

NUC980 OpenWrt Quick Start Guide

Application Note for 32-bit NuMicro® Family

Document Information

Abstract	This document introduces how to develop with NUC980 OpenWrt
Apply to	NuMicro® NUC980 series.

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller and microprocessor based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com



Table of Contents

1	OVERVIEW	3
2	QUICK START	3
	2.1 Requirements	3
	2.2 Installation	3
	2.3 Configuration	3
	2.3.1NUC980 OpenWrt Configuration	5
	2.4 Compilation	6
	2.5 Firmware Programming	7
	2.6 Test the OpenWrt	9
	2.6.1Network Settings	9
	2.6.3Firmware Undate	12



1 Overview

The OpenWrt Project is a Linux operating system targeting embedded devices. Instead of trying to create a single, static firmware, OpenWrt provides a fully writable filesystem with package management. The NUC980 OpenWrt is based on OpenWrt/LEDE 17.01.7.

2 Quick Start

2.1 Requirements

Before installation, please make sure there is enough free space for project building. The expected used space is about 12 GB.

2.2 Installation

Refer to *NUC980 Linux 4.4 BSP User Manual* for the steps to download NUC980 Linux BSP and set up development environment. After that, please clone the NUC980 OpenWrt with following command:

\$ git clone https://github.com/OpenNuvoton/NUC980-OpenWrt.git

After download is completed, execute following commands to update the OpenWrt feeds script.

- \$./scripts/feeds update -a
- \$./scripts/feeds install -a

2.3 Configuration

The NUC980 OpenWrt supports EVB, Chili and IoT boards. Please use the correct setting for the target board. For example, to use the Chili board, user needs to follow the steps below:

In folder NUC980-OpenWrt, use the file Nuvoton/config/config_nuc980_chili as the OpenWrt configuration file.

cp Nuvoton/config/config_nuc980_chili .config

Run "make menuconfig" to configure OpenWrt. Confirm if the Target System is Nuvoton NUC980, and the Subtarget is the "NUC980 Chili", as shown in Figure 2-1.

\$ make menuconfig



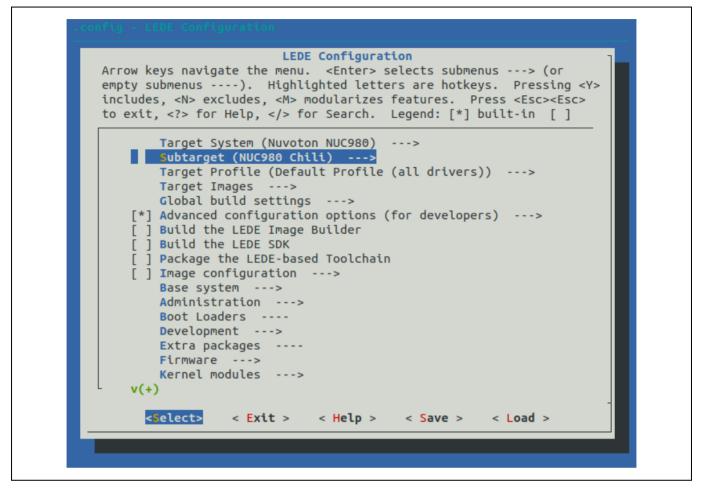


Figure 2-1 Select Target and Subtarget

In Advanced configuration options page, specify the git repository and branch to clone Linux kernel source.



```
Advanced configuration options (for developers)
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]
excluded <M> module < > module capable
           --- Advanced configuration options (for developers)
           [ ] Show broken platforms / packages
           ()
                 Binary folder
           ()
                 Download folder
                 Local mirror for source packages
           [*] Automatic rebuild of packages
               Build suffix to append to the target BUILD_DIR variable
Override the default TARGET_ROOTFS_DIR variable
           [ ] Use ccache
                 Use external kernel tree
           (git@github.com:OpenNuvoton/NUC980-linux-4.4.y.git) Enter git repository to clone
                  Enter path to local reference repository
           (master) Enter git branch to clone
           [ ] Enable log files during build process
                 Enable package source tree override
           (-fno-caller-saves -fno-plt) Additional compiler options
                 Target Options
                 Use external toolchain ----
                  <Select>
                              < Fxit > < Help >
                                                       < Save >
                                                                   < Load >
```

Figure 2-2 Configure Kernel Repository

The default setting uses the SSH method to clone git repository. Please make sure the public key of computer is attached to user's git account. If not, user can change to use the HTTPS method to clone. Table 2-1 lists available NUC980 Linux kernel repositories of using the HTTPS method.

Repository Manger	URL
Github	https://github.com/OpenNuvoton/NUC980-linux-4.4.y.git
Gitlab	https://gitlab.com/OpenNuvoton/NuMicro-ARM7-ARM9- Family/NUC980-linux-4.4.y.git
Gitee	https://gitee.com/OpenNuvoton/NUC980-linux-4.4.y.git

Table 2-1 NUC980 Linux Kernel Repositories

Run "make kernel_menuconfig" to download NUC980 Linux BSP and related packages. This step may take 20 minutes. Then user can exit and save the config directly.

\$ make kernel_menuconfig

2.3.1 NUC980 OpenWrt Configuration

This section introduces how the NUC980 OpenWrt configuration is generated. The following steps show how the configuration file Nuvoton/config/config_nuc980_iot is generated.



 Specify the target and subtarget of NUC980 IoT board, and then run "make defconfig" to generate the OpenWrt default configuration. After this step is done, the OpenWrt configuration is stored in the file .config.

```
echo CONFIG_TARGET_nuc980=y > .config
echo CONFIG_TARGET_nuc980_nuc980_iot=y >> .config
make defconfig
```

2. After the default configuration is generated, run "make menuconfig" to modify the configurations. The necessary steps for NUC980 IoT board are shown in Figure 2-3.

```
[*] Advanced configuration options (for developers) --->
   (git@github.com:OpenNuvoton/NUC980-linux-4.4.y.git) Enter git repository to clone
   (master) Enter git branch to clone
   [*] Toolchain Options --->
     Binutils Version (Binutils 2.27) --->
     GCC compiler Version (gcc 6.x) --->
     C Library implementation (Use glibc) _--->
Boot Loaders --->
  <*> uboot-nuc980-iot...... U-Boot for IoT board
Libraries --->
  SSL --->
     <*> libopenssl...... Open source SSL toolkit (libraries) --->
     <*> luci-ssl...... LuCI with HTTPS support (mbedTLS as SSL backend)
   3. Applications --->
     <*> luci-app-advanced-reboot...... Advanced Linksys Reboot Web UI
   4. Themes --->
     Utilities --->
  Boot Loaders --->
      < > uboot-envtools...... read/modify U-Boot bootloader environment
```

Figure 2-3 Modified Items to Generate NUC980 OpenWrt Configuration

3. After the configuration modification is completed, exit the setting and rename the file .config to Nuvoton/config/config_nuc980_iot.

2.4 Compilation

After configuration is completed, run "make" command to build the OpenWrt. The building may take around 90 minutes.

\$ make

The generated images are located in the bin/targets/nuc980/-glibc/ directory. The output files are shown as Table 2-2. Please note for the OverlayFS Root file system, the Chili board (SPI-NOR) uses the Squashfs+JFFS2, but the EVB (NAND) and IoT board (SPI-NAND) use the Squashfs+UBIFS by default.

Image Name	Description
	OpenWrt firmware image, includes Linux kernel ulmage, and OverlayFS root filesystem.



\${Subtarget}-squashfs-firmware.bin	
lede-nuc980-\${Subtarget}- \${Subtarget}-squashfs-sysupgrade.bin	OpenWrt system upgrade image.
\${Subtarget}-v1.0.dtb	Device tree image.
\${Subtarget}-u-boot.bin	Main U-Boot loader.
\${Subtarget}-u-boot-spl.bin	Load main U-Boot from NAND Flash to DDR (for EVB and IoT boards).
\${Subtarget}-u-boot-env.txt	U-Boot environment file.

Table 2-2 OpenWrt Generated Images in Output Directory

If any error occurs during compilation, use the following command to collect error log for further check.

\$ make -j1 V=sc

2.5 Firmware Programming

Note that U-Boot can also be built through the NUC980BSP rather than OpenWrt. Each target board should use its own U-Boot firmware and U-Boot environment

When the target board uses the NAND related device as the storage media, such as EVB and IoT boards, user should do the "erase all" action first before programming the firmware.

Refer to *NUC980 NuWriter User Manual* to program U-Boot, U-Boot environment, device tree and firmware images to the target storage media, as shown in Figure 2-4 (for Chili board) and Figure 2-5 (for IoT board).



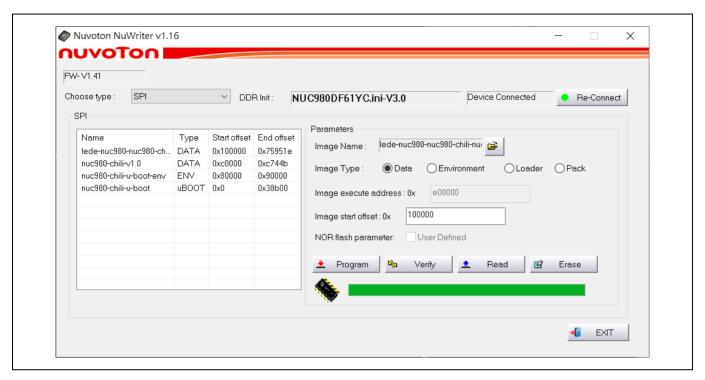


Figure 2-4 Use NuWriter to Program Images for Chili Board

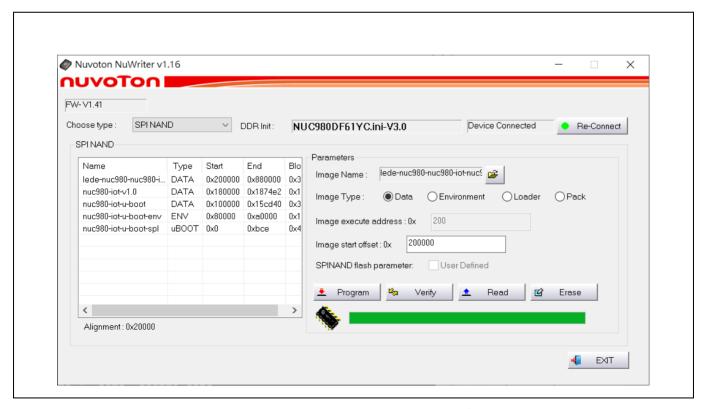


Figure 2-5 Use NuWriter to Program Images for IoT Board



The MTD partitions are defined in U-Boot environment. For the Chili board, user can see there are three MTD partitions as follows.

bootargs=noinitrd rootfstype=squashfs,jffs2 console=ttyS0,115200n8 rdinit=/sbin/init mem=64M mtdparts=m25p80:0x80000@0x0(u-boot),0x10000@0xC0000(dtb),-@0x100000(firmware)

For the IoT board, user can see there are also three MTD partitions as follows.

bootargs=noinitrd rootfstype=squashfs,ubifs console=ttyS0,115200n8 rdinit=/sbin/init mem=64M mtdparts=nand0:0x180000@0x0(u-boot),0x20000@0x180000(dtb),-@0x200000(firmware)

If user wants to change the MTD partitions, please modify the U-Boot environment file. The image start offset in NuWriter should also be changed.

2.6 Test the OpenWrt

After starting the device, user will see the OpenWrt booting messages, as shown in Figure 2-6.

```
<u>14.383924] random: ubusd: uninitialized urandom read (4 bytes read, 78 bits of entropy available</u>
     14.394891] procd: - init -
Please press Enter to activate this console.
    27.863291] nuc980-emac0 nuc980-emac0: eth0 is OPENED
    27.868516] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready 28.680199] nuc980-emac0 nuc980-emac0: eth0 is OPENED
    28.690683] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready
    29.742066] random: nonblocking pool is initialized
31.592358] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
BusyBox v1.25.1 () built-in shell (ash)
               LE
                                                                       lede-project.org
                DE
        ΙF
                       Reboot (17.01.7, r4030-6028f00df0)
 There is no root password defined on this device!
Use the "passwd" command to set up a new password in order to prevent unauthorized SSH logins.
 root@LEDE:/#
```

Figure 2-6 OpenWrt Booting Messages

2.6.1 Network Settings

To get the network setting information, user can run the following command.

uci show network

The default network setting is using the DHCP address, as shown in Figure 2-7.



```
root@LEDE:/#
root@LEDE:/# uci show network
network.loopback=interface
network.loopback.ifname='lo'
network.loopback.proto='static'
network.loopback.netmask='255.0.0.0'
network.lan=interface
network.lan.ifname='eth0'
network.lan.proto='dhcp'
network.lan.netmask='255.255.255.0'
root@LEDE:/#_
```

Figure 2-7 Default Network Settings

To change the network settings, user can modify the file path /etc/config/network directly, or run the "uci set" command. For example, to change to a static address, user can run following commands to modify and reset the network settings.

```
uci set network.lan.proto=static
uci set network.lan.ipaddr=192.168.1.100
uci set network.lan.netmask=255.255.255.0
/etc/init.d/network restart
```

2.6.2 LuCl Web Interface

To login the LuCl Web interface, user can connect to https://YOUR_IP_ADDRESS through a web browser such as Chrome. In the first time login, user may encounter a security warning message, as shown in Figure 2-8.

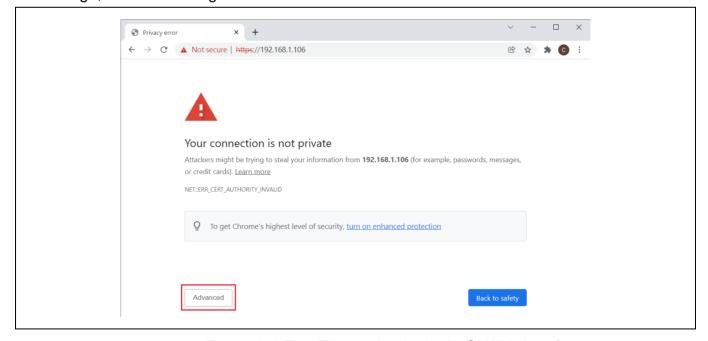


Figure 2-8 First Time to Login the LuCl Web Interface



After clicking the **Advanced** button, user will see a new screen, as shown in Figure 2-9. Please proceed with the connection.

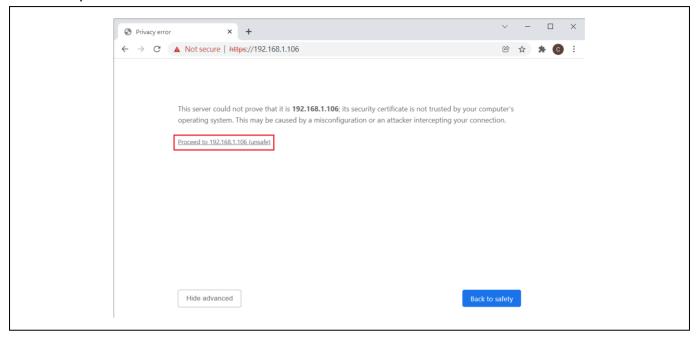


Figure 2-9 Proceed with Connection to LuCI

Then user can see the login screen. Since there is no password by default, user can login directly, or set a new password, as shown in Figure 2-10.

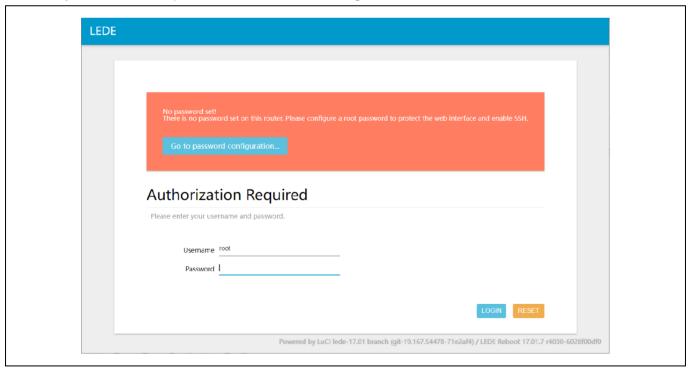


Figure 2-10 Login Screen of LuCI



2.6.3 Firmware Update

With the LuCl interface, user can upgrade the image which includes Linux kernel and OverlayFS root filesystem. For the NUC980 Chili board, the image is as follows.

lede-nuc980-nuc980-chili-nuc980-chili-squashfs-sysupgrade.bin

To do the OpenWrt firmware upgrade, enter "System -> Backup/Flash Firmware". Choose the new sysupgrade image file, and then click "FLASH IMAGE" button, as shown in Figure 2-11.

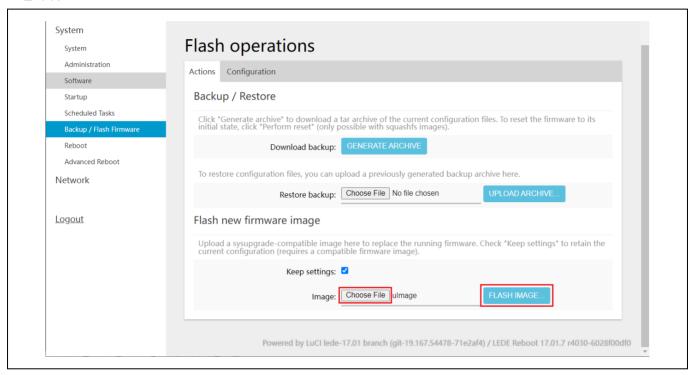


Figure 2-11 Firmware Upgrade in LuCI



Revision History

Date	Revision	Description
2022.03.15	1.00	1. Initially issued.
2022.05.18	1.10	Supported Mtd-Split and OverlayFS features in sections 2.4, 2.5 and 2.6.3.



Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.

All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.