0
rxlongs: 0 rxok : 420
rxcf : 0

Real Time Clock
Date : Di. 30 Nov. 2021
Time : 17:09:02
Alarm: Mo. 23:30:30 [whmSI]
>>> ALARM 0 <<<

Heap
Free total : 574965
Free biggest : 524282
Free smallest: 50683
Free fragment: 2
Alloc calls : 1753
Free calls : 1692
Free min : 572227

Platform buttons
n , T-1 , T , State
Left : 0, 0, 53 Close
Middle : 0, 0, 53 Close
Right : 0, 0, 53 Close
```

```
CP/M 2.2

EZ80F91 CP/M 2.2 Console (C) 2021 v1.0.1
Be patient decompressing and loading of the Ramdisk takes a while.
CP/M 2.2 Loader v1.0 .....OK
64K CP/M Vers. 2.2 / CBIOS V1.5
eZ80 port (C)1998-2021 by Juergen Sievers

a>dir
A: ASM      COM : CREF80   COM : DDT      COM : DISKDEF LIB
A: DSKMAINT COM : DUMP    COM : ED       COM : L80     COM
A: LIB80    COM : LOAD    COM : M80    COM : PIP     COM
A: SDIR     COM : SID     COM : STAT    COM : SUBMIT COM
A: XSUB     COM : ZAP     COM : ZSID    COM : BOOT    ASM
A: CCP      ASM : BDOS   ASM : BIOS   ASM
a>_
```

Regards

Jürgen