

NuMicro[®] Family**ARM926EJ-S[™]-based 32-bit Microprocessor**

NuMaker-Server-NUC980 User Manual

Evaluation Board for 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.....	5
2	Features	6
3	Hardware Configuration.....	7
3.1	Front View.....	7
3.2	Rear View	15
4	Quick Start	16
4.1	Nuvoton CDC Driver Installation.....	16
4.2	Nuvoton Virtual COM Driver Installation.....	17
4.3	BSP Firmware Download.....	21
4.4	Hardware Setup	21
4.5	NuWriter Tool	23
4.6	Executing Sample Code.....	27
4.7	Make your own Embedded Linux	31
5	Block Diagram Schematic	32
5.1	GPIO List Schematic.....	32
5.2	Power Schematic.....	33
5.3	NUC980DK Schematic	34
5.4	Power Filter Schematic	35
5.5	Configure Schematic.....	36
5.6	NUC123ZD4AN0 Schematic.....	37
5.7	Memory Schematic.....	38
5.8	RMII_PE Schematic.....	39
5.9	RMII_PF Schematic.....	40
5.10	UART_A Schematic.....	41
5.11	UART_B Schematic.....	42
5.12	USB Schematic	43
5.13	Expand Schematic	44
5.14	PCB Placement.....	45
6	REVISION HISTORY	47

List of Figures

Figure 1-1 NuMaker-Server Development Board	5
Figure 3-1 Front View of NuMaker-Server	7
Figure 3-2 Rear View of NuMaker-Server	15
Figure 4-1 Nuvoton USB Driver Installation Setup	16
Figure 4-2 CDC Driver Installation	17
Figure 4-3 VCOM Driver Installation Setup	17
Figure 4-4 VCOM Driver Installation	20
Figure 4-5 Hardware Setting	22
Figure 4-6 Nuvoton VCOM	22
Figure 4-7 NuWriter Setting	23
Figure 4-8 Program u-boot-spl	24
Figure 4-9 Program u-boot	25
Figure 4-10 Program dtb	26
Figure 4-11 Program Kernel Image	26
Figure 4-12 Program environment	27
Figure 4-9 NuMaker-Server Board Setup	28
Figure 4-10 Serial COM Port Setting	29
Figure 4-11 Serial COM Port	30
Figure 4-12 TCP/IP Connection Window	30
Figure 4-13 UART Setting Web Page	31
Figure 5-1 GPIO List Schematic	32
Figure 5-2 Power Schematic	33
Figure 5-3 NUC980DK Schematic	34
Figure 5-4 Power Filter Schematic	35
Figure 5-5 Configure Schematic	36
Figure 5-6 NUC123ZD4AN0 Schematic	37
Figure 5-7 Memory Schematic	38
Figure 5-8 RMII_PE Schematic	39
Figure 5-9 RMII_PF Schematic	40
Figure 5-10 UART_A Schematic	41
Figure 5-11 UART_B Schematic	42
Figure 5-12 USB Schematic	43
Figure 5-13 Expand Schematic	44
Figure 5-14 Front PCB Placement	45
Figure 5-15 Back PCB Placement	46

List of Tables

Table 4-1 Power On Setting 23

1 OVERVIEW

NuMaker-Server can gather information from up to 8 UART sensors and transfer to cloud server over two on board Ethernet port. It can also transfer information from cloud server by Ethernet to UART devices in opposite direction by applications. This serial server is a very popular IIoT device used in industrial control, which can easily transfer an onsite industrial machine control into a remote cloud factory control.

NuMaker-Server Development Board has two sets of RS232/RS485 transceiver ports on board and six sets of UART function pins. Company with NUC980 high performance DMA channels, the data transfer is in a very high efficiency way between 8 UARTs and 2 Ethernet. This is why NUC980 can easily satisfy most of the high performance serial transfer requirements.

This document provides a quick start guide for the NuMaker-Server Development Board. Users can understand both software and hardware configurations for the NuMaker-Server. The platform provides Linux OS and plenty of industrial control protocol for users to implement the Ethernet control applications in a very short time.

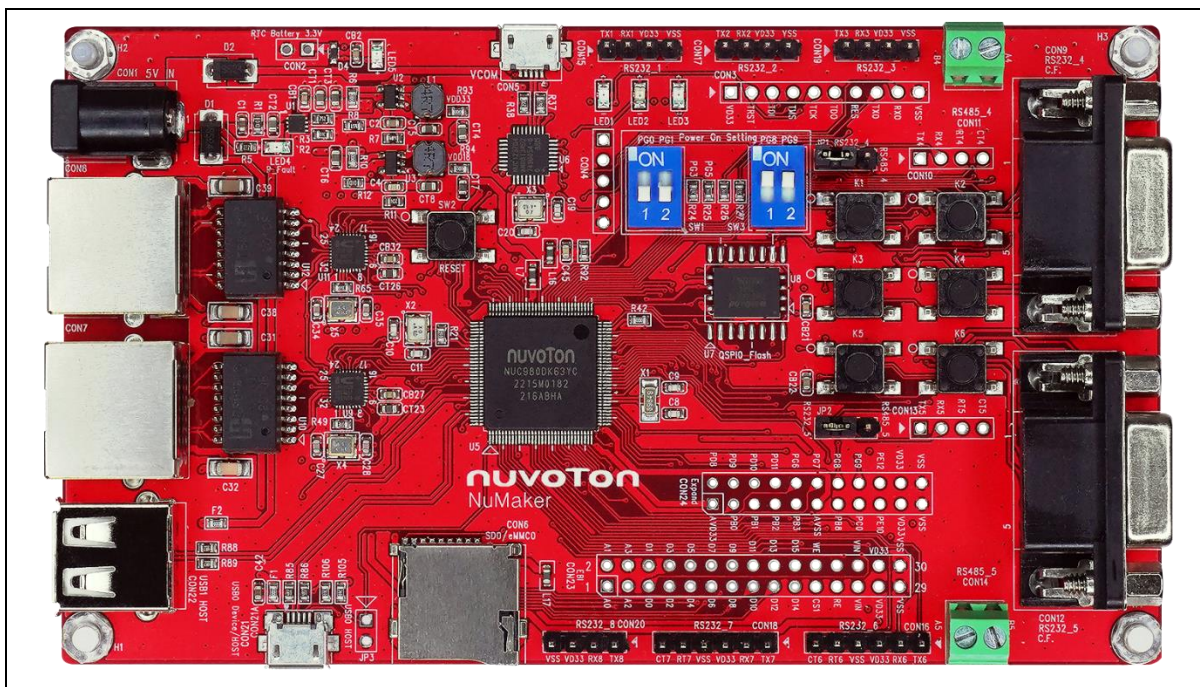


Figure 1-1 NuMaker-Server Development Board

2 FEATURES

- Target Chip NUC980DK63YC (V2.0) or NUC980DK61Y (V1.2/1.0): LQFP128 pin MCP package with DDR2 (64 MB), which can run up to 300MHz operating speed
- SPI Flash: Quad mode system booting or data storage
- SD0/eMMC0: User SD/eMMC memory card for system booting, data storage or SDIO (Wi-Fi) device
- Provides 9 sets of COM ports
 - UART0: Connected to Virtual COM port for system development, debug message output
 - UART4/UART5: 2 sets of DB9 port with RS232 transceiver
 - UART1~3/UART6~8: 6 sets of pin headers
- JTAG interface provided for software development
- 2 sets of RJ45 port with Ethernet 10/100Mbps MAC
- 3 sets of LED for status indication
- 6 sets of user-configurable push button keys
- USB port-0 that can be used as Device/HOST and USB port-1 that can be used as HOST
Supports pen drives, keyboards, mouse and printers
- Provides over-voltage and over current protection
- 3.3V I/O power, 1.8V Memory power and 1.2V core power

3 HARDWARE CONFIGURATION

3.1 Front View

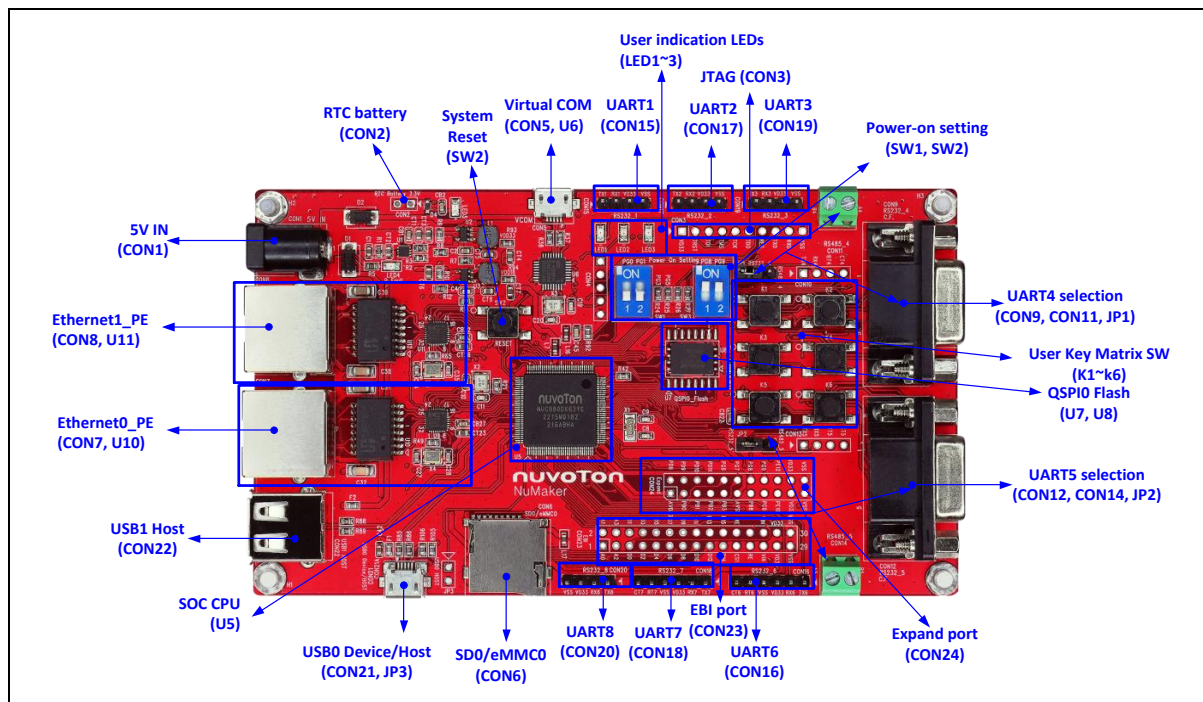


Figure 3-1 Front View of NuMaker-Server

Figure 3-1 shows the main components from the front view of NuMaker-Server Development Board

- +5V In (CON1): Power adaptor 5V input

Power Model	CON5 USB Port (Micro-B)	CON21 USB Port (Micro-B)	CON1
Model 1	Connect to PC	-	-
Model 2	-	Connect to PC	-
Model 3	-	-	VDD5V Input

- Power indication LEDs (LED4, LED5):

LED	Color	Descriptions
LED4	Red	The system power will be terminated and LED4 lighting when the input voltage is over 5.7V or the current is over 1.7A.
LED5	Green	Power normal state.

- RTC Battery (CON2): External Battery supply for RTC 3.3V powered
 - CON2.1: Positive (+)
 - CON2.2: Negative (-)
- System Reset (SW2): System will be reset if the SW2 button is pressed
- Virtual COM (CON5, U6): NUC123ZD4AN0 microcontroller (U6), USB micro-B connector (CON5) to PC, for debug message output
- User indication LEDs (LED1, LED2, LED3):

LED	Color	GPIO pin of NUC980
LED1	Yellow	GPG15
LED2	Green	GPB13
LED3	Red	GPF10

- UART1 pin header (CON15)

Connector	GPIO pin of NUC980	Function
CON15.1	GPA1	UART1_TXD
CON15.2	GPA0	UART1_RXD
CON15.3	-	VDD33
CON15.4	-	VSS

- QSPI0 Flash (U7, U8): Use Winbond W25N01GVZE1G 128 MB SPI-NAND (U7) for system booting, only one (U7 or U8) SPI Flash can be used, support dual / quad mode
- UART2 pin header (CON17)

Connector	GPIO pin of NUC980	Function
CON17.1	GPA10	UART2_TXD
CON17.2	GPA9	UART2_RXD
CON17.3	-	VDD33
CON17.4	-	VSS

- UART3 pin header (CON19).

Connector	GPIO pin of NUC980	Function
CON19.1	GPC3	UART3_TXD
CON19.2	GPC4	UART3_RXD
CON19.3	-	VDD33
CON19.4	-	VSS

- JTAG interface and UART0 (CON3)

Connector	GPIO pin of NUC980	Function
CON3.1	-	VDD33
CON3.2	GPG15	nTRST
CON3.3	GPG14	TDI
CON3.4	GPG13	TMS
CON3.5	GPG12	TCK
CON3.6	GPG11	TDO
CON3.7	-	nRESET
CON3.8	GPF12	UART0_TXD
CON3.9	GPF11	UART0_RXD
CON3.10	-	VSS

- UART4 selection (CON9, CON11, JP1):

- JP1: 1-2 short for RS232 function with RS232 transceiver, and RS232 connected DB9 female (CON9)
- JP1: 2-3 short for RS485 function with RS485 transceiver, and RS485 connected to 2P terminal (CON11)

Function	GPIO pin of NUC980
UART4_232_TXD/485_D	GPD12
UART4_232_RXD/485_R	GPD13
UART4_232_RTS/485_(/RE&DE)	GPD14
UART4_232_CTS	GPD15

- User Key Matrix SWs (K1~K6)

Key	Function	GPIO pin of NUC980
K1	Row0	GPC13
	Column0	GPC1
K2	Row0	GPC13
	Column1	GPC2
K3	Row1	GPC14
	Column0	GPC1
K4	Row1	GPC14
	Column1	GPC2
K5	Row2	GPC15
	Column0	GPC1
K6	Row2	GPC15
	Column1	GPC2

- UART5 selection (CON12, CON14, JP2):
 - JP2: 1-2 short for RS232 function with RS232 transceiver, and RS232 connected DB9 female (CON12)
 - JP2: 2-3 short for RS485 function with RS485 transceiver, and RS485 connected to 2P terminal (CON14)

Function	GPIO pin of NUC980
UART5_232_TXD/485_D	GPG14
UART5_232_RXD/485_R	GPG13
UART5_232_RTS/485_(/RE&DE)	GPG12
UART5_232_CTS	GPG11

- UART6 pin header (CON16)

Connector	GPIO pin of NUC980	Function
CON16.1	GPA5	UART6_TXD
CON16.2	GPA4	UART6_RXD
CON16.3	-	VDD33
CON16.4	-	VSS
CON16.5	GPA3	UART6_RTS
CON16.6	GPA2	UART6_CTS

- Expand port for user use (CON24)

Connector	GPIO pin of NUN980	Function
CON24.1	-	ADC VSS
CON24.2	GPD8	SPI0_SS0
CON24.3	GPB0	ADC_AIN[0]
CON24.4	GPD9	SPI0_CLK
CON24.5	GPB1	ADC_AIN[1]
CON24.6	GPD10	SPI0_DO
CON24.7	GPB2	ADC_AIN[2]
CON24.8	GPD11	SPI0_DI
CON24.9	GPB3	ADC_AIN[3]
CON24.10	GPG6	PWM10
CON24.11	-	ADC VDD33
CON24.12	GPG7	PWM11
CON24.13	GPB8	CAN2_RXD
CON24.14	GPG8	PWM12
CON24.15	GPC0	CAN2_TXD
CON24.16	GPG9	PWM13
CON24.17	GPE10	I2C0_SDA
CON24.18	GPE12	I2C0_SCL
CON24.19	-	VDD33
CON24.20	-	VDD33
CON24.21	-	VSS
CON24.22	-	VSS

- UART7 pin header (CON18)

Connector	GPIO pin of NUC980	Function
CON18.1	GPB6	UART7_TXD
CON18.2	GPB4	UART7_RXD
CON18.3	-	VDD33
CON18.4	-	VSS
CON18.5	GPB5	UART7_RTS
CON18.6	GPB7	UART7_CTS

- EBI port for user use (CON23)

Connector	GPIO pin of NUN980	Function
CON23.1	GPG0	EBI_ADDR0
CON23.2	GPG1	EBI_ADDR1
CON23.3	GPB2	EBI_ADDR2
CON23.4	GPG3	EBI_ADDR3
CON23.5	GPC0	EBI_DATA0
CON23.6	GPC1	EBI_DATA1
CON23.7	GPC2	EBI_DATA2
CON23.8	GPC3	EBI_DATA3
CON23.9	GPC4	EBI_DATA4
CON23.10	GPC5	EBI_DATA5
CON23.11	GPC6	EBI_DATA6
CON23.12	GPC7	EBI_DATA7
CON23.13	GPC8	EBI_DATA8
CON23.14	GPC9	EBI_DATA9
CON23.15	GPC10	EBI_DATA10
CON23.16	GPC11	EBI_DATA11
CON23.17	GPC12	EBI_DATA12
CON23.18	GPC13	EBI_DATA13
CON23.19	GPC14	EBI_DATA14
CON23.20	GPC15	EBI_DATA15
CON23.21	GPA6	EBI_nCS1
CON23.22	GPA7	EBI_nWE
CON23.23	GPA8	EBI_nRE
CON23.24	-	-
CON23.25	-	VIN
CON23.26	-	VIN
CON23.27	-	VDD33
CON23.28	-	VDD33
CON23.29	-	VSS
CON23.30	-	VSS

- UART8 pin header (CON20)

Connector	GPIO pin of NUC980	Function
CON20.1	GPA12	UART8_TXD
CON20.2	GPA11	UART8_RXD
CON20.3	-	VDD33
CON20.4	-	VSS

- SD0/eMMC0 (CON6): Use Micro SD/eMMC memory card for system booting, data storage or SDIO (Wi-Fi) device
- Power on setting (SW1, SW2)

Switch	Status	Function	GPIO pin of NUC980
SW1.2/SW1.1	ON/ON	Boot from USB	GPG1/GPG0
SW1.2/SW1.1	ON/OFF	Boot from SD/eMMC	GPG1/GPG0
SW1.2/SW1.1	OFF/OFF	Boot from QSPI0 Flash	GPG1/GPG0

Resistance	Status	Function	GPIO pin of NUC980
R24	Solder R	Watchdog Timer OFF	GPG3
R24	Remove	Watchdog Timer ON	GPG3

Resistance	Status	Function	GPIO pin of NUC980
R25	Solder R	UART0 debug message ON	GPG5
R25	Remove	UART0 debug message OFF	GPG5

If SW1.2/SW1.1 status is ON / OFF

Resistance	Status	Function	GPIO pin of NUC980
SW2.2/SW2.1	ON/ON	SD0/eMMC0 boot from GPC group	GPG9/GPG8

If SW1.2/SW1.1 status is OFF / OFF

Switch	Status	Function	GPIO pin of NUC980
SW2.2/SW2.1	ON/ON	SPI-NAND Flash boot with 1-bit mode	GPG9/GPG8
SW2.2/SW2.1	ON/OFF	SPI-NAND Flash boot with 4-bit mode	GPG9/GPG8

SW2.2/SW2.1	OFF/ON	SPI-NOR Flash boot with 4-bit mode	GPG9/GPG8
SW2.2/SW2.1	OFF/OFF	SPI-NOR Flash boot with 1-bit mode	GPG9/GPG8

- USB0 Device/HOST (CON21, JP3): USB0 Device/HOST Micro-B connector, By JP3 status or defined by the ID pin of the USB cable
- USB1 HOST (CON22): USB1 for USB HOST with type-A connector
- Ethernet0_PE (CON7, U9): For Ethernet port, the NUC980 support RMII interface which add one Ethernet PHY IP101GR to RJ45 connector with LED indicator
- SOC CPU: NUC980DK63YC (V2.0) or NUC980DK61Y (V1.2/1.0) (U5)
- Ethernet1_PF (CON8, U11): For Ethernet port, the NUC980 support RMII interface which add one Ethernet PHY IP101GR to RJ45 connector with LED indicator

3.2 Rear View

Figure 3-2 shows the main components from the rear view of NuMaker-Server Development Board

- RS232-4/5 transceivers with SN75C3232E (U13 and U15)
- RS485-4/5 transceivers with SN65HVD10 (U14 and U16)

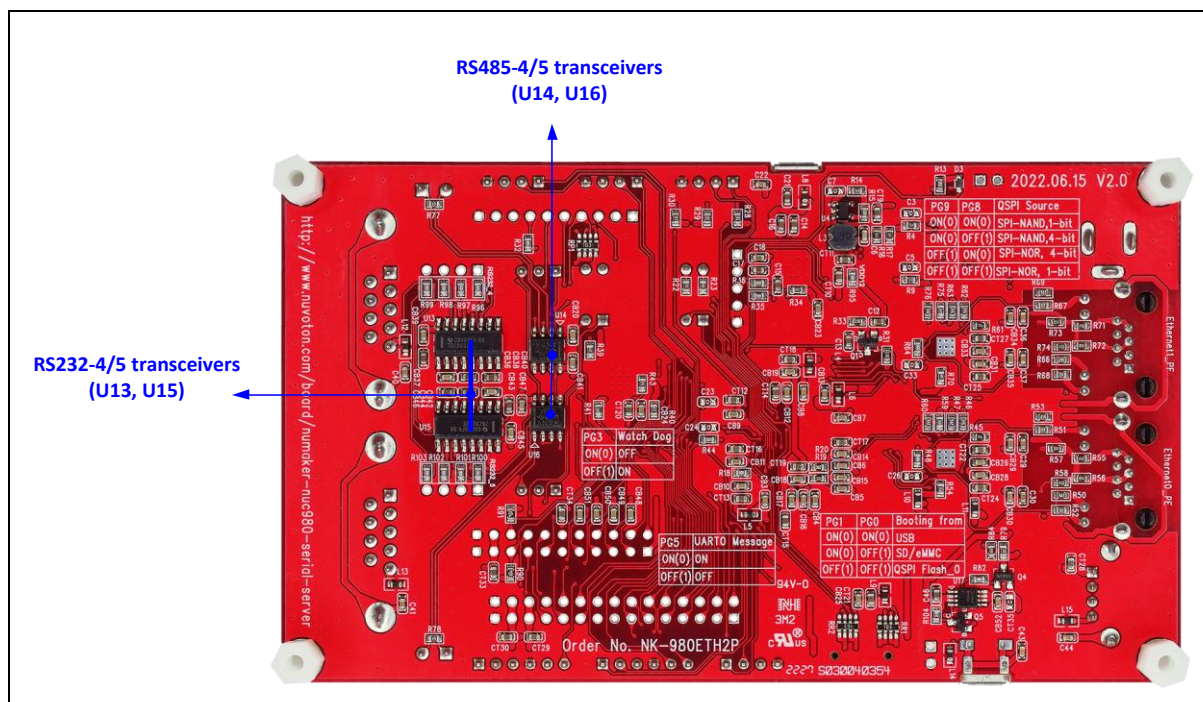


Figure 3-2 Rear View of NuMaker-Server

4 QUICK START

4.1 Nuvoton CDC Driver Installation

The USB serial port function is used to print some messages on PC API, such as SecureCRT, through the standard UART protocol to help user to debug program.

Download and install the latest Nuvoton CDC driver:

- https://www.nuvoton.com/resource-download.jsp?tp_GUID=SW1020160914071736

The installation is presented in Figure 4-1 and Figure 4-3.

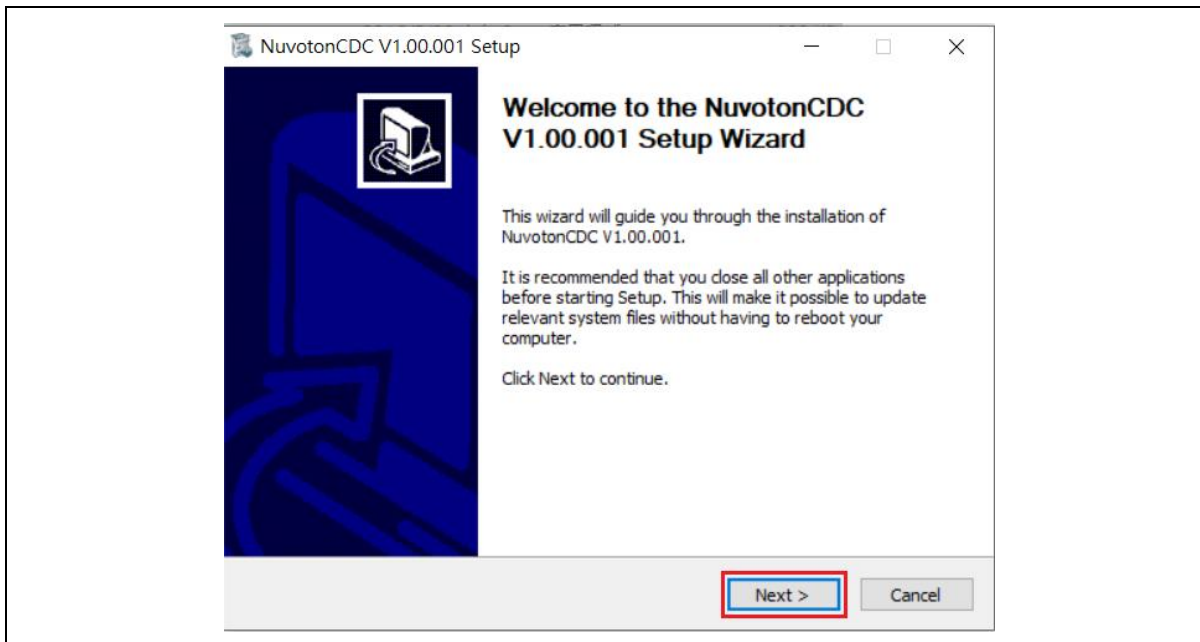
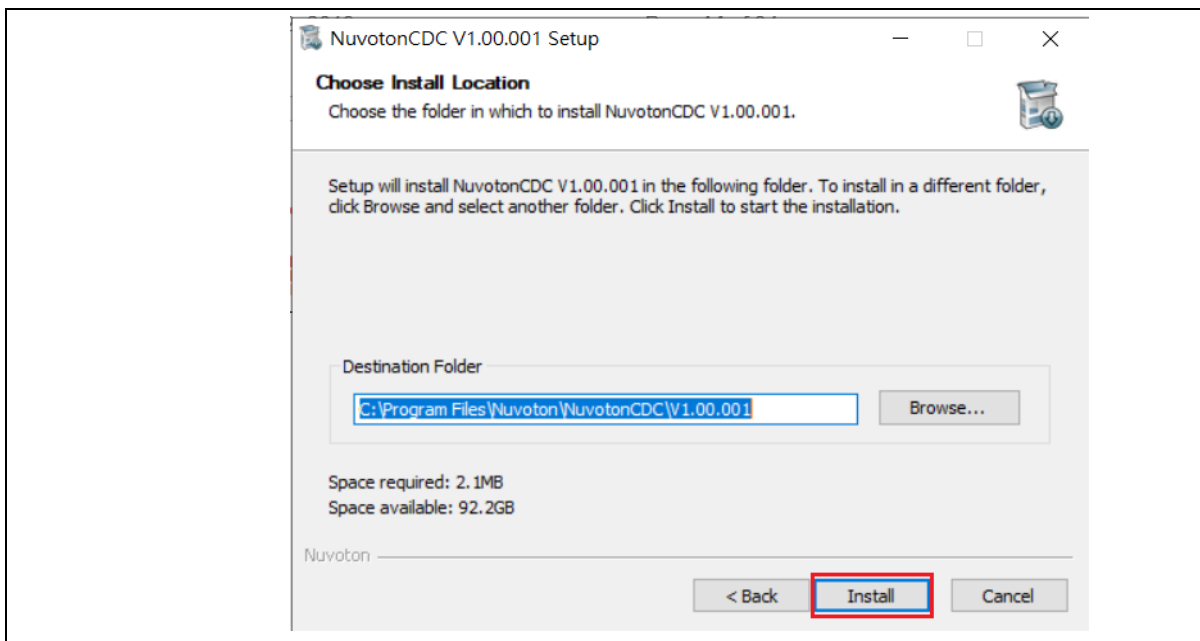


Figure 4-1 Nuvoton USB Driver Installation Setup



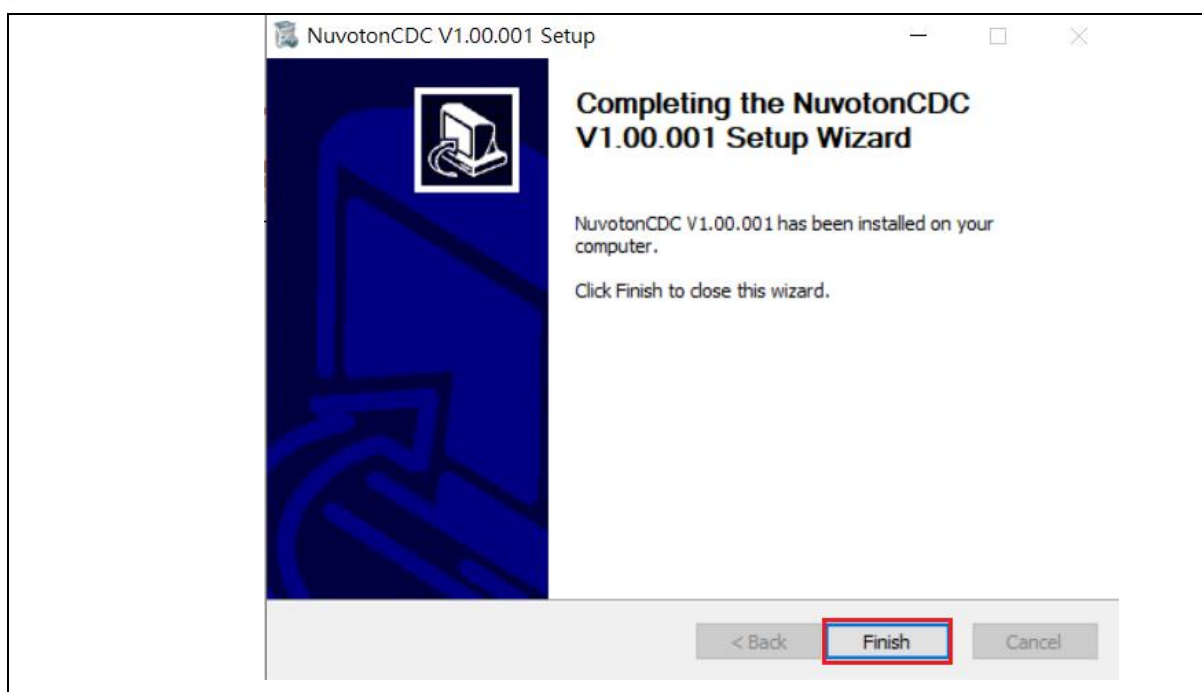


Figure 4-2 CDC Driver Installation

4.2 Nuvoton Virtual COM Driver Installation

The firmware programming tool NuWriter requires a NuWriter driver to be installed on PC first. Please follow the steps below to install the driver.

Download and install the latest Nuvoton Virtual COM driver:

- https://github.com/OpenNuvoton/NUC980_NuWriter/tree/master/Driver

The installation is presented in Figure 4-3 and Figure 4-4.

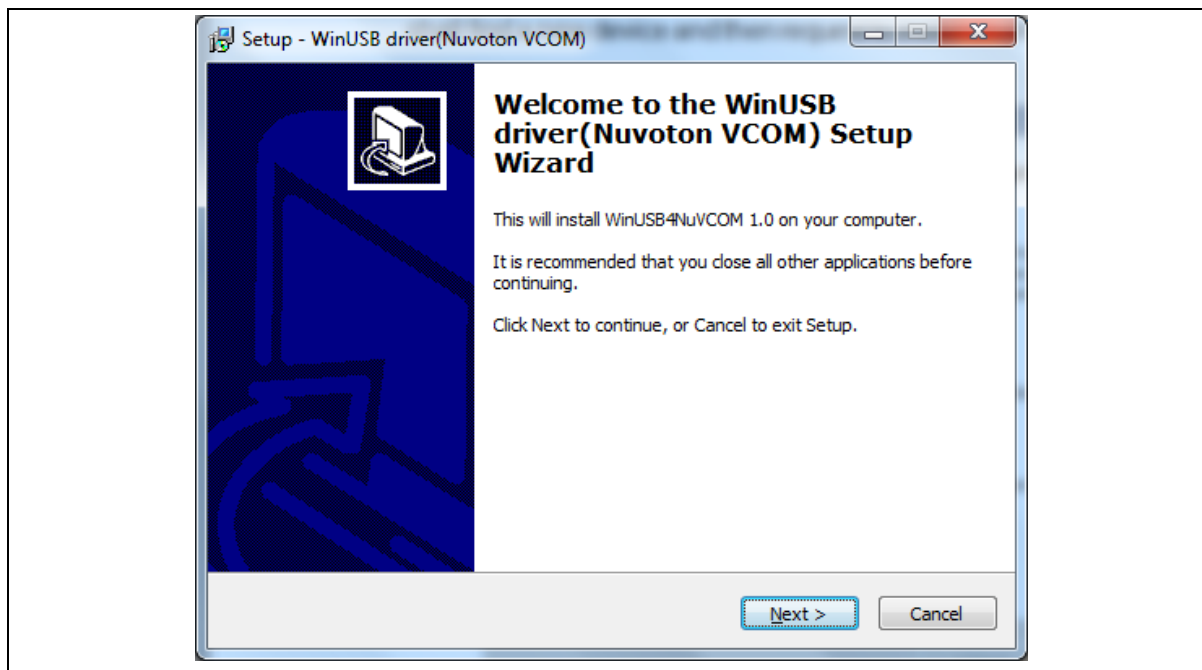
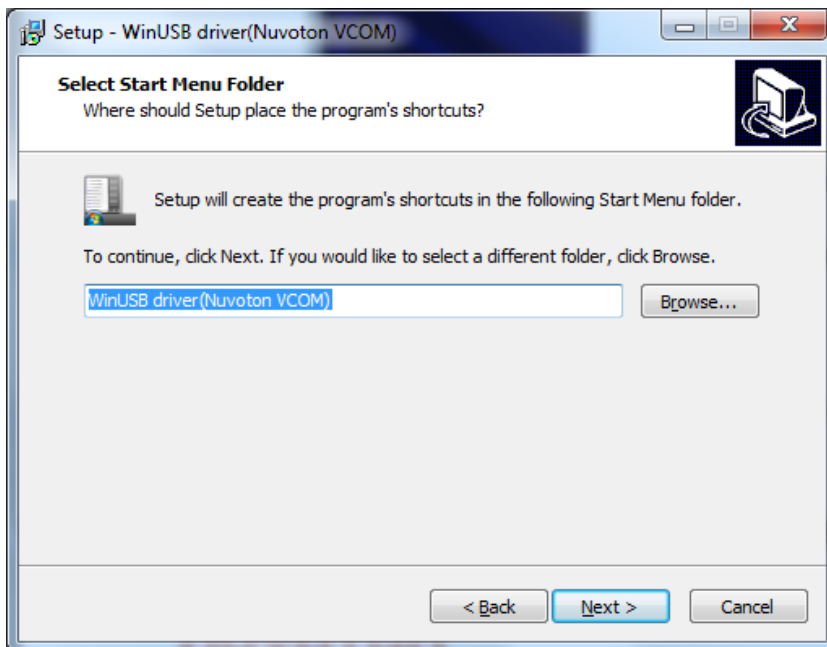
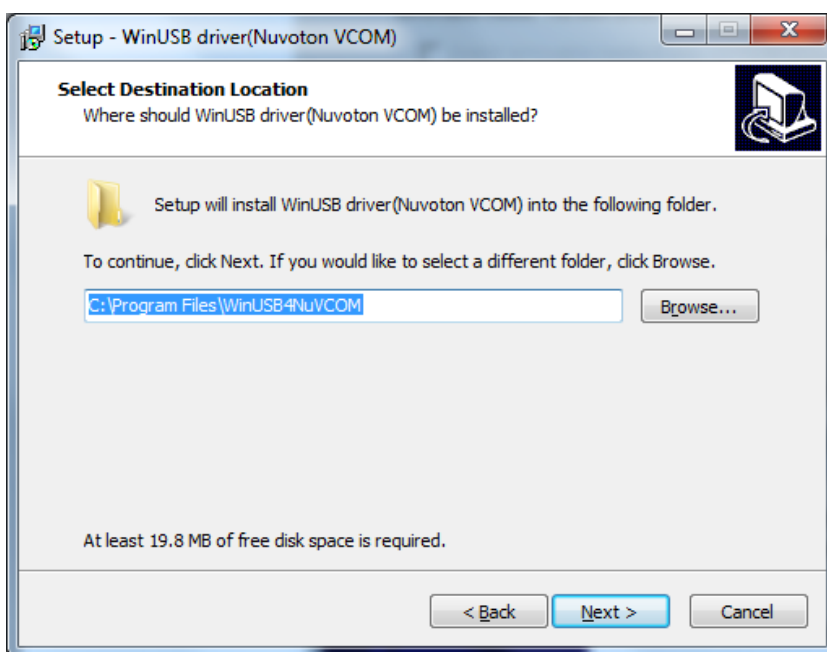
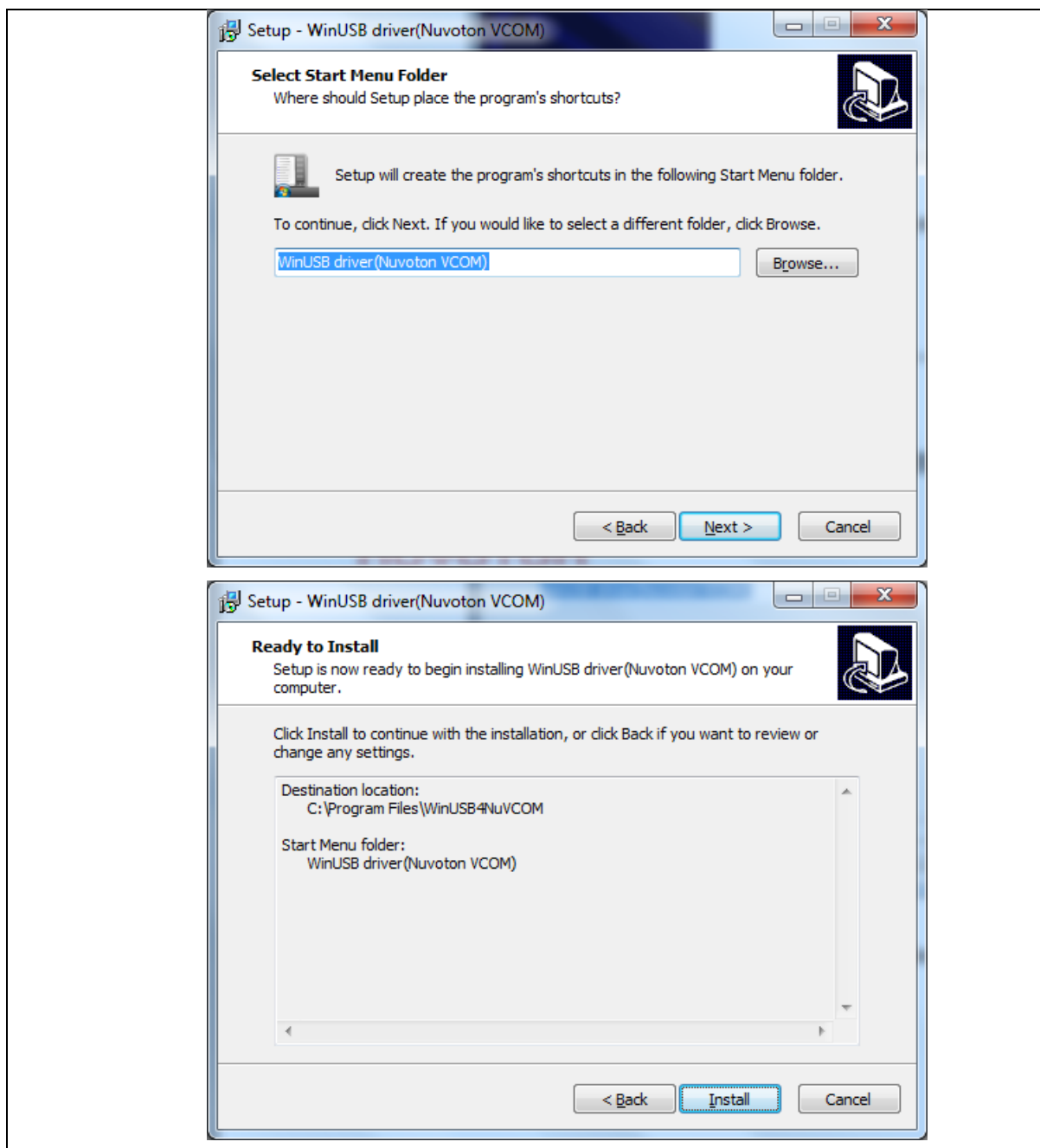


Figure 4-3 VCOM Driver Installation Setup





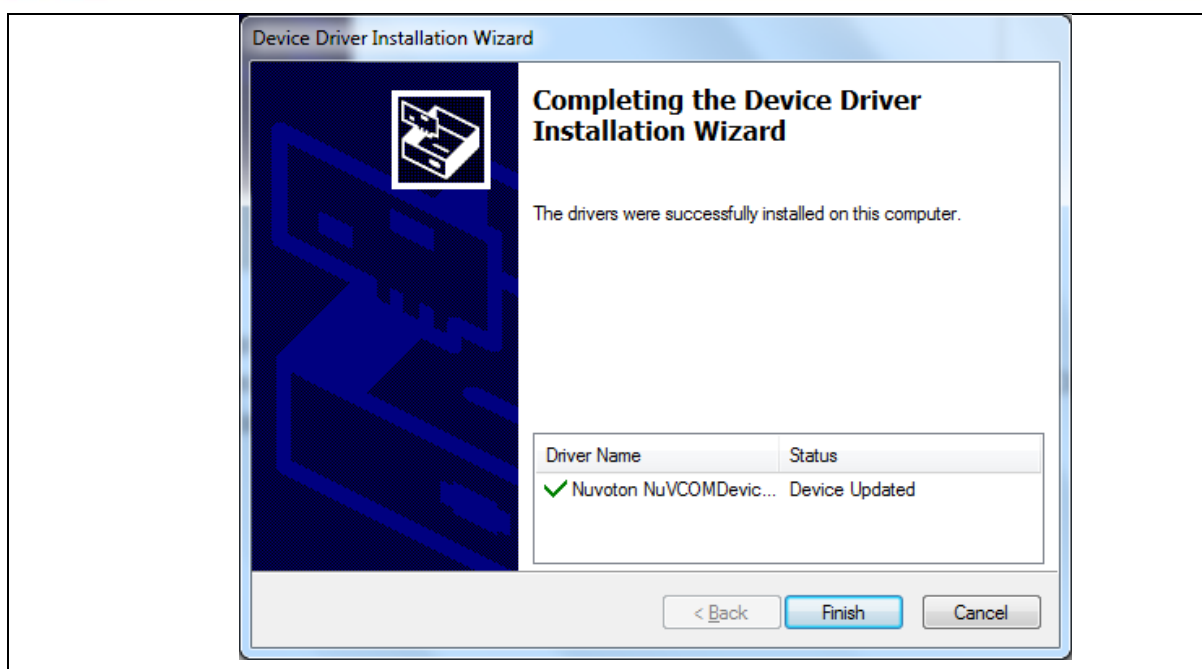


Figure 4-4 VCOM Driver Installation

4.3 BSP Firmware Download

NUC980 Linux BSP provides cross compilation tools based on Linux. This BSP has been tested in different x86 Linux distributions, including Ubuntu, CentOS, and Debian, etc. Since there are many distributions out there with different system configuration, sometimes it is necessary to change system setting or manually install some missing components for cross compilation.

Linux development environment could either be native, or install in a virtual machine executed on top of other operating system.

BSP download locations:

Official website:

<https://www.nuvoton.com/products/iot-solution/iot-platform/numaker-server-nuc980/>

- VMware Linux Virtual machine image
 - An UBUNTU18.04 VMware Image with NUC980 toolchain and Buildroot
- VMware Linux Virtual machine image User Manual
 - Introduction of NUC980 Buildroot usage and how to compile firmware for NUC980
- Hardware
 - Schematics and Gerber files
- NUC980 Linux V5.10 BSP
 - Linux BSP, NuWriter tool and documents

Github:

<https://github.com/OpenNuvoton/MPU-Family>

For more details about NUC980 Linux 5.10 BSP, please refer to [NUC980 Linux 5.10 BSP User Manual](#).

For more details about NUC980 Linux 4.4 BSP, please refer to *NUC980 Linux 4.4 BSP User Manual* in the “/Documents” directory from [NUC980 Linux 4.4 BSP](#).

4.4 Hardware Setup

1. Connect the USB micro-B port (CON5) to the PC HOST.

The PC HOST will supply 5V power to the NuMaker-Server and will recognize the board as a USB composite device.

The VCOM port function is used to print some messages on some Terminal Tools, such as Tera Term, PuTTY, etc. It is through the standard UART protocol to help user to debug.

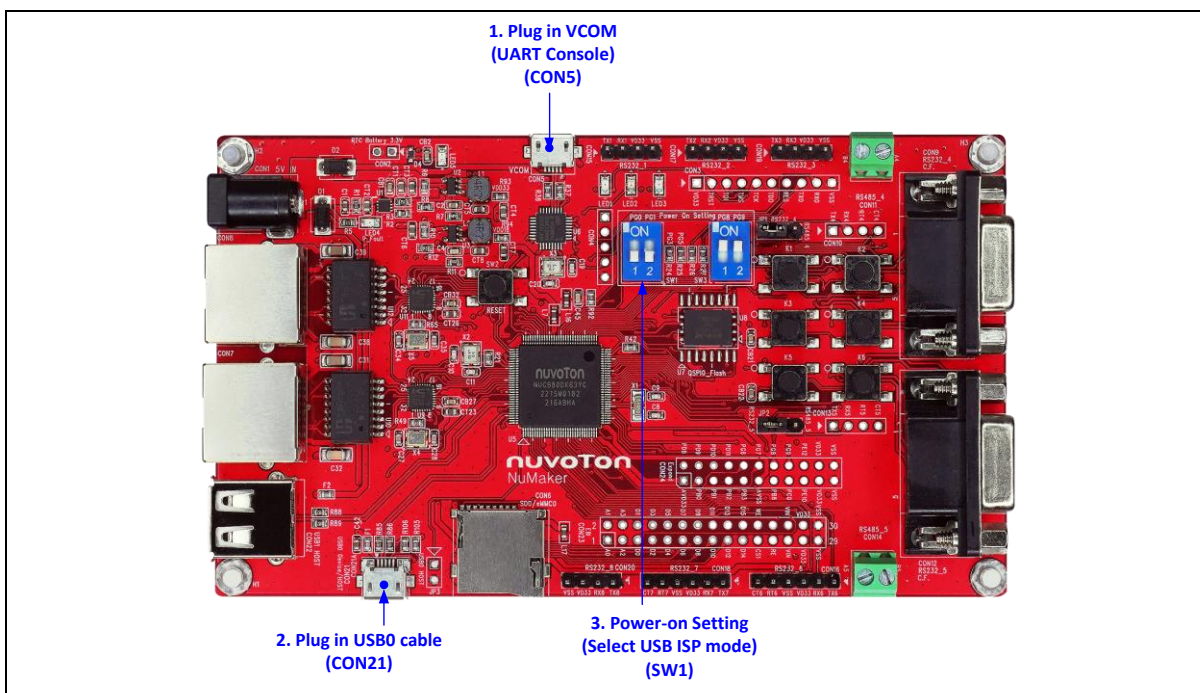


Figure 4-5 Hardware Setting

2. Plug in the USB0 cable (CON21)

If the installation is successful, a virtual COM port named "WinUSB driver (Nuvoton VCOM)" can be found in the "Device Manager".

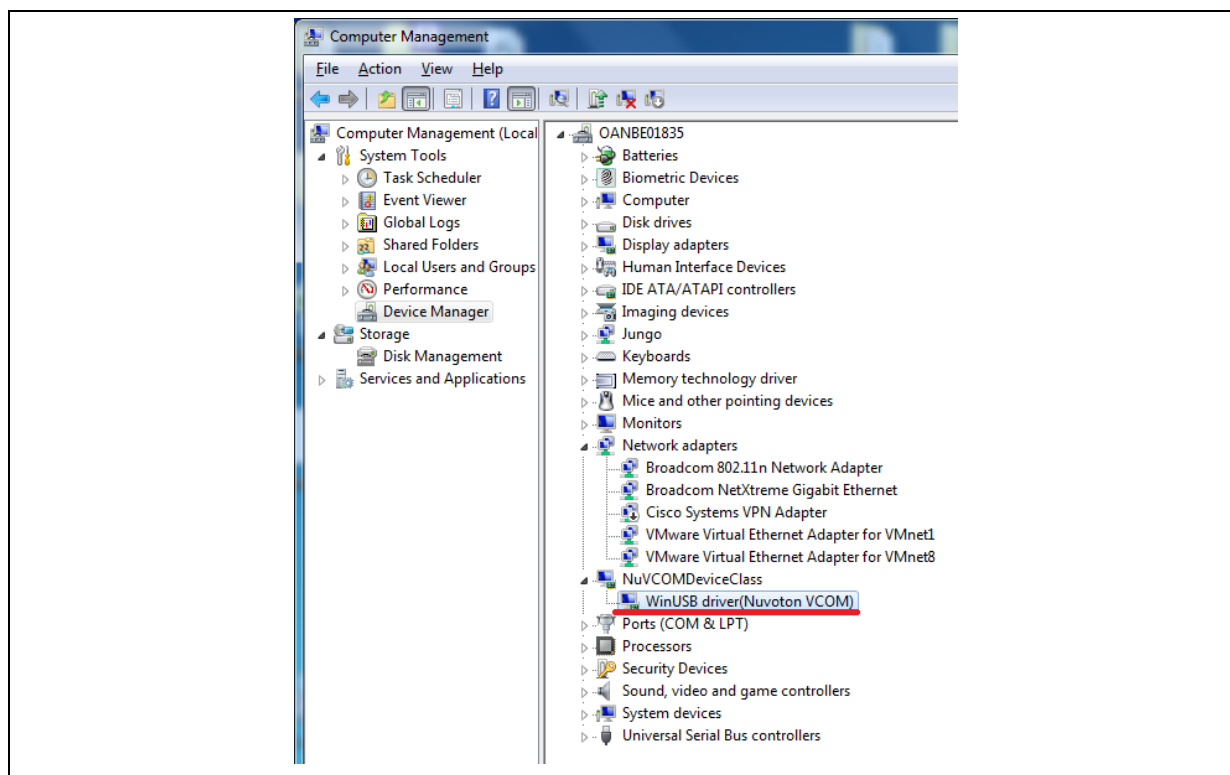


Figure 4-6 Nuvoton VCOM

3. Select the USB mode(SW1) Plug in the USB to UART cable (CON2)

Set power on setting(SW1) to ON/ON to Boot from USB.

SW	Description (Status and Function)	GPIO pin of NUC980
SW1.2/ SW1.1	Power on setting ON/ON = Boot from USB. ON/OFF = Boot from SD/eMMC. OFF/OFF = Boot from QSPI0 Flash.	GPG1/GPG0

Table 4-1 Power On Setting

4. Open the Serial Port Terminal and Reset

After pressing the reset button(SW2), the MPU will reprogram the application and print out debug message on the terminal.

4.5 NuWriter Tool

1. Install NuWriter Driver. (Please refer to “**NUC980 NuWriter User Manual**”)
2. Set SW1(Power On Setting) to Boot from USB (shown in Table 4-1 and Figure 4-5). Connect USB0 to PC and connect UART console to PC.
3. double click “**NuWriter.exe**” on PC. Select target chip as “NUC980 series” and select DDR parameter is “NUC980DK63YC.ini” (V2.0) or “NUC980DK61Y.ini” (V1.2/1.0). And then, press “**Continue**” button.

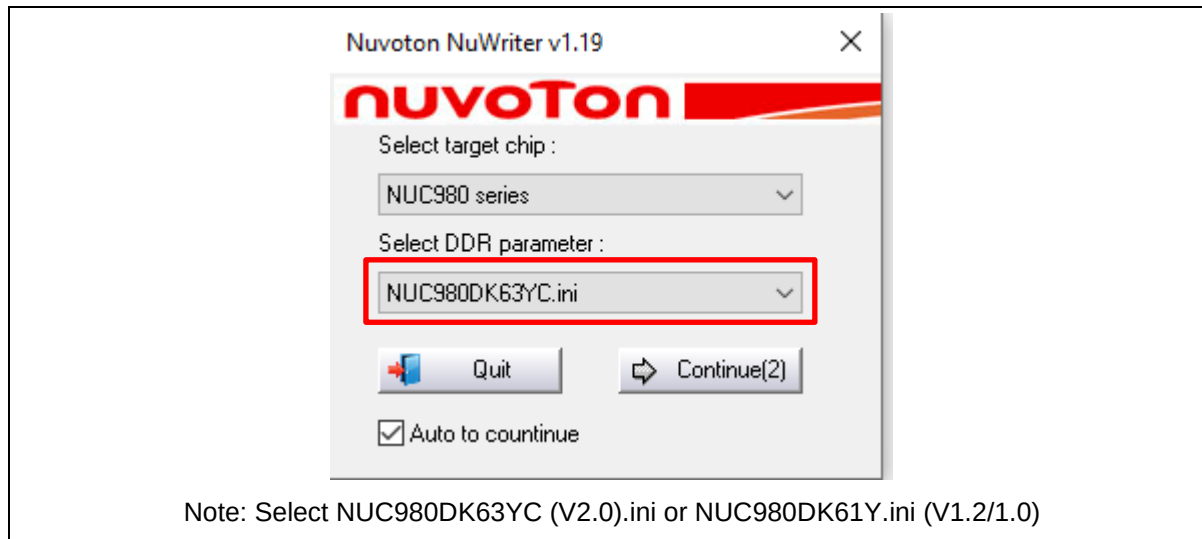


Figure 4-7 NuWriter Setting

4. According to Figure 4-8, following the steps below to program u-boot-spl.bin:
 - a. Select the “**SPI NAND**” type.
 - b. Fill in the image information :
 - Image Name: u-boot-spl.bin
 - Image Type: Loader
 - Image execute address: 0x200
 - c. Click “**Program**”.
 - d. Waiting for the progress bar to be finished.

- e. After “**Program**” the image, click the “**Verify**” button to read back the image data to make sure the burning status.

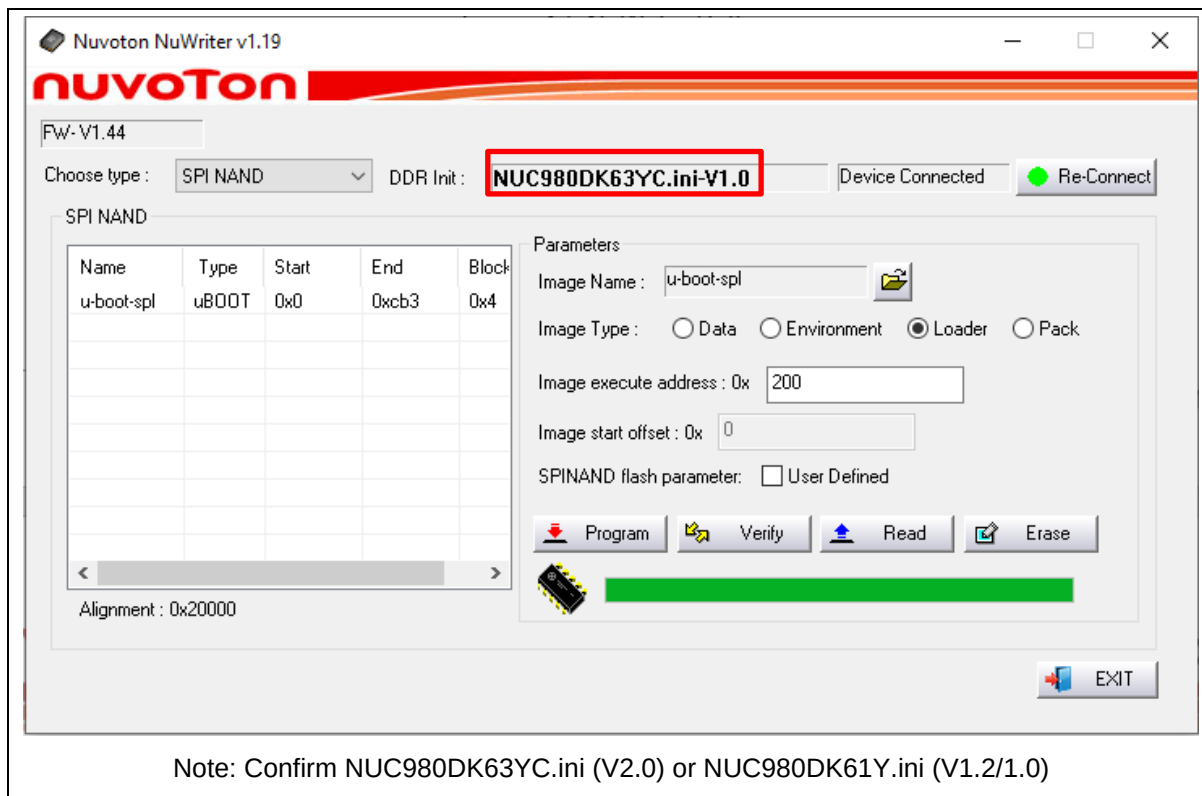


Figure 4-8 Program u-boot-spl

5. According to Figure 4-9, following the steps to program u-boot.bin:
 - a. Select the “**SPI NAND**” type.
 - b. Fill in the image information :
 - Image Name: u-boot.bin
 - Image Type: Data
 - Image execute address: 0x100000
 - c. Click “**Program**”.
 - d. Waiting the progress bar to be finished.
 - e. After “**Program**” the image, click the “**Verify**” button to read back the image data to make sure the burning status.

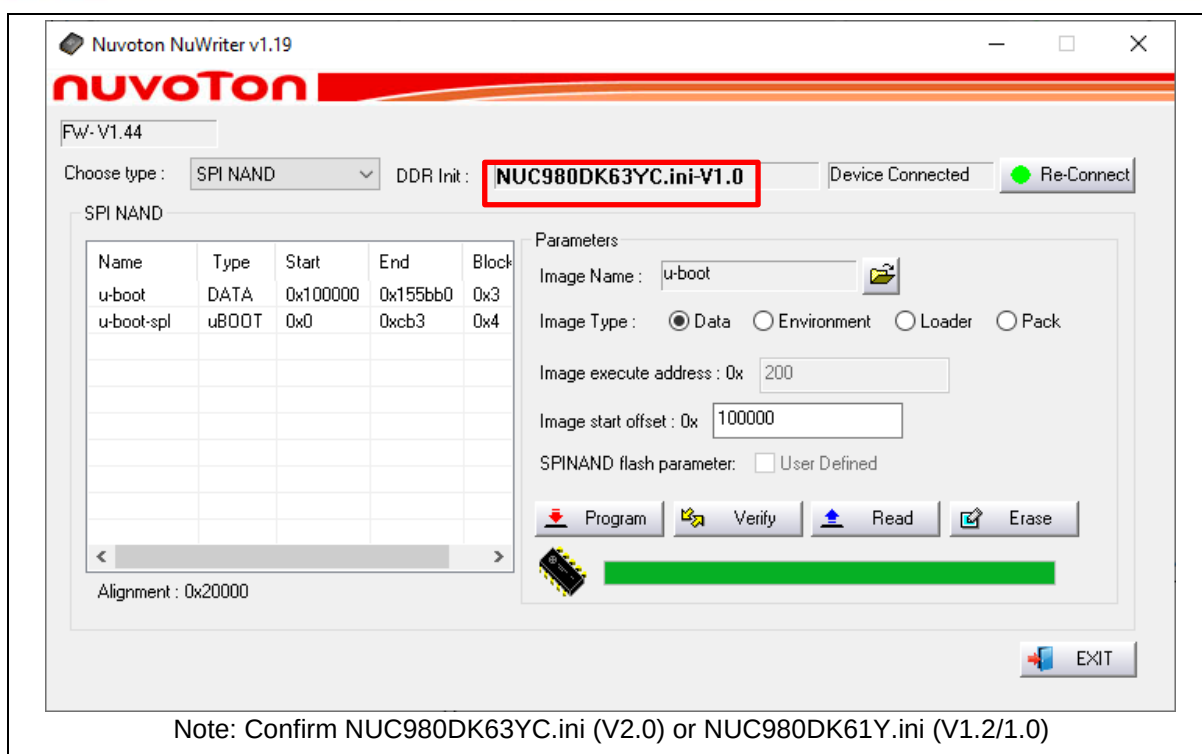
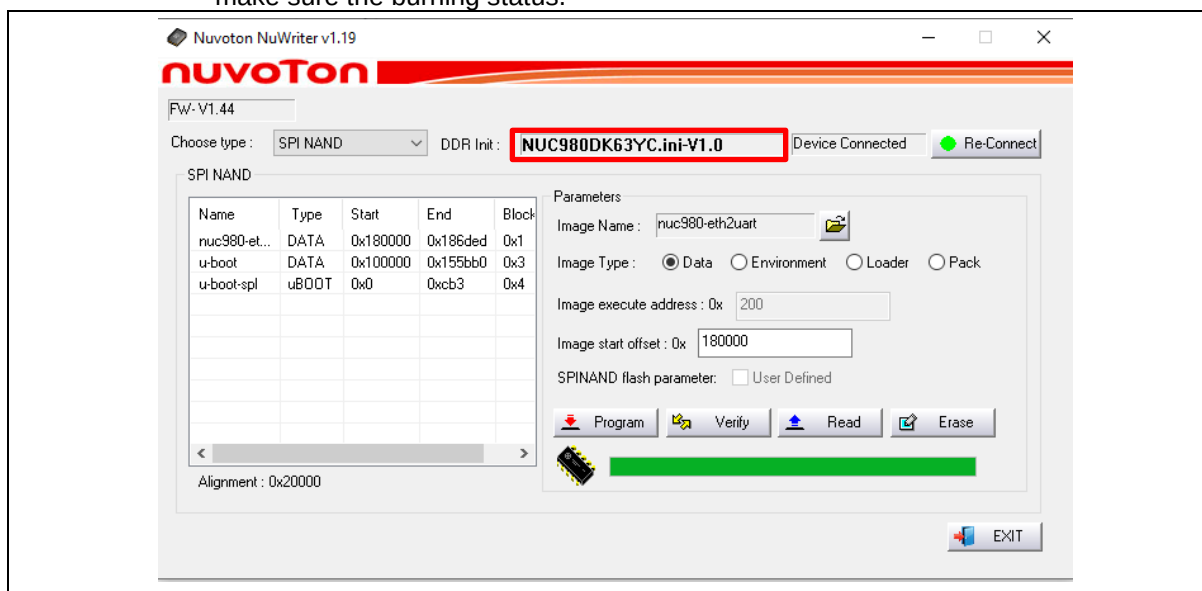


Figure 4-9 Program u-boot

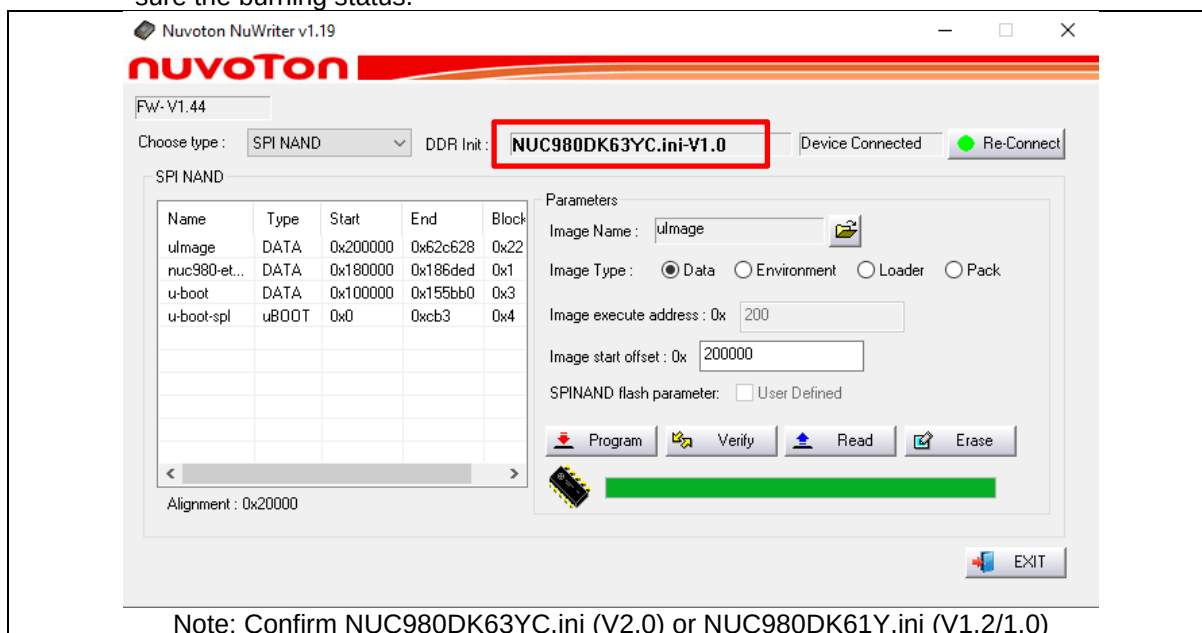
6. According to Figure 4-10, following the steps below to program kernel image:
 - a. Select the “**SPI NAND**” type.
 - b. Fill in the image information :
 - Image Name: nuc980-eth2uart.dtb
 - Image Type: Data
 - Image start offset address: 0x180000
 - c. Click “**Program**”.
 - d. Waiting for the progress bar to be finished.
 - e. After “**Program**” the image, click the “**Verify**” button to read back the image data to make sure the burning status.



Note: Confirm NUC980DK63YC.ini (V2.0) or NUC980DK61Y.ini (V1.2/1.0)

Figure 4-10 Program dtb

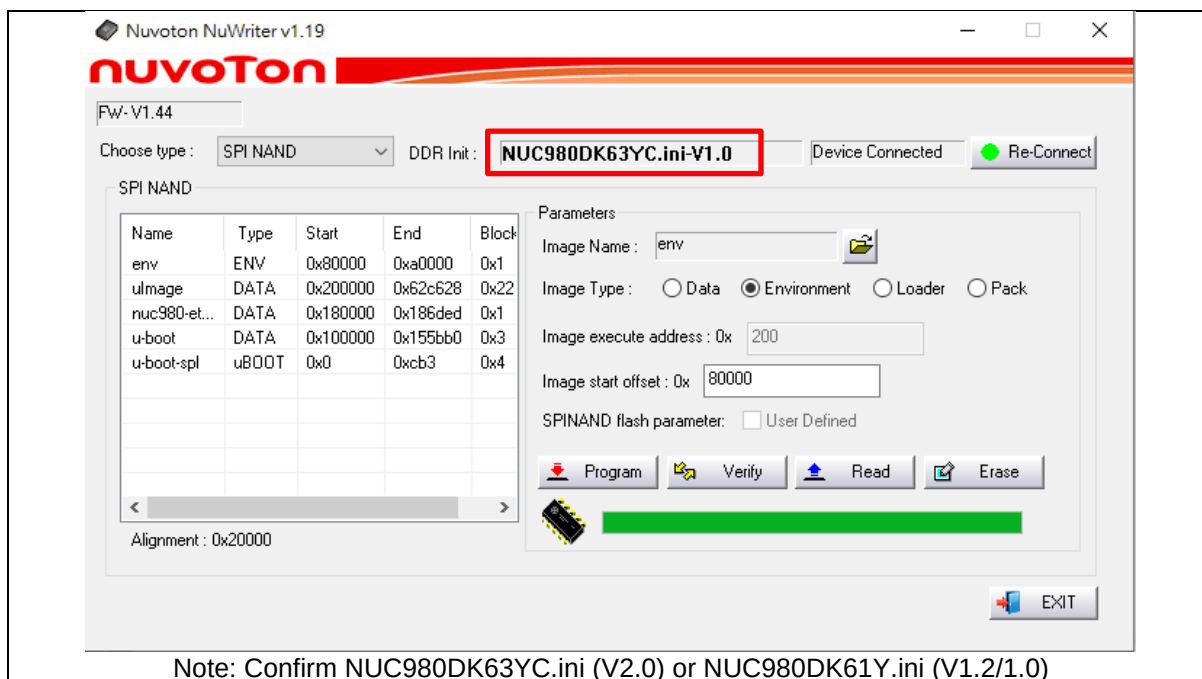
7. According to Figure 4-11, following the steps below to program kernel image:
 - a. Select the “**SPI NAND**” type.
 - b. Fill in the image information :
 - Image Name: uimage.bin
 - Image Type: Data
 - Image start offset address: 0x200000
 - c. Click “**Program**”.
 - d. Waiting for the progress bar to be finished.
 - e. After “**Program**” the image, click the “**Verify**” button to read back the image data to make sure the burning status.



Note: Confirm NUC980DK63YC.ini (V2.0) or NUC980DK61Y.ini (V1.2/1.0)

Figure 4-11 Program Kernel Image

8. According to Figure 4-12, following the steps below to program environment:
 - a. Select the “**SPI NAND**” type.
 - b. Fill in the image information :
 - Image Name: env.txt
 - Image Type: environment
 - Image start offset address: 0x80000
 - c. Click “**Program**”.
 - d. Waiting for the progress bar to be finished.
 - e. After “**Program**” the image, click the “**Verify**” button to read back the image data to make sure the burning status.



Note: Confirm NUC980DK63YC.ini (V2.0) or NUC980DK61Y.ini (V1.2/1.0)

Figure 4-12 Program environment

You can create a TXT file extension and add contents. NuWriter will transform env.txt to an environment image and download the image to SPI NAND.

Here is an example for NuMaker-server-NUC980 environment variables:

```
baudrate=115200
bootdelay=1
stderr=serial
stdin=serial
stdout=serial
loadkernel=nand read 0x7fc0 0x200000 0x800000
loaddtb=nand read 0x1400000 0x180000 0x20000
bootcmd=run loadkernel;run loaddtb;bootm 0x07fc0 - 0x1400000
```

For more details about NuWriter tool, please refer to “**NUC980 NuWriter User Manual**” in the “Documents” directory.

For more details about kernel image and uboot, please refer to **VMware Linux Virtual machine image User Manual**.

4.6 Executing Sample Code

Please make sure the UART console connect to PC first and follow the steps.

- Set SW1(Power On Setting) to Boot from QSPI 0 Flash(refer to Table 4-1 and Figure 4-5).
- Connect UART console port.
- Connect Ethernet0 to PC and connect UART1~8 to other UART device (ex:PC COM port).

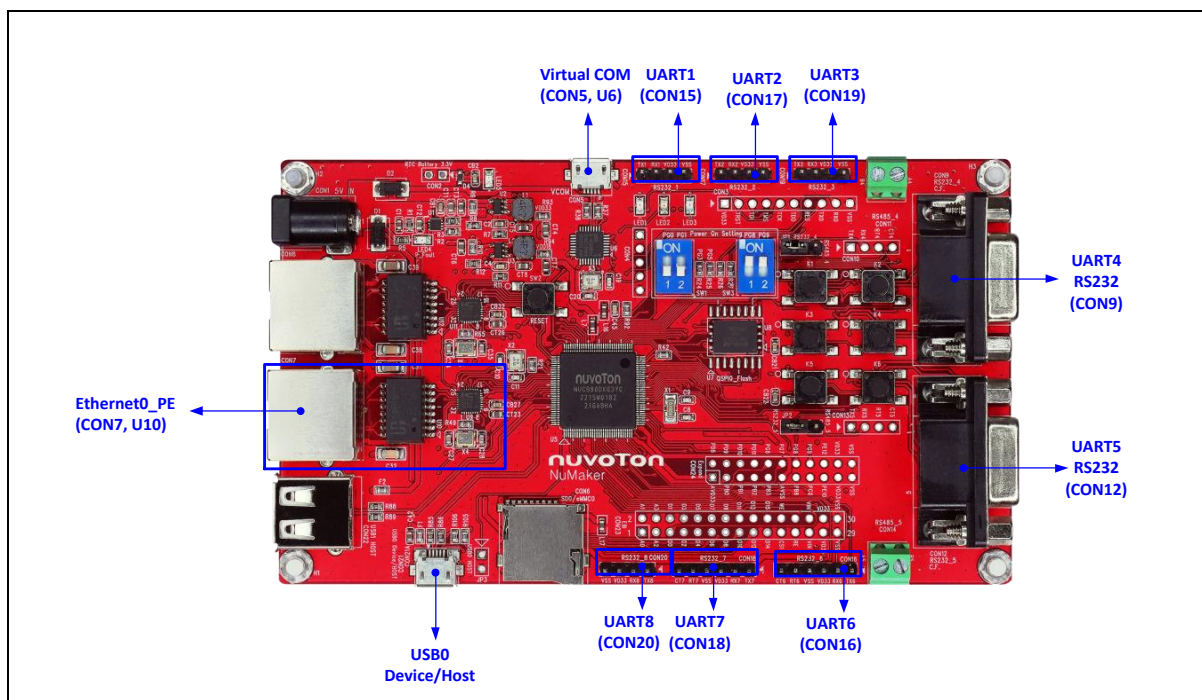


Figure 4-13 NuMaker-Server Board Setup

Then, use Terminal tool, such as PuTTY, Tera term, etc to open the serial COM port. The COM port configuration is baudrate 115200bps, 8-bit data length and no-parity.

User also need to ensure that the PC Ethernet port is connected to the Development Board Ethernet0 (or Ethernet1) port. PC Internet settings can refer to Figure 4-14.

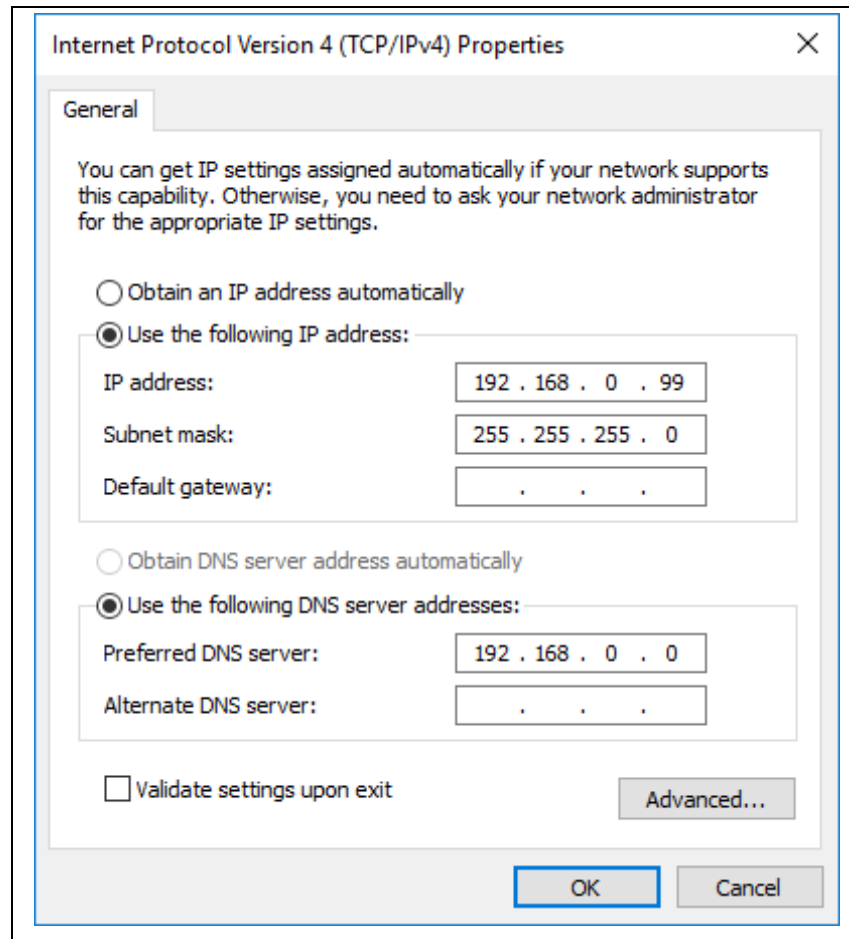


Figure 4-14 Serial COM Port Setting

Power ON or push the Reset key and boot system.

```
Freeing unused kernel memory: 5456K
[Mount JFFS2]: /dev/mtdblock0 --> /mnt/mtdblock0
nuc980-eth0 nuc980-eth0: eth0 is OPENED
nuc980-eth1 nuc980-eth1: eth1 is OPENED
random: arm-linux-light: uninitialized urandom read (8 bytes read, 7 bits
of entropy available)

BusyBox v1.22.1 (2016-02-03 14:11:04 CST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

~ #
```

Use Terminal tool to open serial COM port (from UART1 to UART8)

And use Terminal tool open TCP/IP connection. Ethernet 0 IP Address is 192.168.0.100, Port number from 50001 to 50008. Ethernet 1 IP Address is 192.168.10.100, Port number from 50001 to 50008. Where Transmitting and receiving of port numbers 50001~50008 maps to to UART1~8 respectively.

Below is an example transmit data from Ethernet to UART. When type "123" in TCP/IP connection window which port number is 50001. The UART1 serial COM port window will show "123".

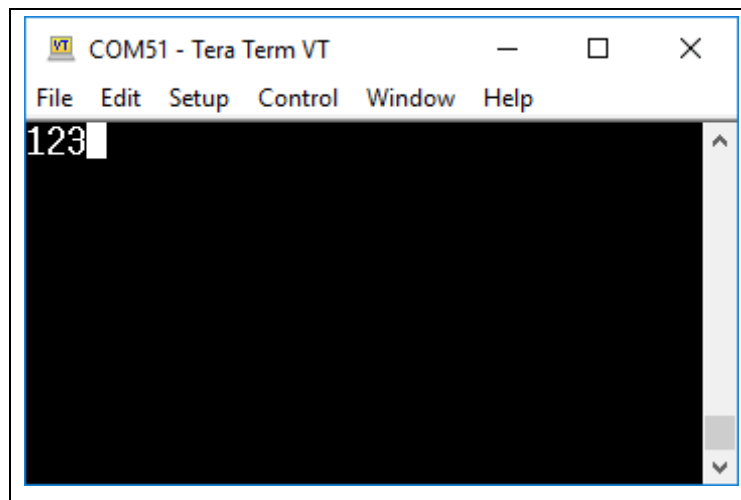


Figure 4-15 Serial COM Port

And below is an example with other direction, transmit data from UART to Ethernet. When type "123" in UART1 serial COM port. The TCP/IP connection window which port number is 50001 will show "123"

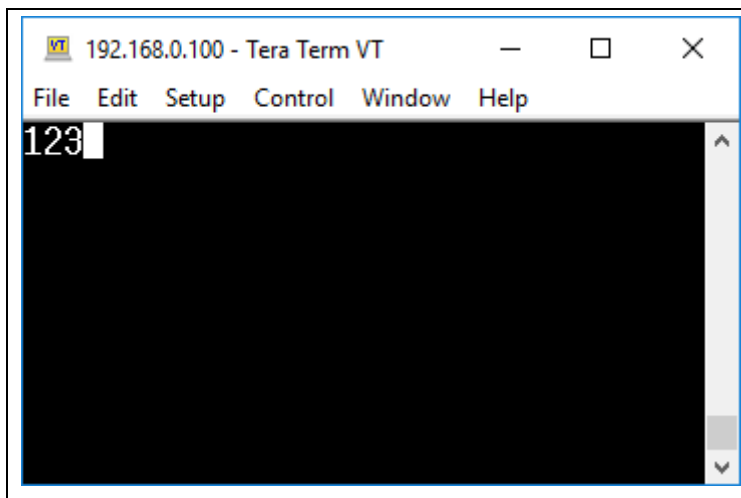


Figure 4-16 TCP/IP Connection Window

User can configure UART port via browser with following steps.

1. Use browser connect to <http://192.168.0.100> (Ethernet 0) or <http://192.168.10.100> (Ethernet 1).
2. Configure the UART settings. Including port, baudrate, data length, parity, stop bit, flow control, disable/enable RS485.
3. Press Submit

UART settings:

Port (1~8)

Baudrate bps

Data (8 or 7 bits)

Parity (0:none, 1:odd, 2:even)

Stop (1 or 2 bits)

Flow Control (0:none, 1:hardware, 2:Xon/Xoff)

RS485 (0:Disable, 1:Enable)

Figure 4-17 UART Setting Web Page

4.7 Make your own Embedded Linux

“**NUC980/NUC970 Linux Environment on VMware User Manual**” describes how to compile and modify your own Embedded Linux step by step. Please refer from [here](#) to get more details.

If it is your first time to use the NUC980/NUC970 Linux Environment on VMware, you must update the buildroot, kernel and uboot first (refer to chapter2.4 in **NUC980/NUC970 Linux Environment on VMware User Manual**) and choose the corresponding board configuration (refer to chapter2.5 in **NUC980/NUC970 Linux Environment on VMware User Manual**).

5 BLOCK DIAGRAM SCHEMATIC

5.1 GPIO List Schematic

PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION
PA0	UART1_RXD	PB0	ADC_AIN[0]	PC0	CAN2_TXD	PD2	QSPI0_SS0	PE0	RMII0_RXERR	PF0	RMII1_RXERR	PG0	EBI_ADDR0
PA1	UART1_TXD	PB1	ADC_AIN[1]	PC1	EBI_DATA0	PD3	QSPI0_CLK	PE1	RMII0_CRSDV	PF1	RMII1_CRSDV	PG1	EBI_ADDR1
PA2	UART6_CTS	PB2	ADC_AIN[2]	PC2	Col0	PD4	QSPI0_D0	PE2	RMII0_RXD1	PF2	RMII1_RXD1	PG3	EBI_ADDR3
PA3	UART6_RTS	PB3	EBI_ADDR2	PC3	Col1	PD5	QSPI0_D1	PE3	RMII0_RXD0	PF3	RMII1_RXD0	PG5	CFG[3]
PA4	UART6_RXD	PB4	ADC_AIN[3]	PC4	UART3_TXD	PD6	QSPI0_D2	PE4	RMII0_REFCLK	PF4	RMII1_REFCLK	PG6	CFG[5]
PA5	UART6_TXD	PB5	UART7_RXD	PC5	EBI_DATA3	PD7	QSPI0_D3	PE5	RMII0_TXEN	PF5	RMII1_TXEN	PG7	PWM10
PA6	EBI_nCS1	PB6	UART7_RTS	PC6	UART3_RXD	PD8	SPI0_SS0	PE6	RMII0_TXD1	PF6	RMII1_TXD1	PG8	PWM11
PA7	EBI_nWE	PB7	UART7_TXD	PC7	EBI_DATA4	PD9	SPI0_CLK	PE7	RMII0_TXD0	PF7	RMII1_TXD0	PG9	PWM12
PA8	EBI_nRE	PB8	UART7_CTS	PC8	EBI_DATA5	PD10	SPI0_DO	PE8	RMII0_TXD0	PF8	RMII1_TXD0	PG11	CFG[8]
PA9	UART2_RXD	PB13	CAN2_RXD	PC9	SD0_CMD	PD11	SPI0_DI	PE9	RMII0_MDIO	PF9	RMII1_MDIO	PG12	PWM13
PA10	UART2_TXD			PC10	EBI_DATA6	PD12	UART4_TXD	PE10	I2C0_SDA	PF10	LED3	PG13	CFG[9]
PA11	UART8_RXD			PC11	SD0_DATA0	PD13	UART4_RXD	PE11	USB0_VBUSVLD	PF11	UART0_RXD	PG14	UART5_CTS
PA12	UART8_TXD			PC12	EBI_DATA7	PD14	UART4_RTS	PE12	I2C0_SCL	PF12	UART0_TXD	PG15	UART5_RTS
				PC13	SD0_DATA1	PD15	UART4_CTS						JTAG0_TCK
				PC14	EBI_DATA8								UART5_RXD
				PC15	SD0_DATA2								JTAG0_TMS
					EBI_DATA9								UART5_TXD
					SD0_DATA3								JTAG0_TDI
					EBI_DATA10								JTAG0_NTRST
					LED1								LED1
					EBI_DATA11								
					SD0_nCD								
					EBI_DATA12								
					Row0								
					EBI_DATA13								
					Row1								
					EBI_DATA14								
					Row2								
					EBI_DATA15								

nuvoTon Technology Corp.

Title: **NK-980ETH2P**

Size: A Document Number: **GPIO List** Rev: 2

Date: Tuesday, May 31, 2022 Sheet: 2 of 14

Figure 5-1 GPIO List Schematic

5.2 Power Schematic

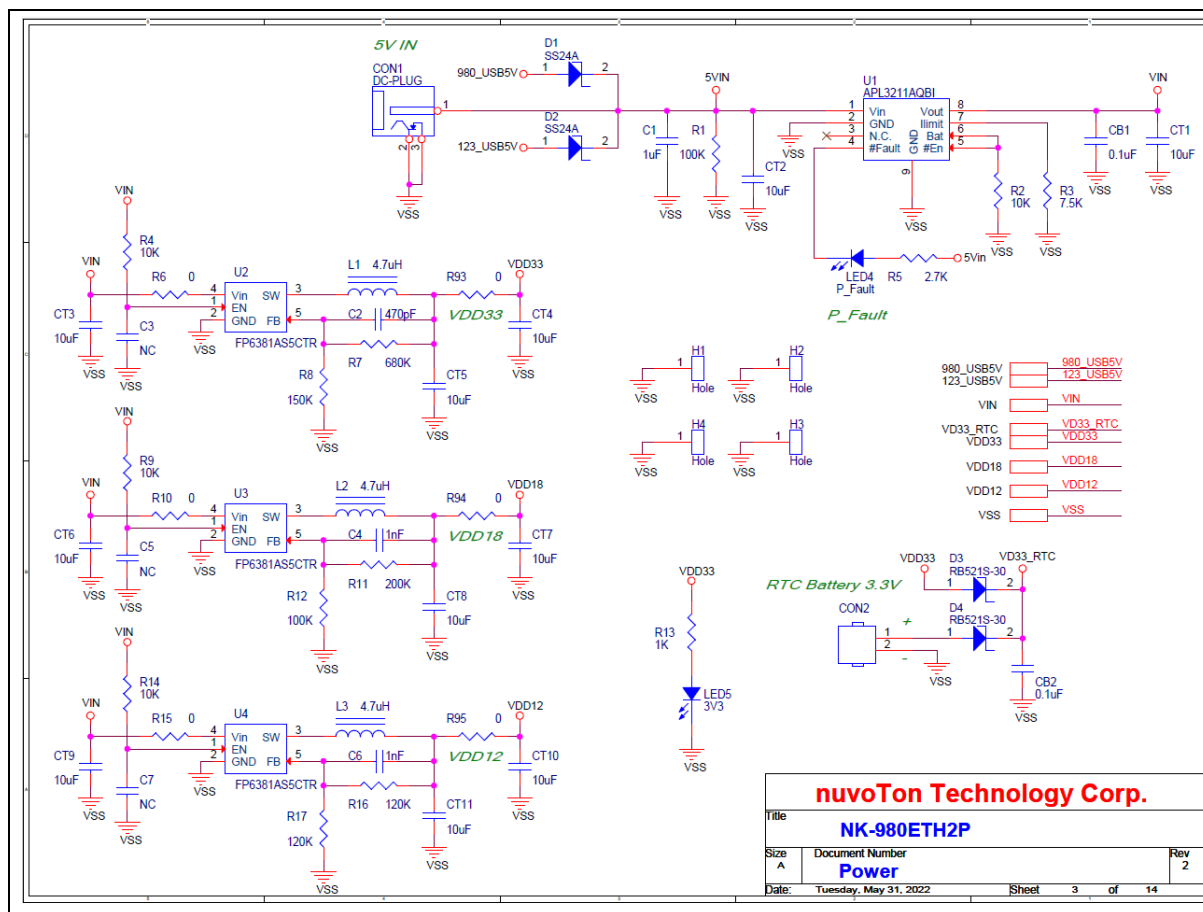


Figure 5-2 Power Schematic

5.3 NUC980DK Schematic

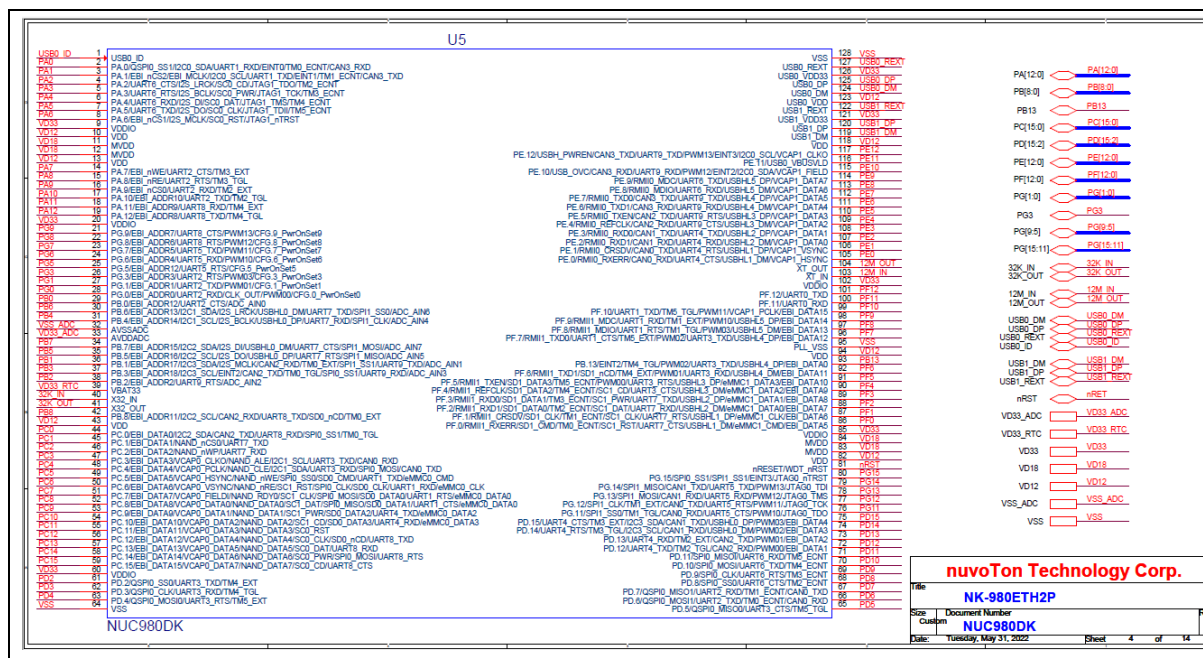


Figure 5-3 NUC980DK Schematic

Power On Setting

PG0 PG1 Booting from
 ON(0) ON(0) USB
 ON(0) OFF(1) SD/eMMC
 OFF(1) ON(0) NAND Flash
 OFF(1) OFF(1) QSPI Flash 0

PG3 Watch Dog
 ON(0) OFF
 OFF(1) ON

PG5 UART0 Message
 ON(0) ON
 OFF(1) OFF

PG9 PG8 If booting from QSPI
 ON(0) ON(0) SPI-NAND, 1-bit
 ON(0) OFF(1) SPI-NAND, 4-bit
 OFF(1) ON(0) SPI-NOR, 4-bit
 OFF(1) OFF(1) SPI-NOR, 1-bit

PG9 PG8 If booting from SD/eMMC
 OFF(1) OFF(1) SD/eMMC0 EC booting
 Others Sbl/eMMC1 EP booting

PC13 Row0
 PC14 Row1
 PC15 Row2
 PC1 Col0
 PC2 Col1

PC13 CB20 0.1uF
 PC14 CB21 0.1uF
 PC15 CB22 0.1uF

PG15 LED_YELLOW
 PG3 LED_GREEN
 PG5 LED_RED
 PG9:8 LED_YELLOW
 PG15:11 LED_GREEN
 nRESET LED_RED
 nRST LED_RED
 VDD33 LED_YELLOW
 VDD12 LED_GREEN
 VSS LED_RED

nuvoTon Technology Corp.
 NK-980ETH2P
 Configure
 Tuesday, May 31, 2022
 6 of 14

Figure 5-5 Configure Schematic

5.6 NUC123ZD4AN0 Schematic

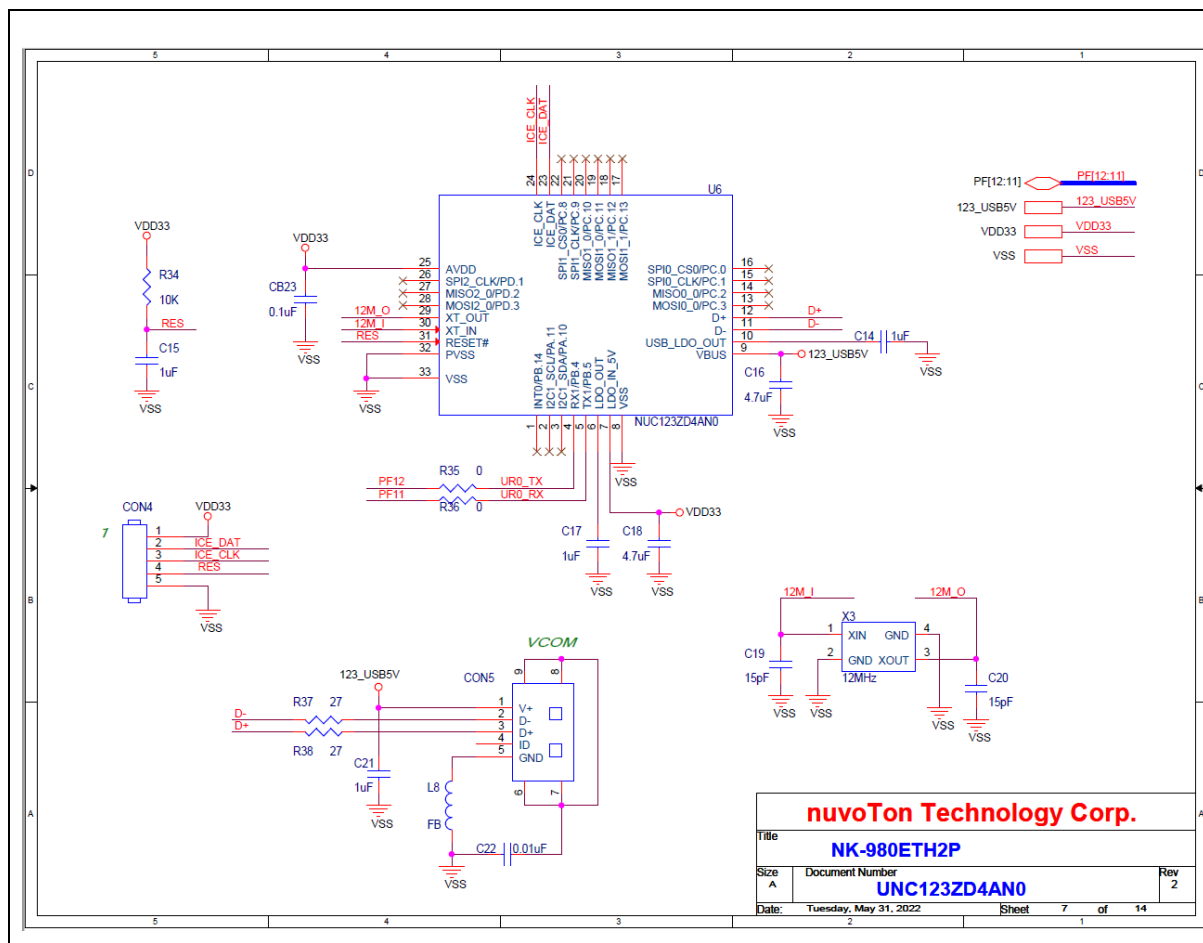


Figure 5-6 NUC123ZD4AN0 Schematic

5.7 Memory Schematic

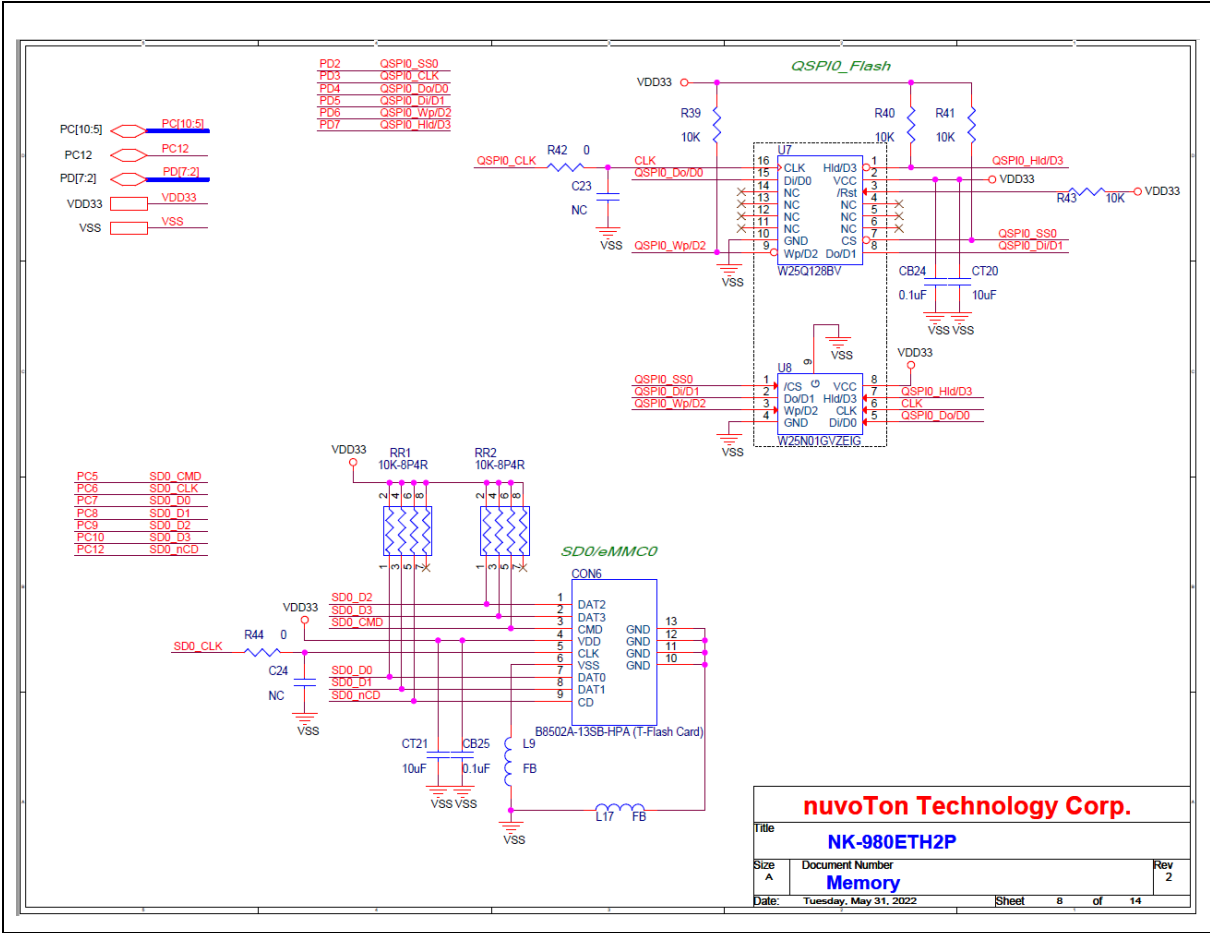


Figure 5-7 Memory Schematic

5.8 RMII_PE Schematic

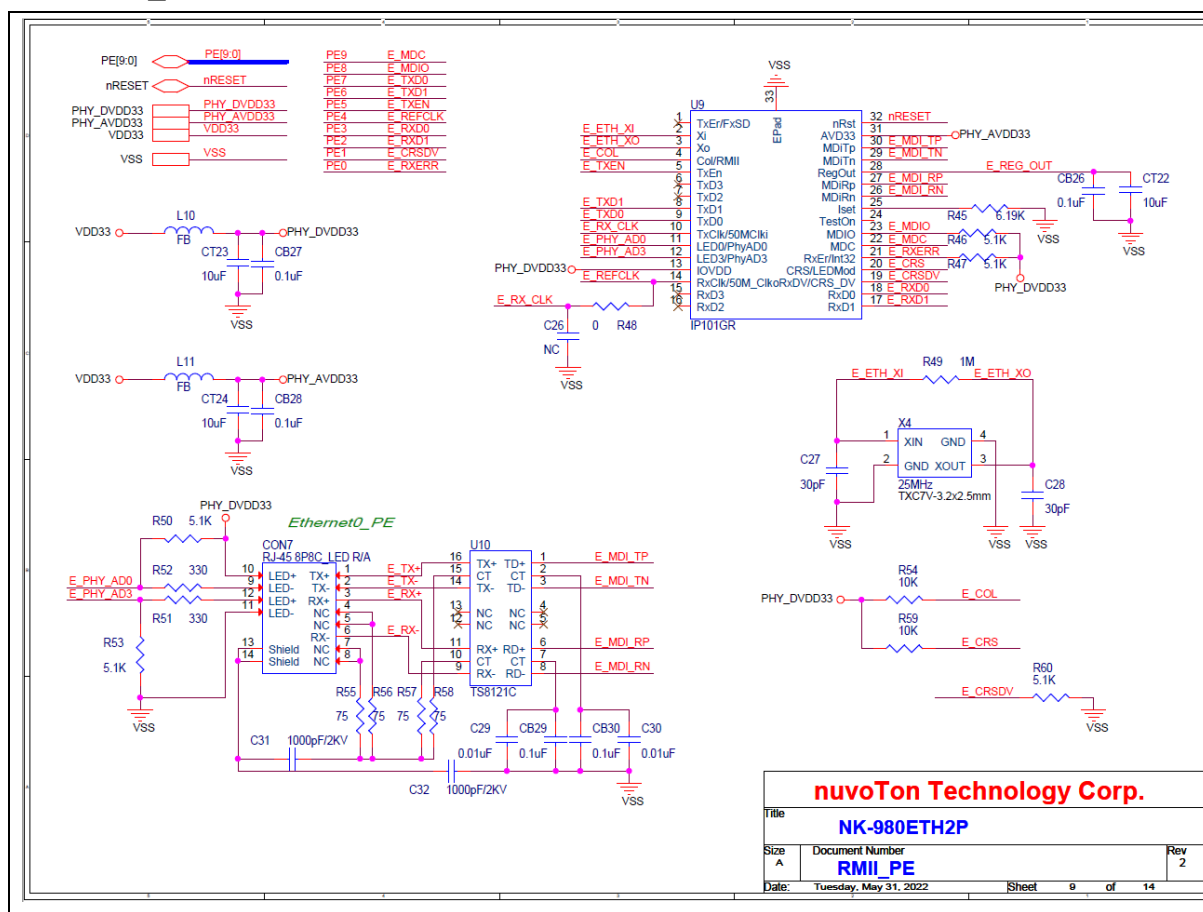


Figure 5-8 RMII_PE Schematic

5.9 RMII_PF Schematic

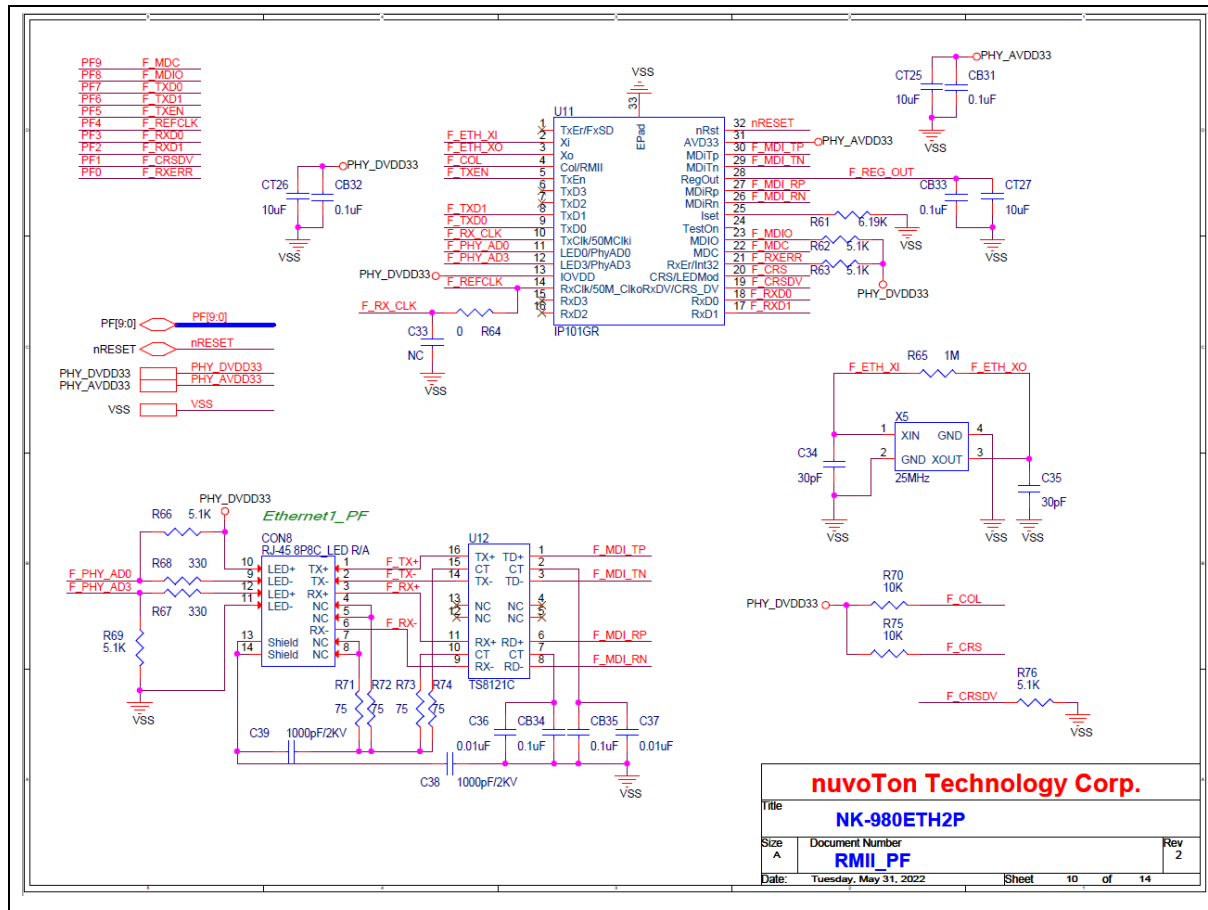


Figure 5-9 RMII_PF Schematic

5.10 UART_A Schematic

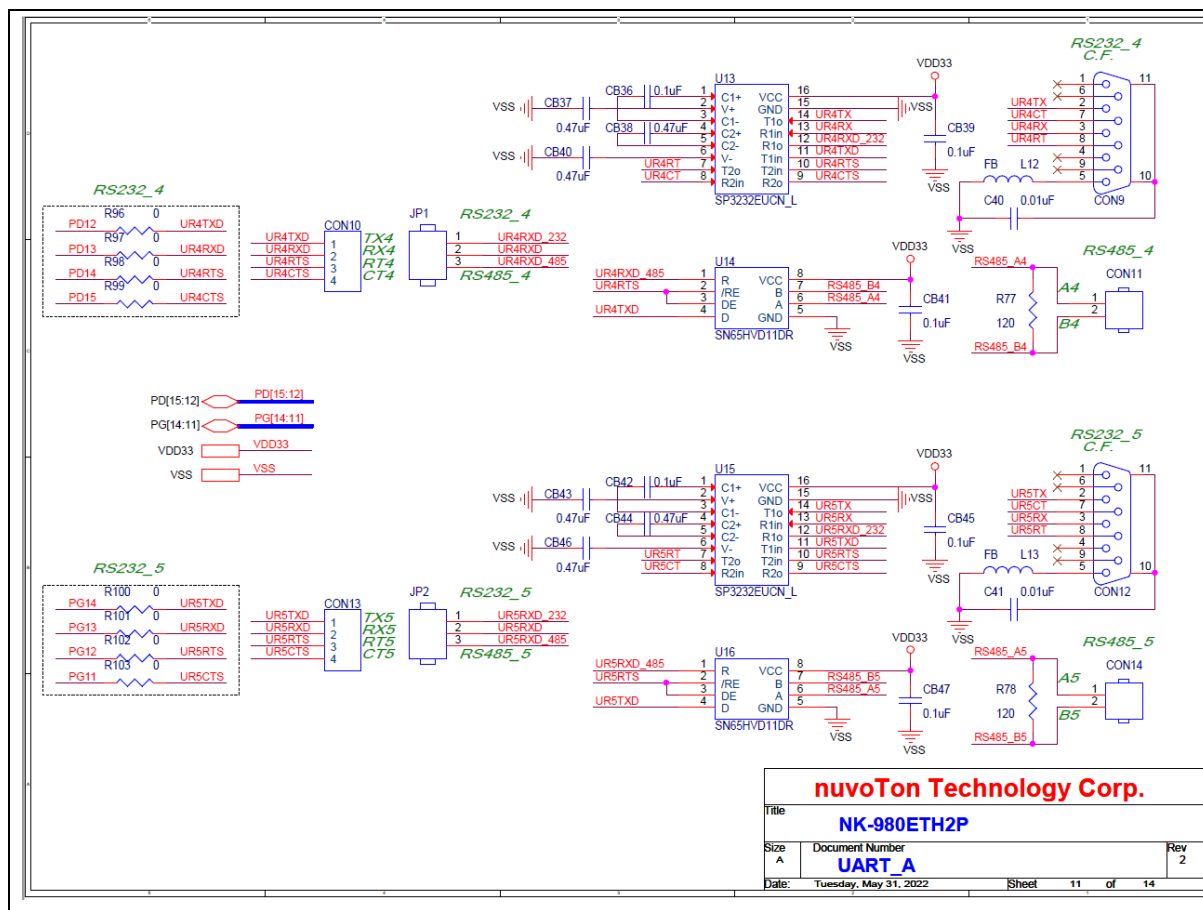


Figure 5-10 UART_A Schematic

5.11 UART_B Schematic

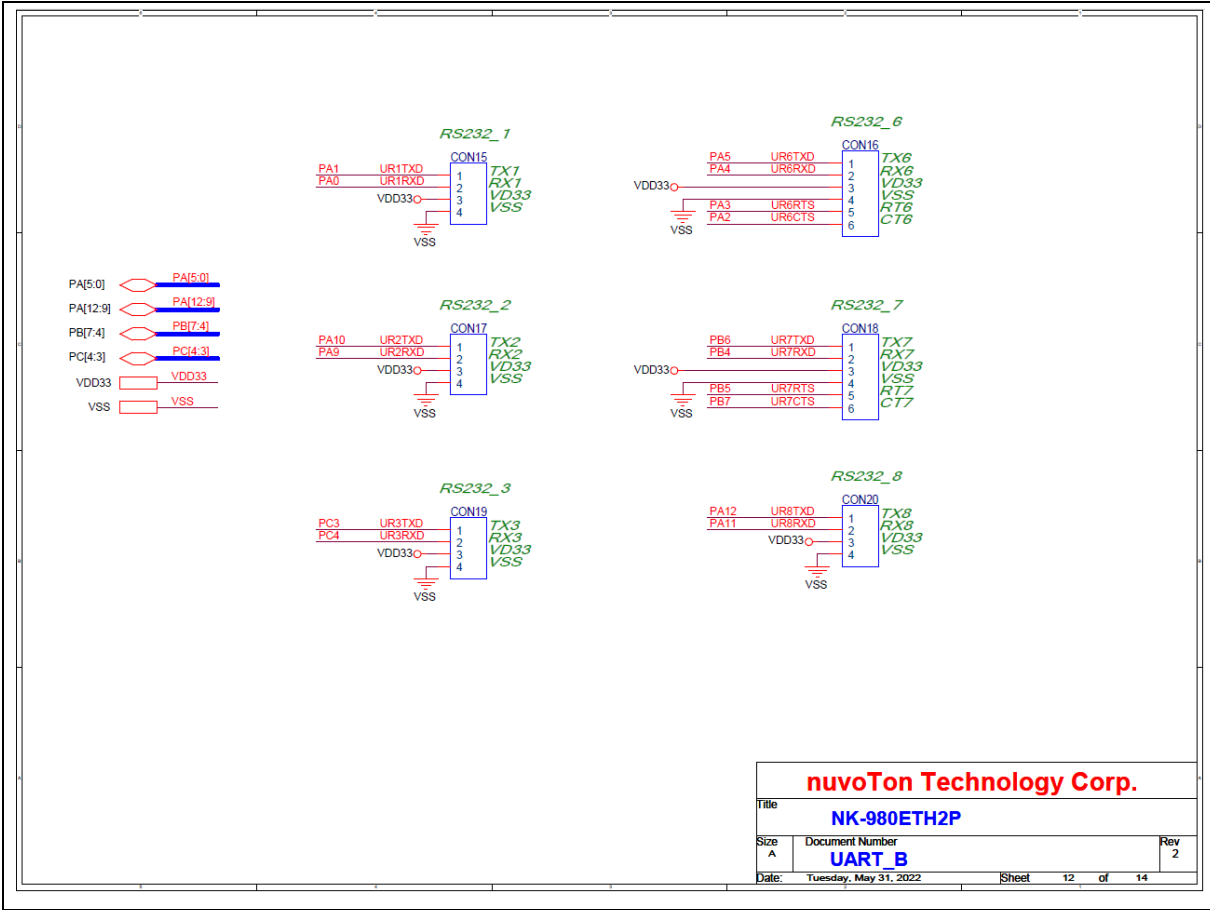


Figure 5-11 UART_B Schematic

5.12 USB Schematic

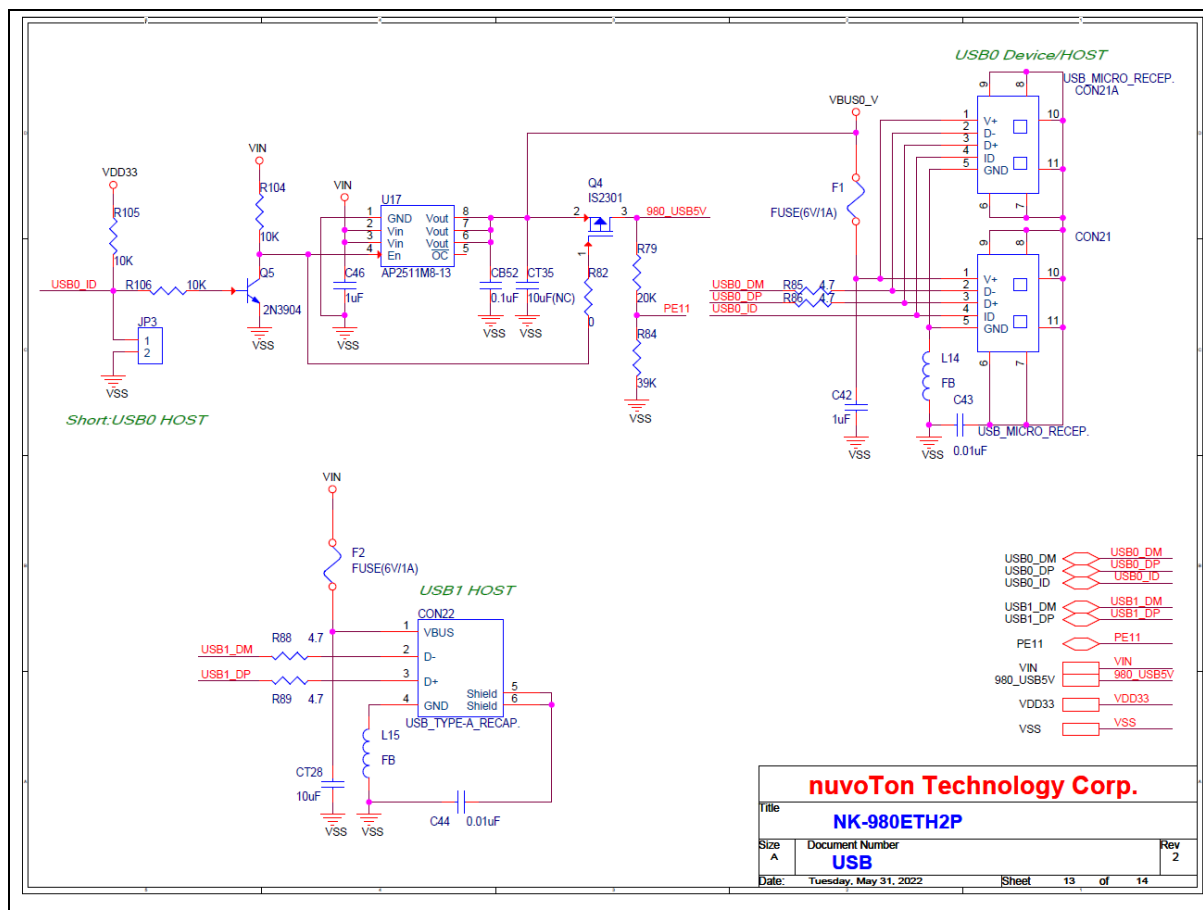


Figure 5-12 USB Schematic

5.13 Expand Schematic

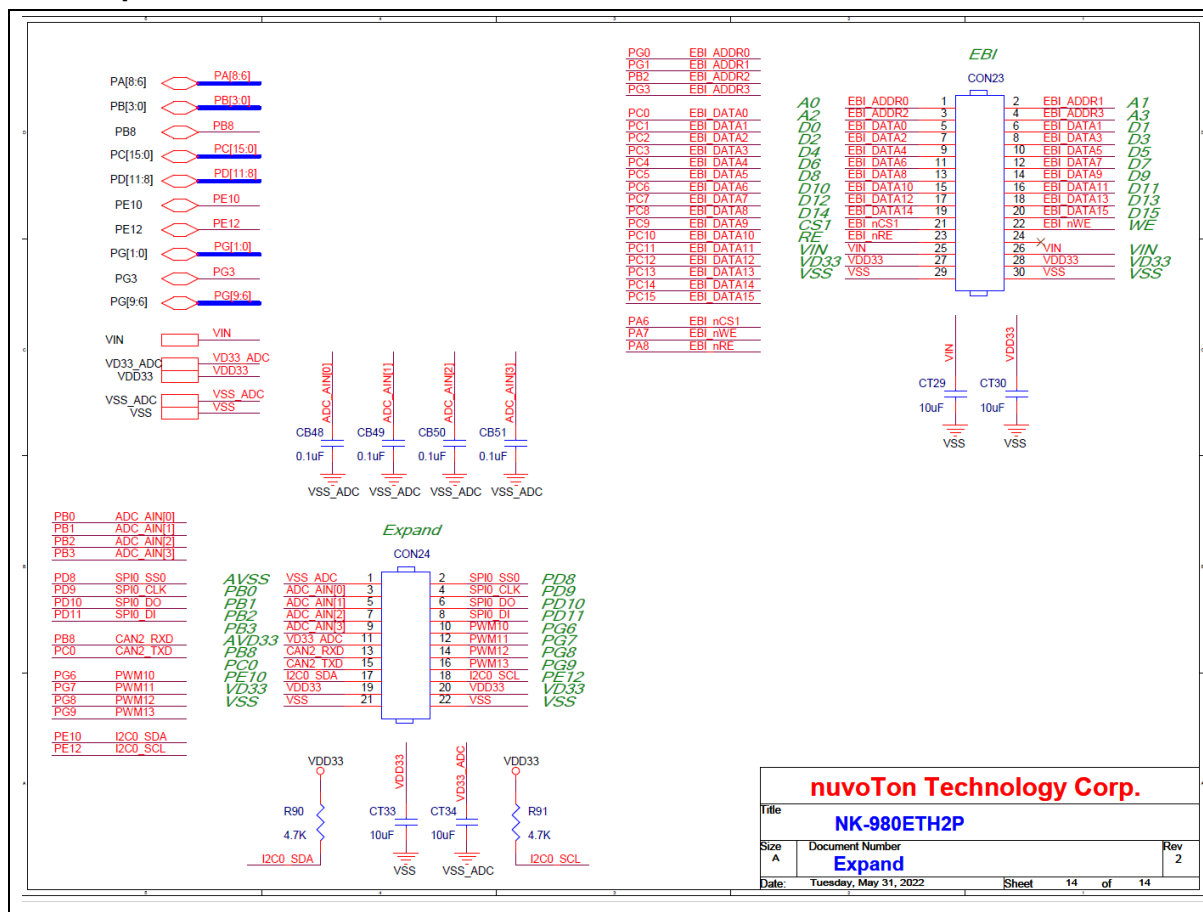


Figure 5-13 Expand Schematic

5.14 PCB Placement

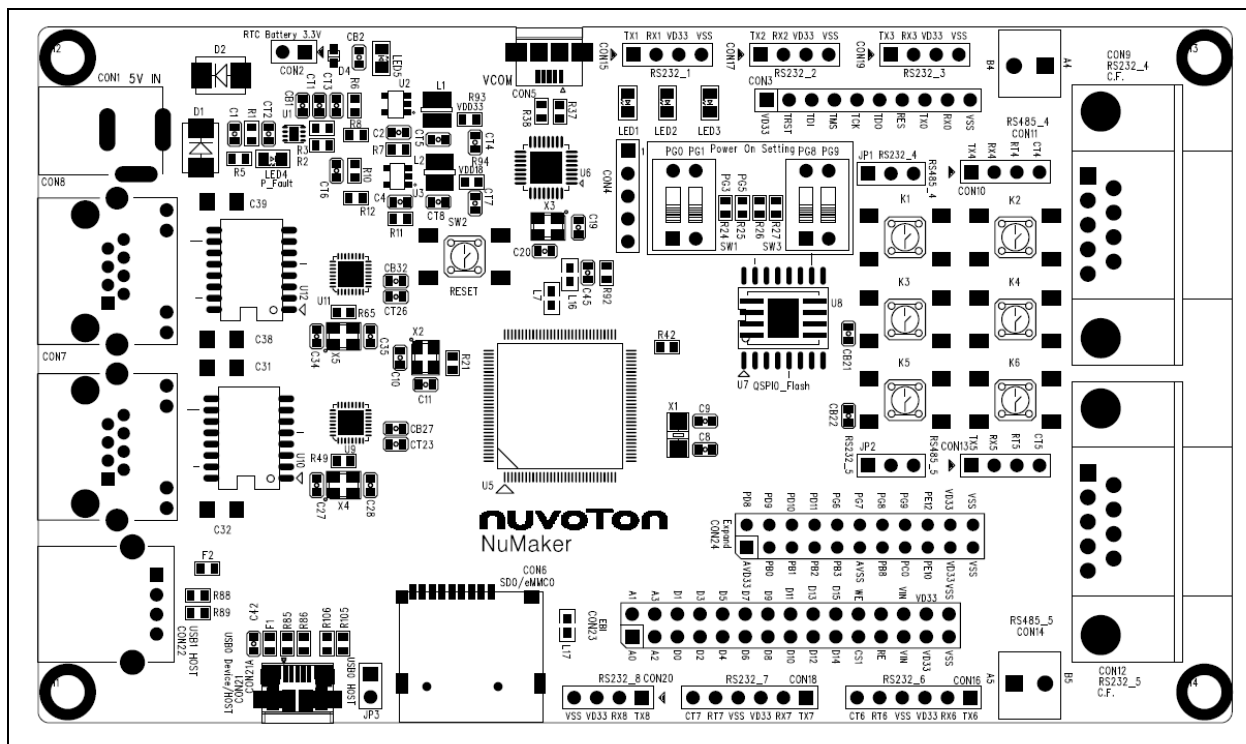


Figure 5-14 Front PCB Placement

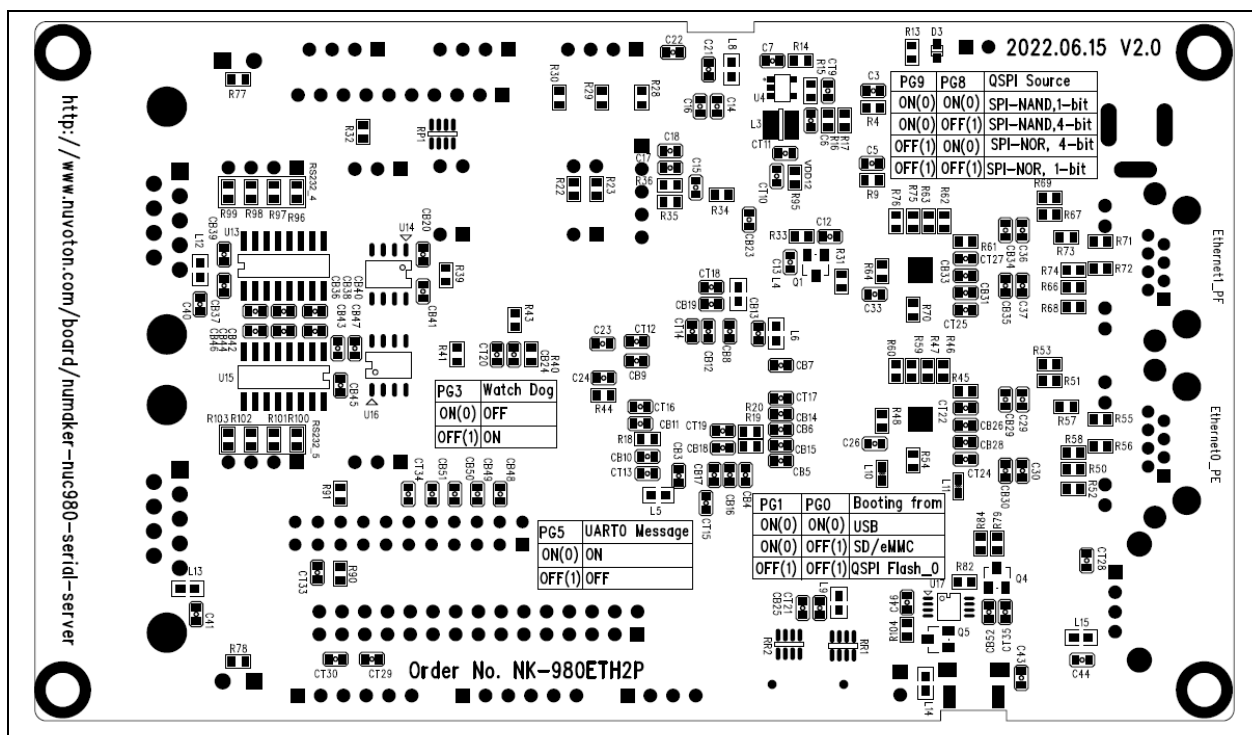


Figure 5-15 Back PCB Placement

6 REVISION HISTORY

Date	Revision	Description
2020.05.22	1.00	Initial version
2020.07.22	1.20	For board versionV1.2
2023.07.31	1.21	For board version V2.0 SOC change NUC980DK61YC to NUC980DK63YC Modified Chapter 4
2024.03.04	1.22	Modified Section 4.3 and Section 4.5 content of b Updated Figure 4-12

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.*