

NuMaker RTU NUC980 User Manual

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 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	11
4	Quick Start	12
4.1	BSP Download	12
4.2	Driver Installation	12
4.3	Hardware Setting	15
4.4	Programing Kernel and U-Boot to SPI NAND Flash	17
5	Block Diagram Schematic	21
5.1	GPIO List Schematic	21
5.2	Power Schematic	22
5.3	NUC980DR Schematic	23
5.4	Power Filter Schematic	24
5.5	Configure Schematic	25
5.6	NUC123ZD4AN0 Schematic	26
5.7	Memory Schematic	27
5.8	RMII_PF connector Schematic	28
5.9	RS485 and CAN Schematic	29
5.10	USB Schematic	30
5.11	PCB Placement	31
6	REVISION HISTORY	32

List of Figures

Figure 1-1 NuMaker RTU NUC980 Development Board	5
Figure 3-1 Front View of NuMaker RTU NUC980.....	7
Figure 3-2 Rear View of NuMaker RTU NUC980	11
Figure 4-1 Nuvoton USB Driver Installation Setup.....	12
Figure 4-2 Nuvoton USB Driver Installation	14
Figure 4-3 Hardware Setting	15
Figure 4-4 Nuvoton VCOM.....	16
Figure 4-5 NuWriter Setting	17
Figure 4-6 Program u-boot.....	18
Figure 4-7 Program uimage	19
Figure 4-8 Program environment	20
Figure 5-1 GPIO List Schematic	21
Figure 5-2 Power Schematic.....	22
Figure 5-3 NUC980DR Schematic	23
Figure 5-4 Power Filter Schematic.....	24
Figure 5-5 Configure Schematic	25
Figure 5-6 NUC123ZD4AN0 Schematic	26
Figure 5-7 Memory Schematic	27
Figure 5-8 RMII_PF connector Schematic.....	28
Figure 5-9 RS485 and CAN Schematic	29
Figure 5-10 USB Schematic.....	30
Figure 5-14 Front PCB Placement	31
Figure 5-15 Back PCB Placement	31

List of Tables

Table 4-1 Power On Setting..... 15

1 OVERVIEW

This document provides a quick start guide for the NuMaker RTU NUC980 Development Board. Users can understand both software and hardware configurations for the NuMaker RTU NUC980. The platform provides Linux OS and plenty of industrial control protocol for users to implement the Ethernet control applications in a very short time.

The NuMaker RTU NUC980 board uses NUC980DR61YC microprocessor run up to 300 MHz with built-in 64MB DDR2 memory, 16 KB I-cache, 16 KB D-cache and MMU, 16 KB embedded SRAM and 16.5 KB IBR (Internal Boot ROM) for system booting from USB and SPI flash, all functions of the NUC980DR61YC are placed on the board, including peripheral interfaces such as SPI Flash memory, UART, 10/100 Mb Ethernet MAC controller, high speed USB (Device, Host), JTAG, RS485 and CAN transceiver controller. Users can use it to develop and verify applications to emulate the real behavior.

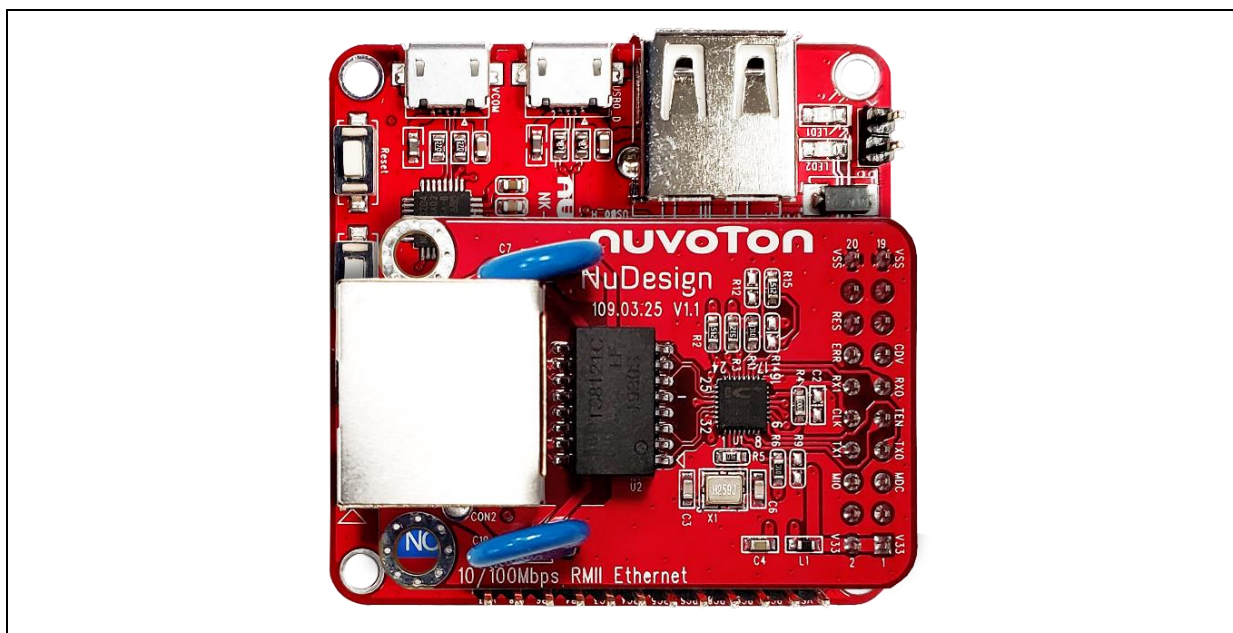


Figure 1-1 NuMaker RTU NUC980 Development Board

2 FEATURES

- NUC980DR61YC: LQFP64 pin MCP package with DDR2 (64 MB), which can run up to 300MHz operating speed
- SPI Flash: Normal mode system booting or data storage, use W25Q256JV SPI-NOR (256 M-Bit)
- UART0: Connected to Virtual COM port for system development, debug message output
- Peripheral interface connector, including UART, SPI, I2C
- JTAG interface provided for software development
- RJ45 port (Ethernet0) connector
- UART8-RS485 header with transceiver controller interface
- CAN3 header with transceiver controller interface
- 2 sets of LED for status indication
- 1 sets of user-configurable push button keys
- 1 sets of system-reset push button keys
- USB port-0 that can be used as Device/HOST to support pen drives, keyboards, mouse and printers
- 3.3V I/O power, 1.8V Memory power and 1.2V core power

3 HARDWARE CONFIGURATION

3.1 Front View

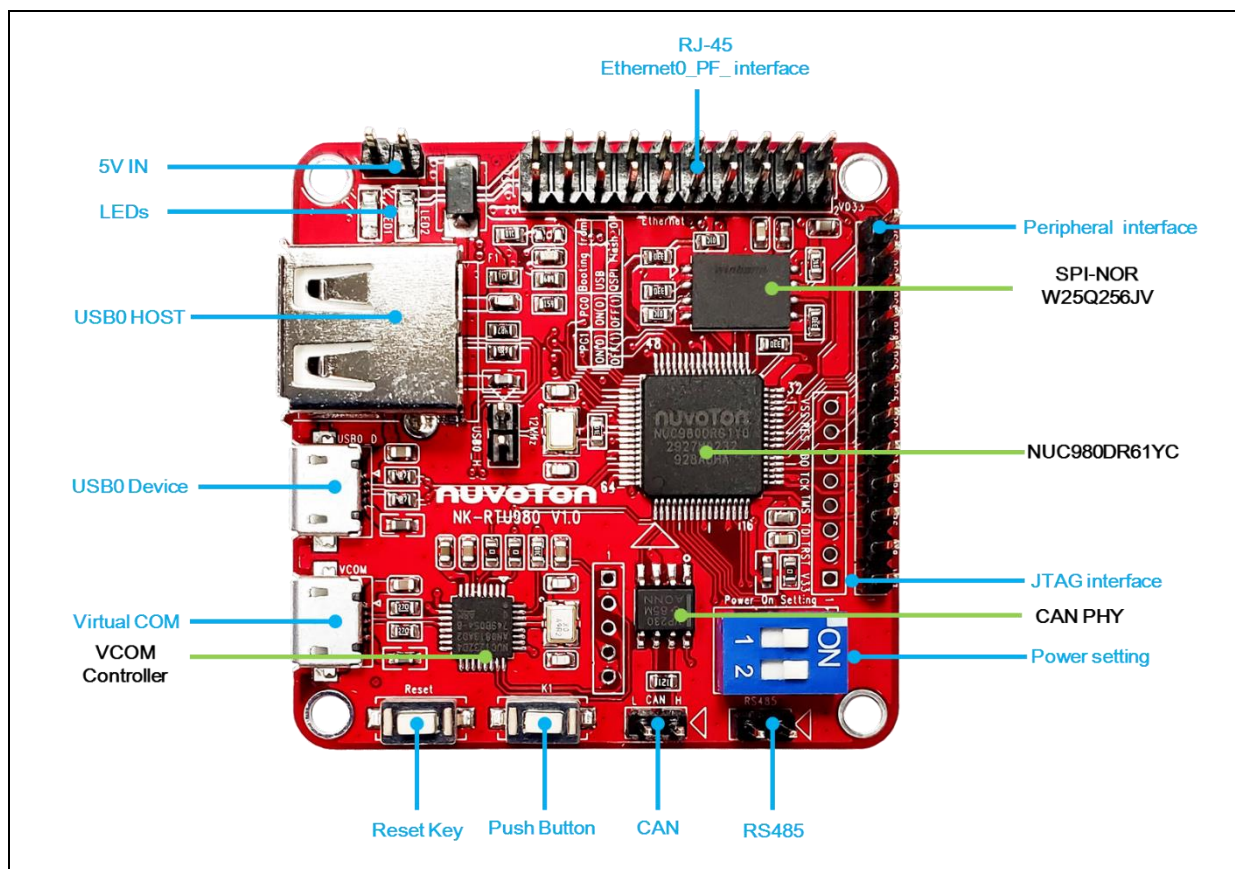


Figure 3-1 Front View of NuMaker RTU NUC980

Figure 3-1 shows the main components from the front view of NuMaker RTU NUC980 Development Board

- +5V In (J1): Power 5V input

Power Model	CON2 USB Port (Micro-B)	CON4 USB Port (Micro-B)	J1
Model 1	Connect to PC	-	-
Model 2	-	Connect to PC	-
Model 3	-	-	VDD5V Input

- System Reset (SW3): System will be reset if the SW3 button is pressed
- Virtual COM (CON2, U8): NUC123ZD4AN0 microcontroller (U8), USB micro-B connector (CON2) to PC, for debug message output

- User indication LEDs (LED1, LED2):

LED	Color	GPIO pin of NUC980
LED1	Green	PC11
LED2	Green	PC3

- SPI NOR Flash (U5): Use Winbond W25Q256JV 256M Bit (U5) for system booting, supporting normal mode

- JTAG interface (J1/NC)

Connector	GPIO pin of NUC980	Function
J1.1	-	VDD33
J1.2	GPA6	nTRST
J1.3	GPA5	TDI
J1.4	GPA4	TMS
J1.5	GPA3	TCK
J1.6	GPA2	TDO
J1.7	-	nRESET
J1.8	-	VSS

- USB0 Device/HOST (CON3, JP4): USB0 Device/HOST Micro-B connector, By JP4 status or defined by the ID pin of the USB cable

- User Key SW (K1)

Key	GPIO pin of NUC980
K1	GPC15

- Ethernet port interface(CON1)

Connector	GPIO pin of NUN980	Function
CON1.1	-	VDD33
CON1.2	-	VDD33
CON1.3	-	NC
CON1.4	-	NC
CON1.5	GPF9	F_MDC
CON1.6	GPF8	F_MDIO

CON1.7	GPF7	F_TXD0
CON1.8	GPF6	F_TXD1
CON1.9	GPF5	F_TXEN
CON1.10	GPF4	F_REFCLK
CON1.11	GPF3	F_RXD0
CON1.12	GPF2	F_RXD1
CON1.13	GPF1	F_CRSDV
CON1.14	GPF0	F_RXERR
CON1.15	-	NC
CON1.16	-	nRESET
CON1.17	-	NC
CON1.18	-	NC
CON1.19	-	VSS
CON1.20	-	VSS

- Power on setting (SW1, R15, R16)

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

- CAN (JP2, U7): SN65HVD230 transceiver controller of CAN(U7), CAN header(JP2) connect to device for communication
- Peripheral user interface(J2), including I2C, SPI, UART

Connector	GPIO pin of NUC980	Function
J2.1	-	VDD33
J2.2	-	VDD18
J3.3	GPB6	I2C1_SDA
CON1.4	GPB4	I2C1_SCL
CON1.5	GPC3	GPIO
CON1.6	GPC4	SPI0_DO
CON1.7	GPC5	SPI0_SS0

CON1.8	GPC6	SPI0_CLK
CON1.9	GPC8	SPI0_DI
CON1.10	GPC9	UART4_TXD
CON1.11	GPC10	UART4_RXD
CON1.12	-	VSS

- SOC CPU: NUC980DR61YC (U4)

3.2 Rear View

Figure 3-2 shows the main components from the rear view of NuMaker RTU NUC980 Development Board

- VCOM ICE interface: ICE Controller NUC123ZD4AN0 (U6), USB connector (CON3) to PC Host

Connector	Pin Name	Functions
CON3.1	VDD33	DC 3.3V
CON3.2	ICE_DAT	Serial Wired Debugger Data
CON3.3	ICE_CLK	Serial Wired Debugger Clock
CON3.4	RST#	VCOM Chip Reset, Active Low.
CON3.5	VSS	Power Ground

- RS485 (JP1, U6): SN65HVD11DR transceiver controller of RS485(U6), RS485 header(JP1) connect to device for communication

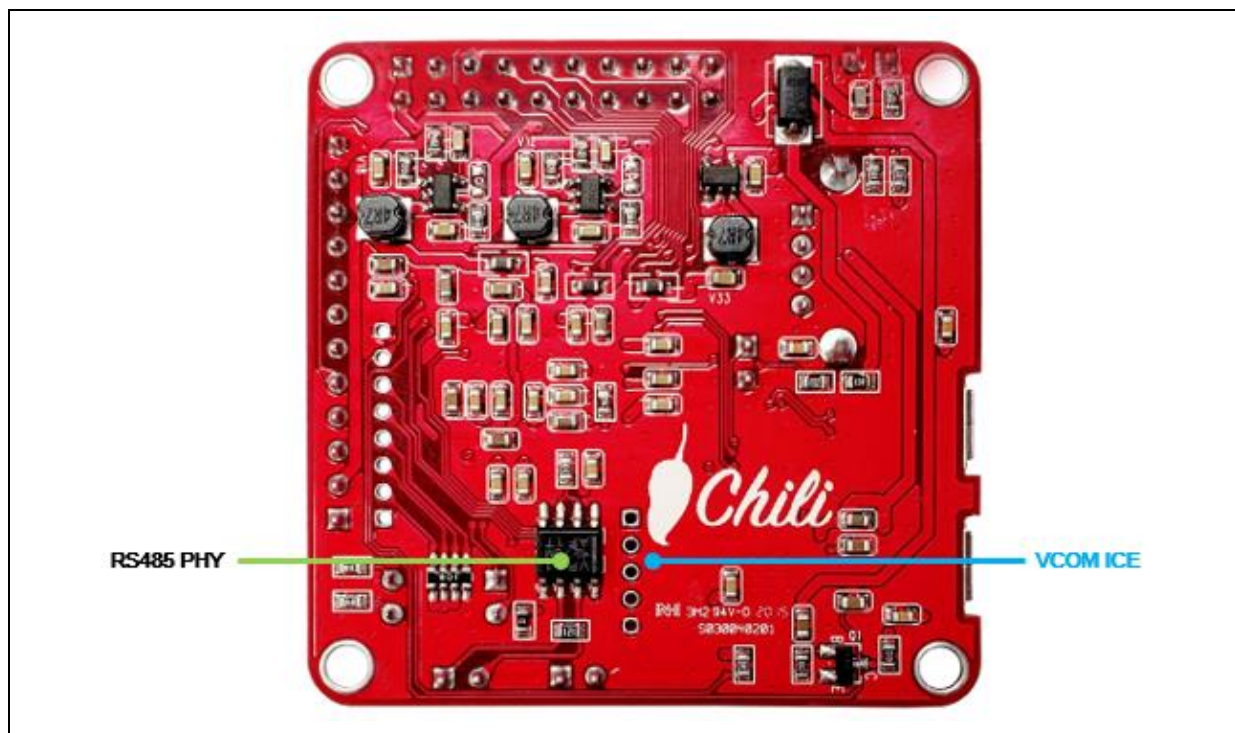


Figure 3-2 Rear View of NuMaker RTU NUC980

4 QUICK START

4.1 BSP Download

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

Please visit nuvoTon's NuMicro™ website <https://www.nuvoton.com/products/iot-solution/iot-platform/numaker-server-nuc980/?group=Software&tab=2> to download the "NUC980_Linux-4.4_BSP_v1.02.001". Run the "WinUSB4NuVCOM.exe" before the USB cable is plugged in. The "WinUSB4NuVCOM.exe" can be found in the "Tool" directory. Power on the NUC980 Series MPU EVB and plug the USB cable into PC, the Windows shall find a new device and then request to install its driver. Simply follow the installation and optional steps to install USB Driver, included VCOM driver.

4.2 Driver Installation

The programming tool requires a Nuvoton USB driver to be installed on PC first. Please follow the steps below to install the WinUSB driver.

Run the "WinUSB4NuVCOM.exe" before the USB cable is plugged in. The "WinUSB4NuVCOM.exe" can be found in the "Tool" directory. Power on the NUC980 Series MPU EVB and plug the USB cable into PC, the Windows shall find a new device and request to install the driver.

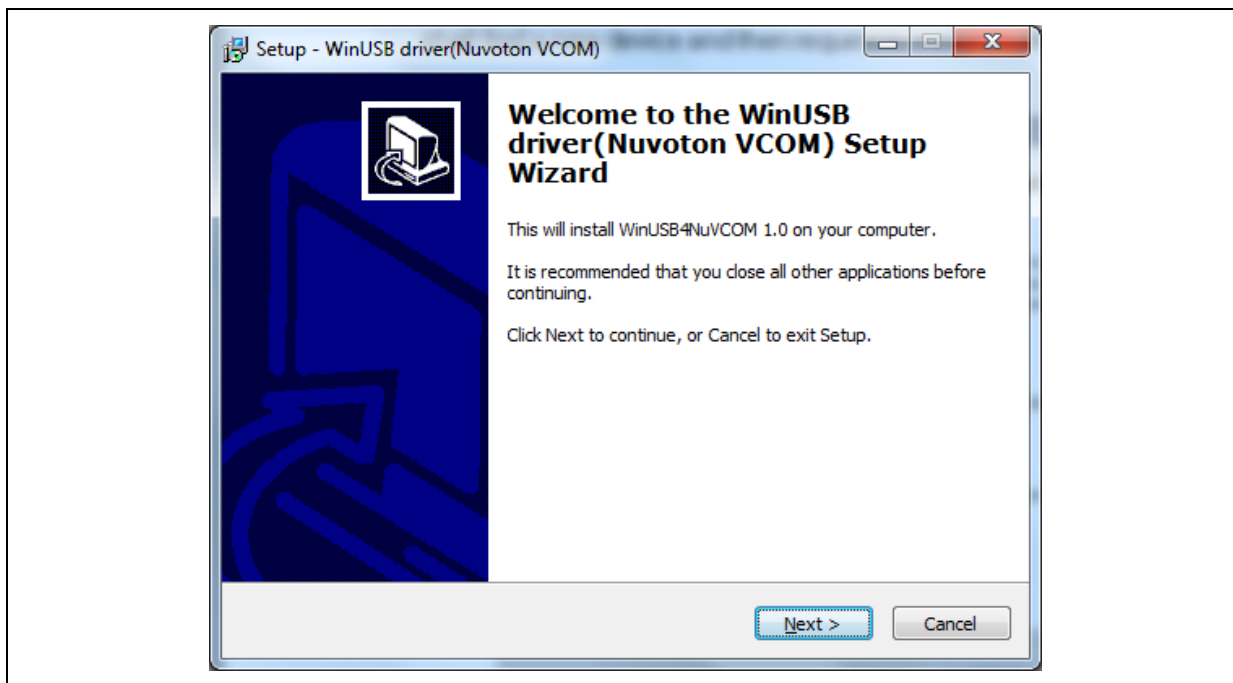
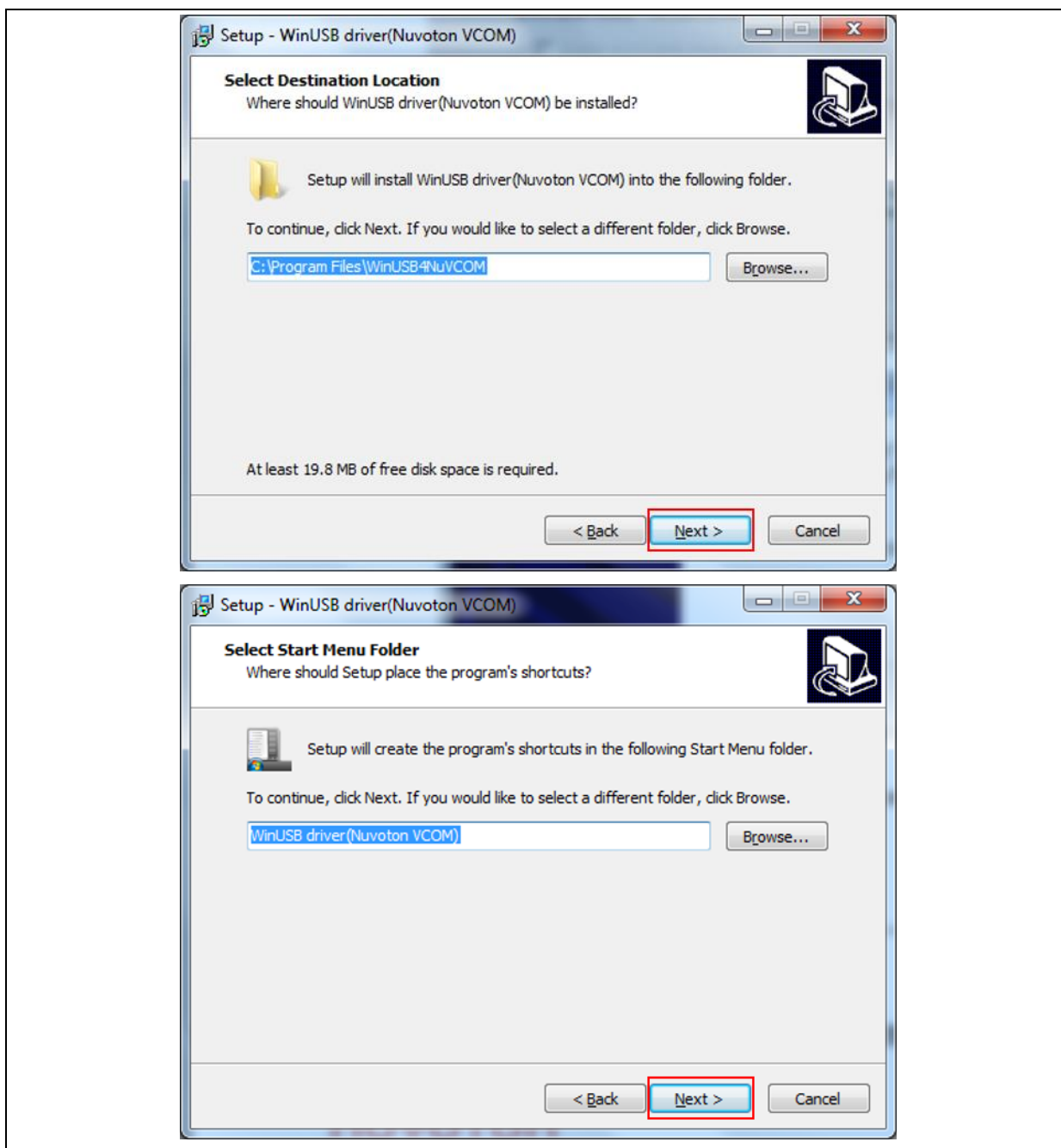


Figure 4-1 Nuvoton USB Driver Installation Setup

Click "Next". The WinUSB driver Setup Wizard will be started.



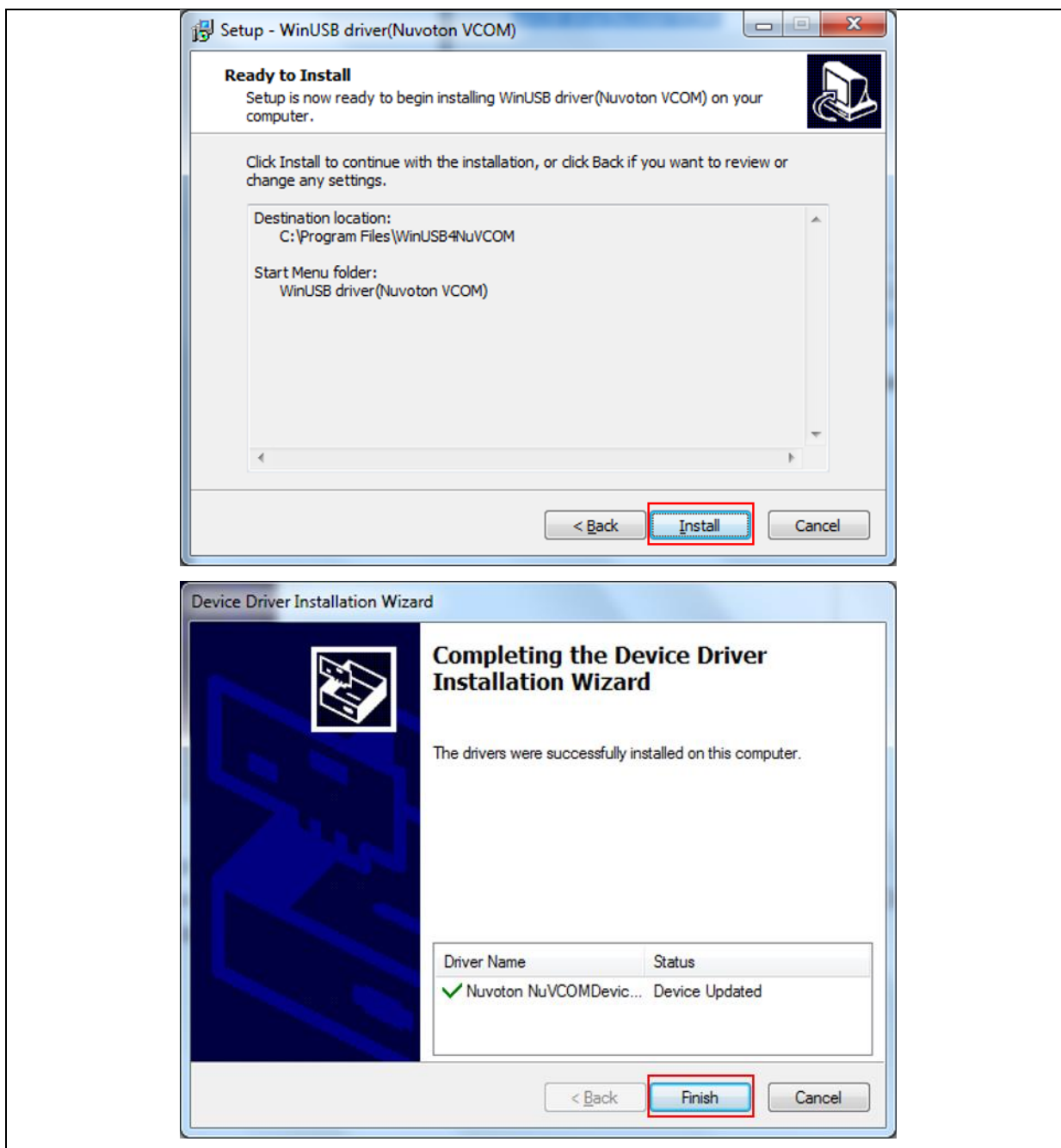


Figure 4-2 Nuvoton USB Driver Installation

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.

Please download USB CDC driver" TomatoUSB CDC driver" from Nuvoton's official webpage, executing the "NuvotonCDC_V1.00.001_Setup.exe" to install the driver:

<https://www.nuvoton.com/products/iot-solution/iot-platform/numaker-tomato/?group=Software&tab=2>

4.3 Hardware Setting

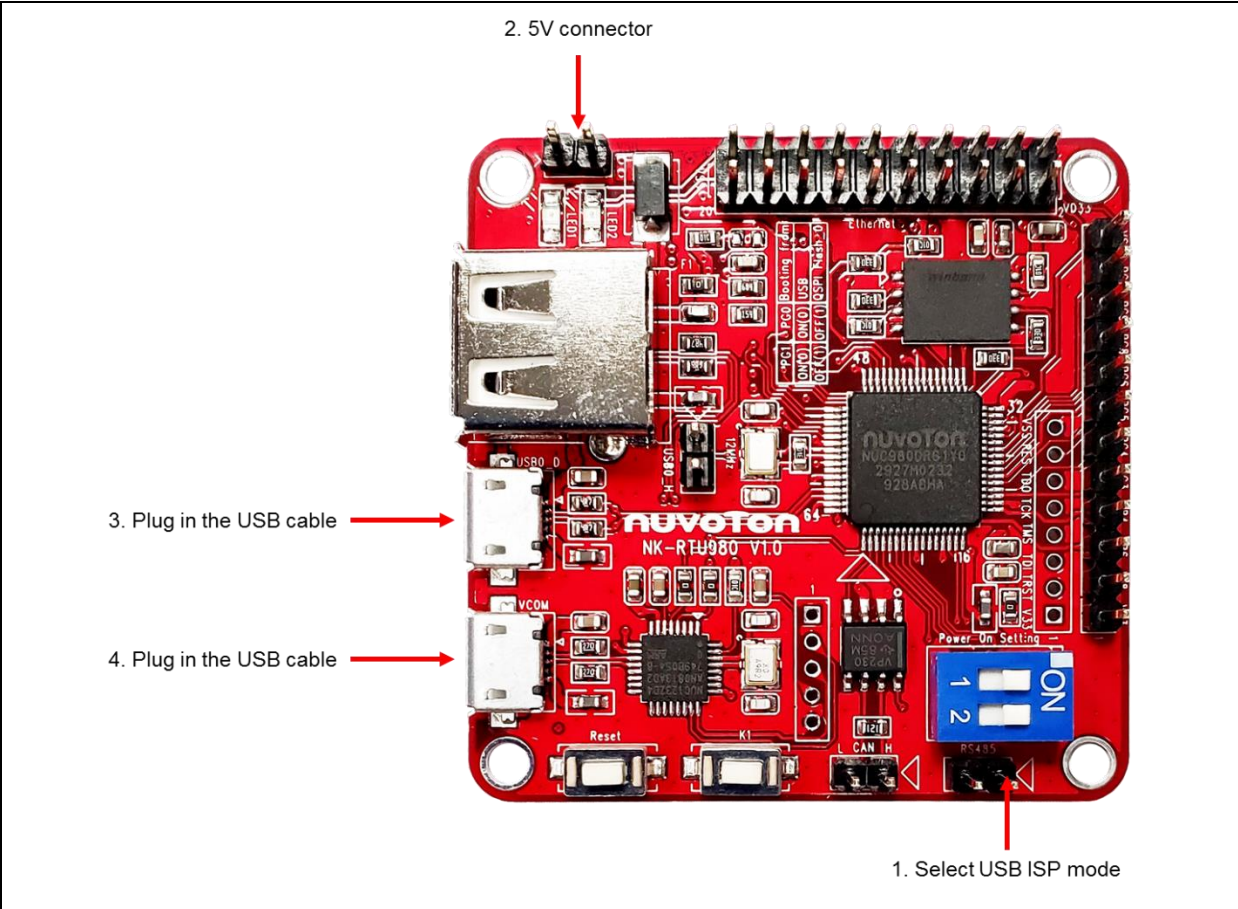


Figure 4-3 Hardware Setting

1. NuMaker RTU NUC980 provides jumpers (SW1) to select boot-up conditions. The jumpers (SW1) ON to select USB ISP mode.

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

Table 4-1 Power On Setting

2. 5V input connector
3. Plug in the USB cable

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

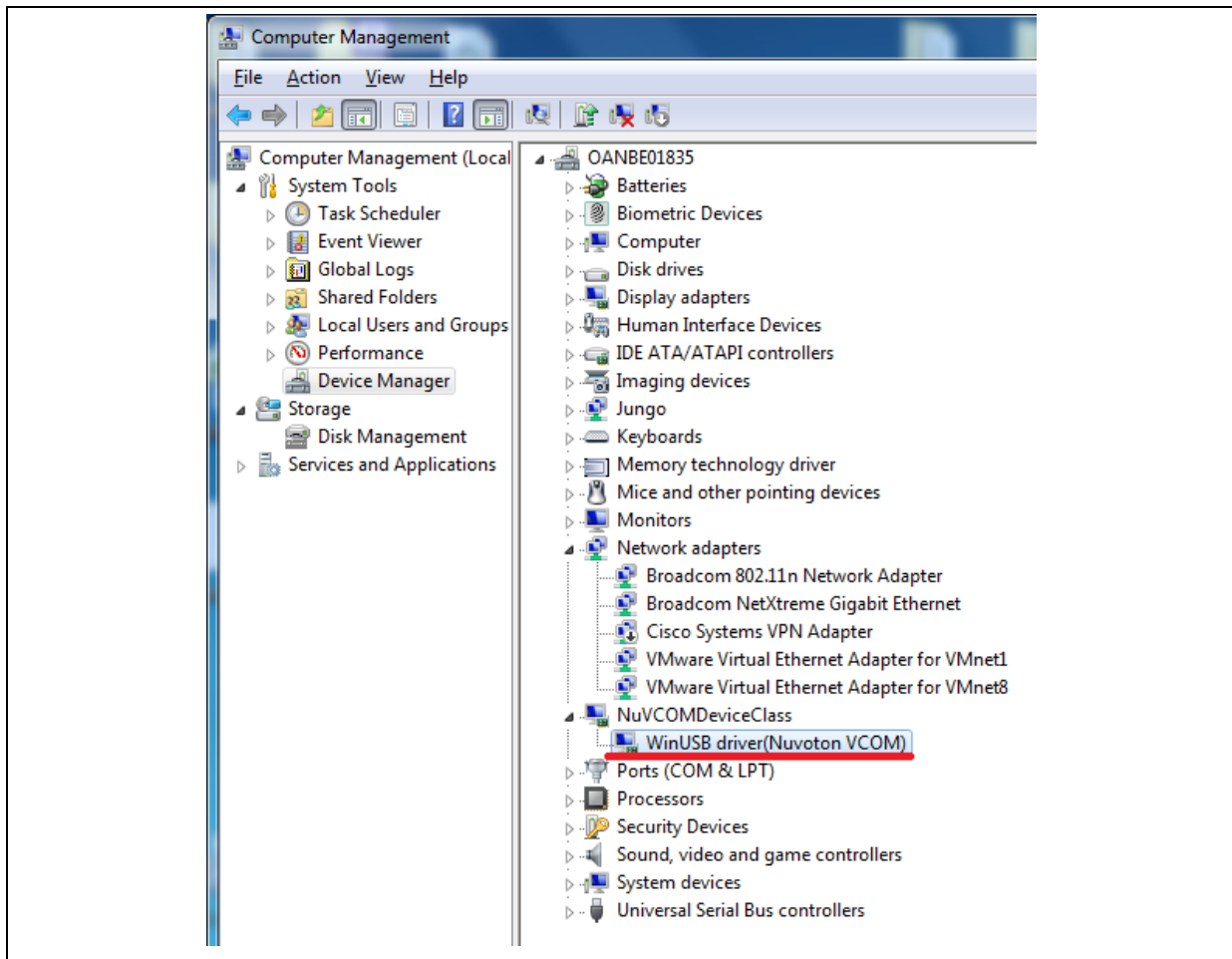


Figure 4-4 Nuvoton VCOM

4. Plug in the USB cable

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.

4.4 Programing Kernel and U-Boot to SPI NAND Flash

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 4-3). 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 “**NUC980DR61YC.ini**”. And then, press “**Continue**” button.

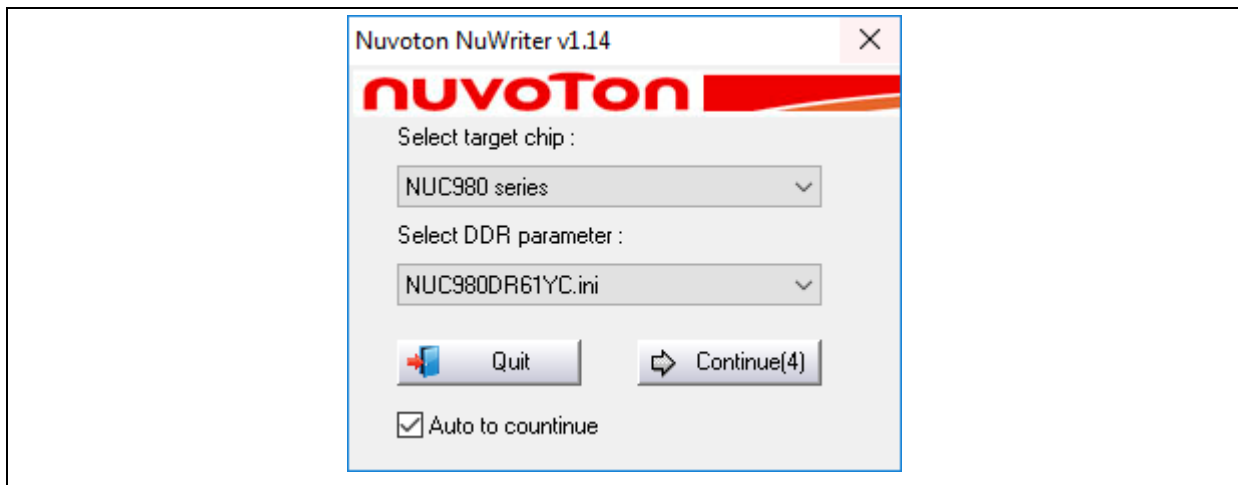


Figure 4-5 NuWriter Setting

4. According to Figure 4-6, following the steps below to program u-boot.bin:
 - a. Select the “**SPI**” type.
 - b. Fill in the image information :
 - Image Name: u-boot.bin
 - Image Type: Loader
 - Image execute address: 0xe00000
 - 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.

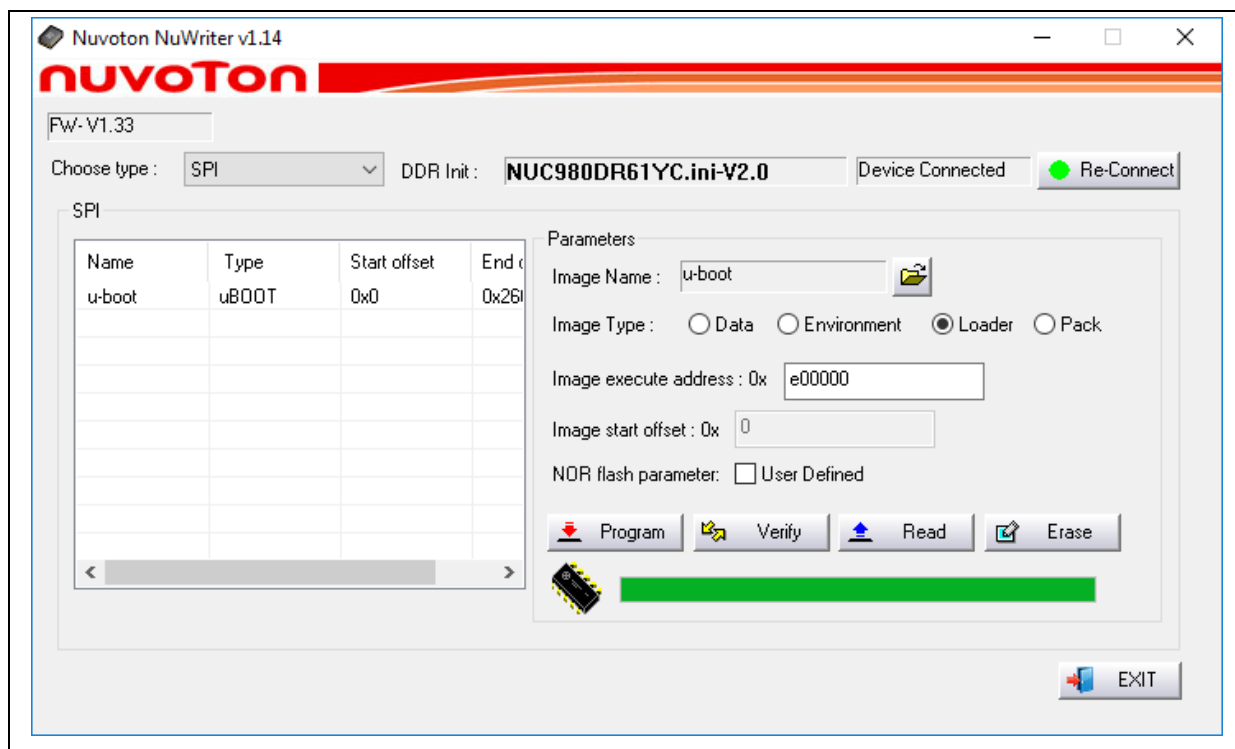


Figure 4-6 Program u-boot

5. According to Figure 4-7, following the steps to program kernel image:
 - a. Select the "SPI" type.
 - b. Fill in the image information :
 - Image Name: uimage
 - Image Type: Data
 - Image execute address: 0x200000
 - 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.

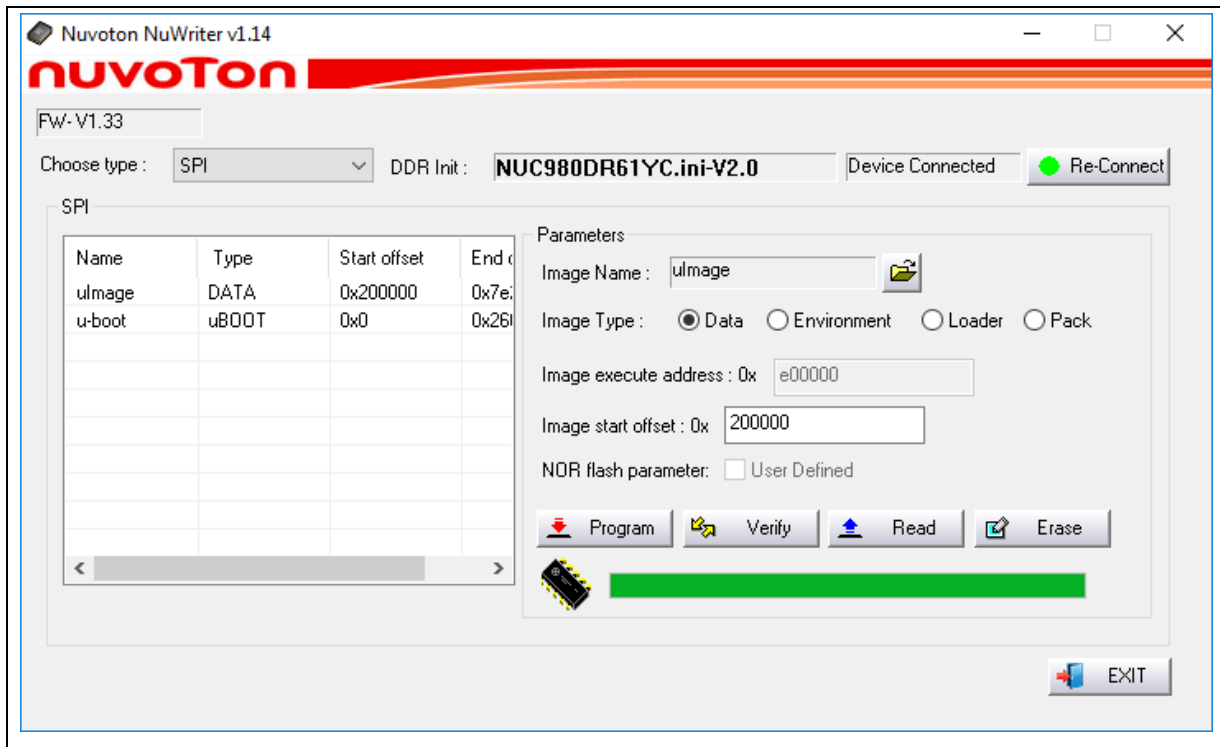


Figure 4-7 Program uimage

6. According to Figure 4-8, following the steps below to program environment:
 - a. Select the “**SPI**” 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.

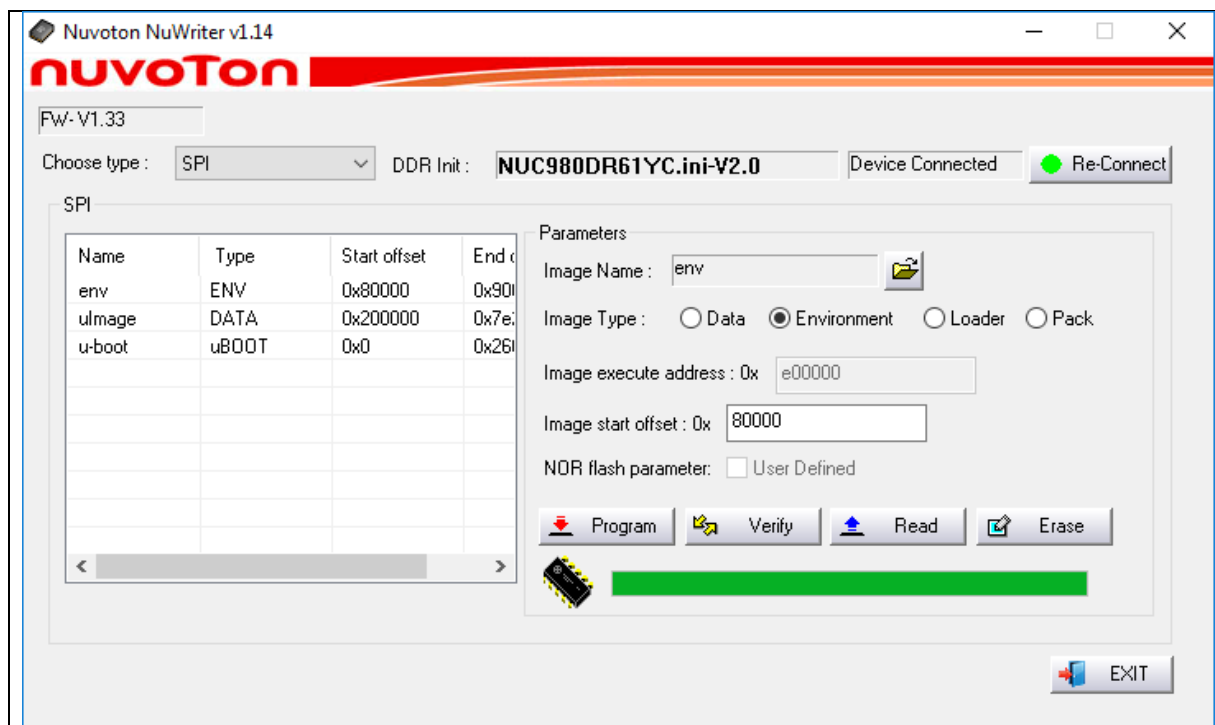


Figure 4-8 Program environment

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

For more details about kernel image and u-boot, please refer to "NUC980_970 Linux environment on VMware User Manual" from Nuvoton website.

URL: <https://www.nuvoton.com/products/iot-solution/iot-platform/numaker-server-nuc980/?group=Document&tab=2>

5 BLOCK DIAGRAM SCHEMATIC

5.1 GPIO List Schematic

PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION	PIN	FUNCTION
PA0	CAN3_RXD	PE4	I2C1_SCL	PC3	LED_G	PD2	QSPI0_SS0	PE11	USB0_VBUSVLD	PF0	RMII1_RXERR	PG0	CFG[0]
PA1	CAN3_TXD	PE6	I2C1_SDA	PC4	SPI0_DO	PD3	QSPI0_CLK			PF1	RMII1_CRSDV	PG1	CFG[1]
PA2	JTAG1_TDO			PC5	SPI0_SS0	PD4	QSPI0_DO			PF2	RMII1_RXD1		
PA3	JTAG1_TCK			PC6	SPI0_CLK	PD5	QSPI0_DI			PF3	RMII1_RXD0		
PA4	JTAG1_TMS			PC8	SPI0_DI					PF4	RMII1_REFCLK		
PA5	JTAG1_TDI			PC9	UART4_TXD					PF5	RMII1_TXEN		
PA6	JTAG1_nTRST			PC10	UART4_RXD					PF6	RMII1_TXD1		
				PC11	LED_G					PF7	RMII1_TXD0		
				PC12	UART8_TXD					PF8	RMII1_MDIO		
				PC13	UART8_RXD					PF9	RMII1_MDC		
				PC14	UART8_RTS					PF11	UART0_RXD		
				PC15	button					PF12	UART0_TXD		

nuvoTon Technology Corp.

Title: **NK-RTU980**

Size: A Document Number: **GPIO List** Rev: 1.0

Date: Monday, April 06, 2020 Sheet: 2 of 11

Figure 5-1 GPIO List Schematic

5.2 Power Schematic

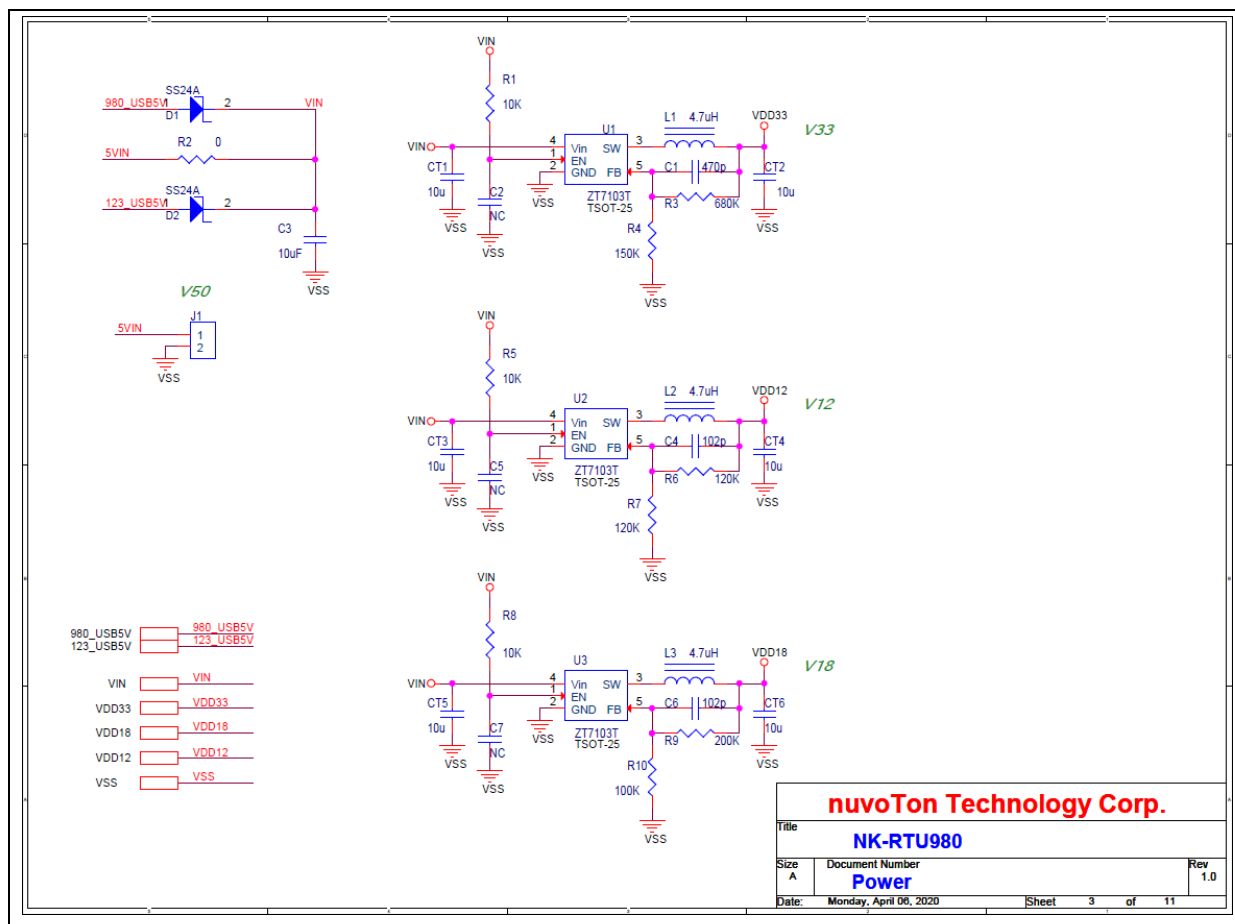
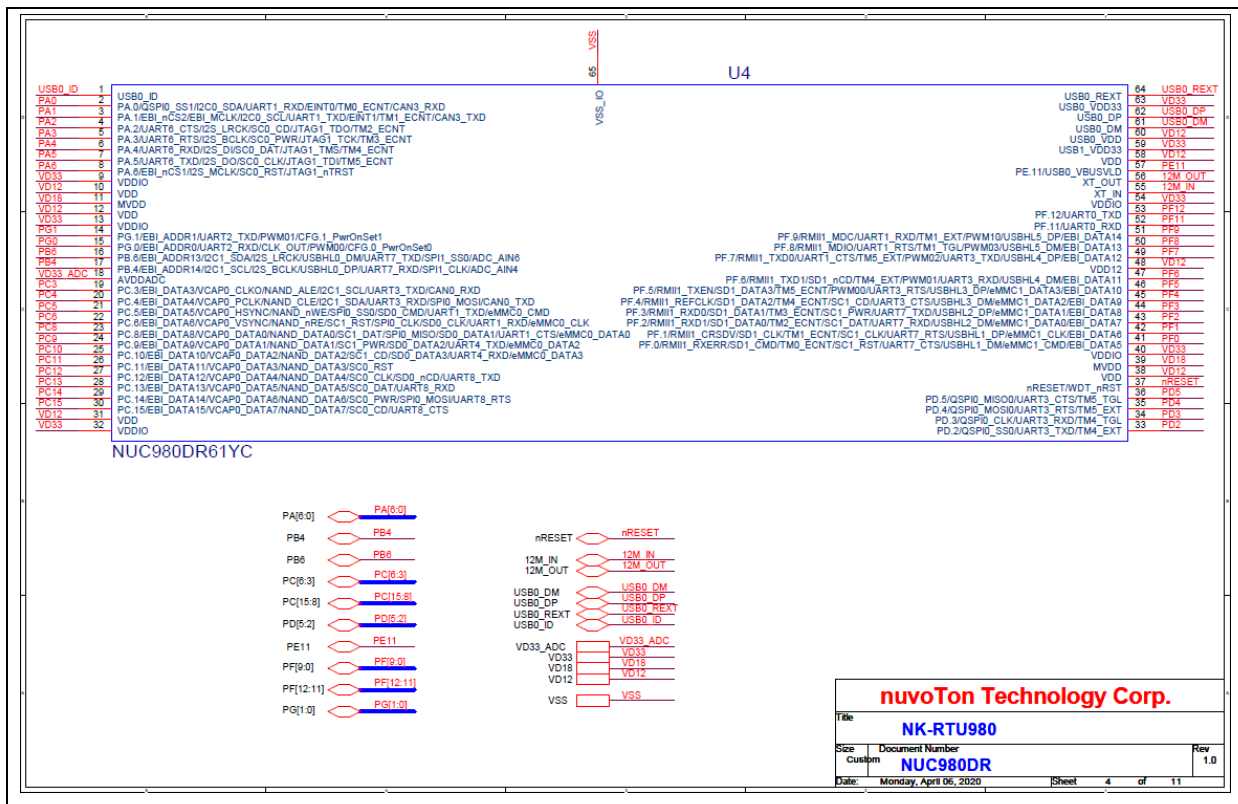


Figure 5-2 Power Schematic

5.3 NUC980DR Schematic



5.4 Power Filter Schematic

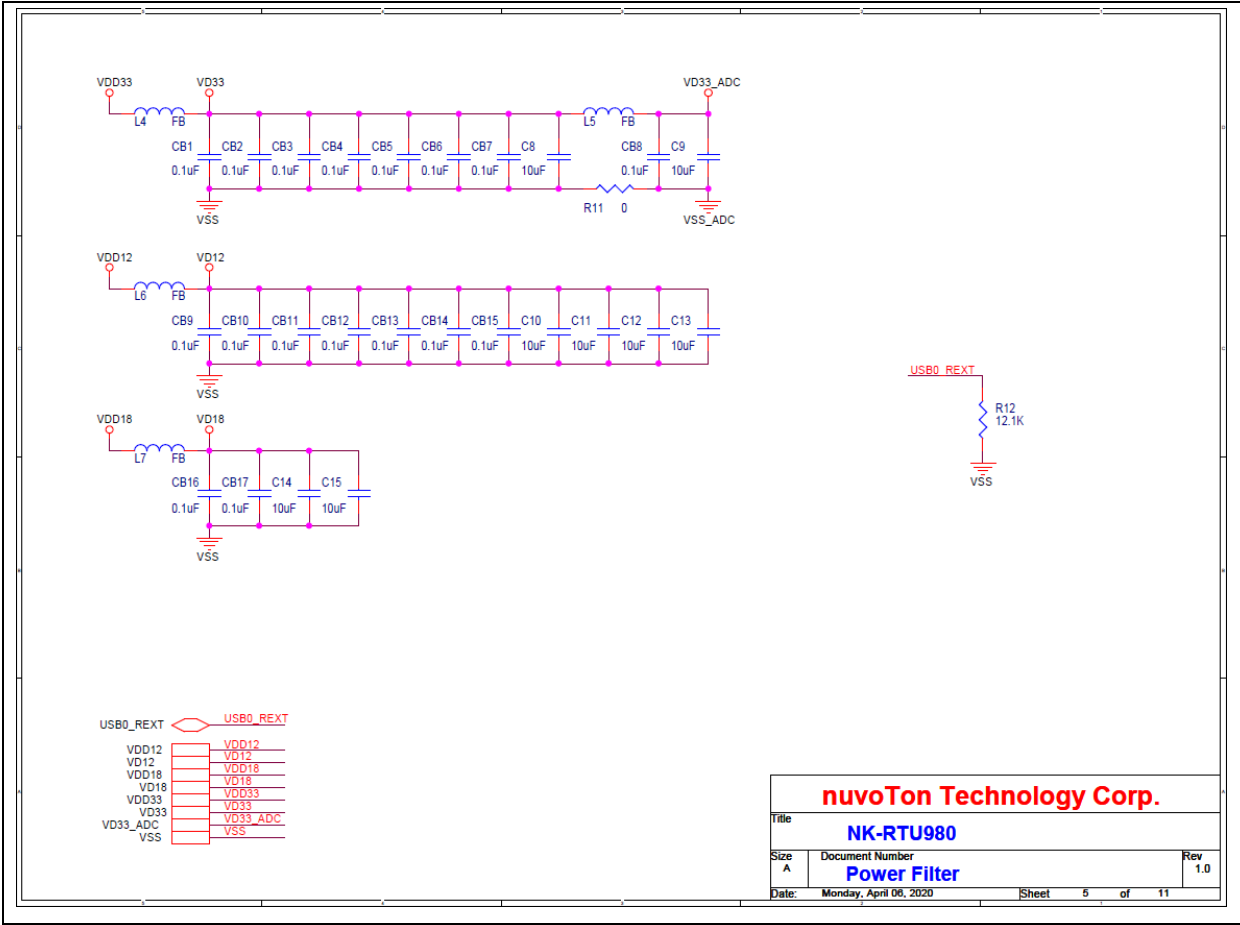


Figure 5-4 Power Filter Schematic

5.5 Configure Schematic

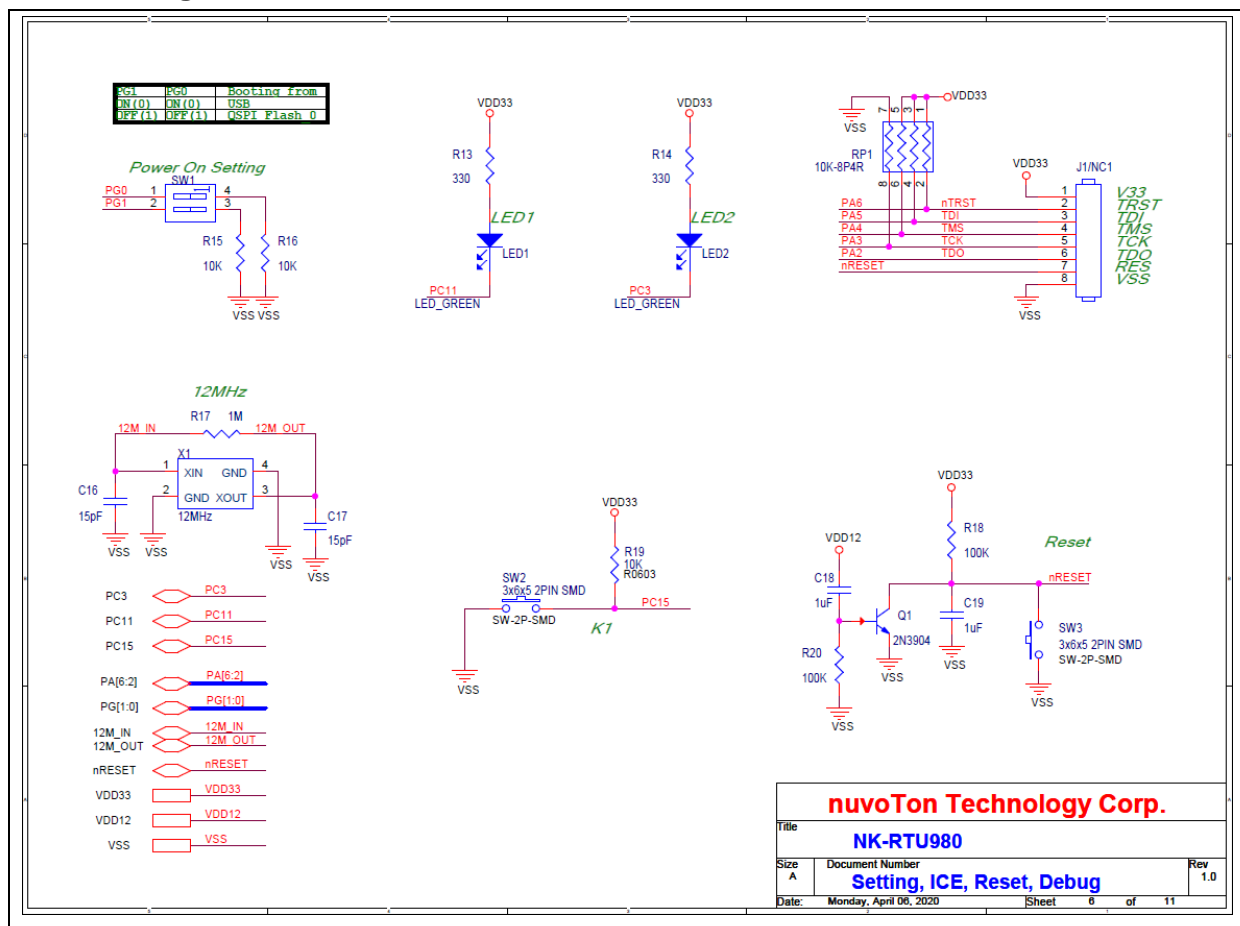


Figure 5-5 Configure Schematic

5.6 NUC123ZD4AN0 Schematic

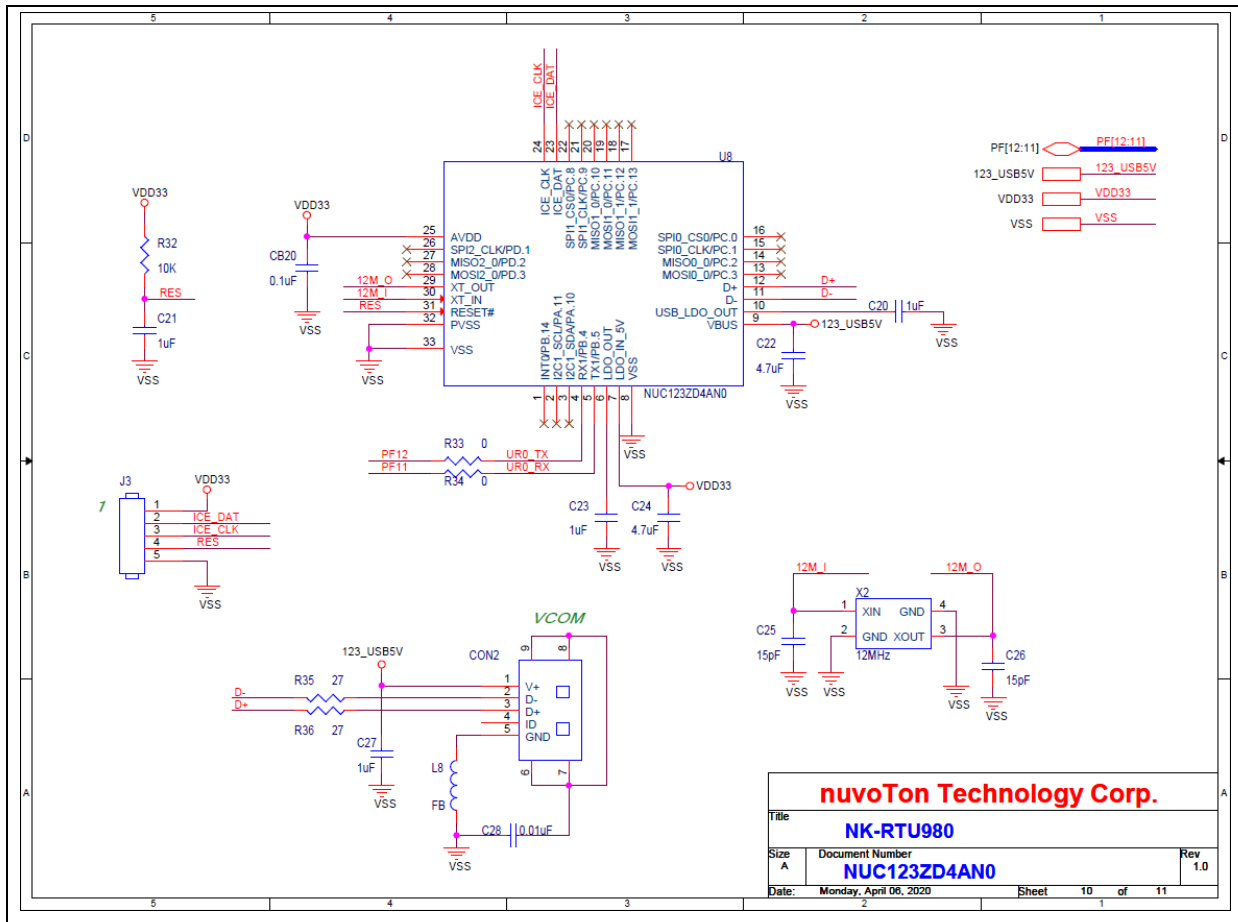


Figure 5-6 NUC123ZD4AN0 Schematic

5.7 Memory Schematic

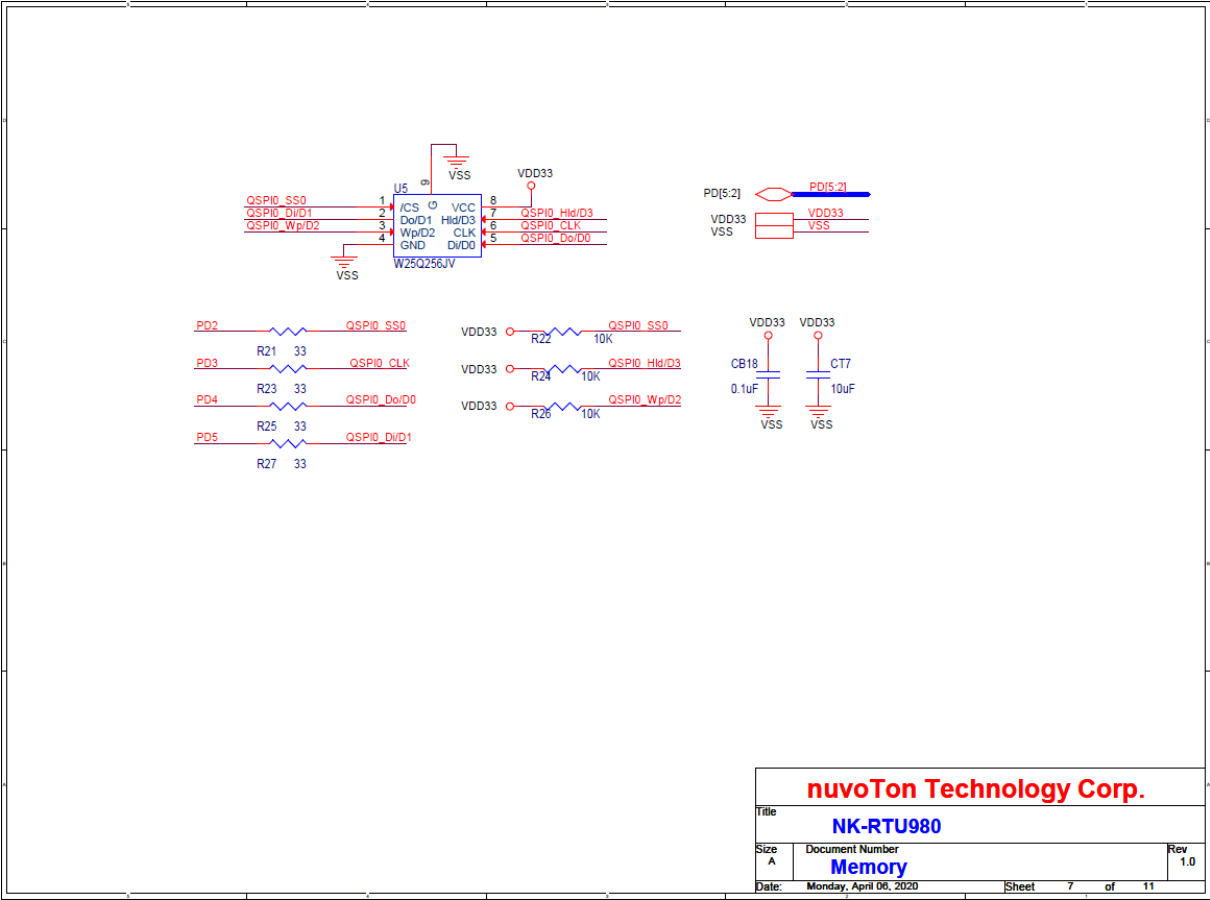


Figure 5-7 Memory Schematic

5.8 RMII_PF connector Schematic

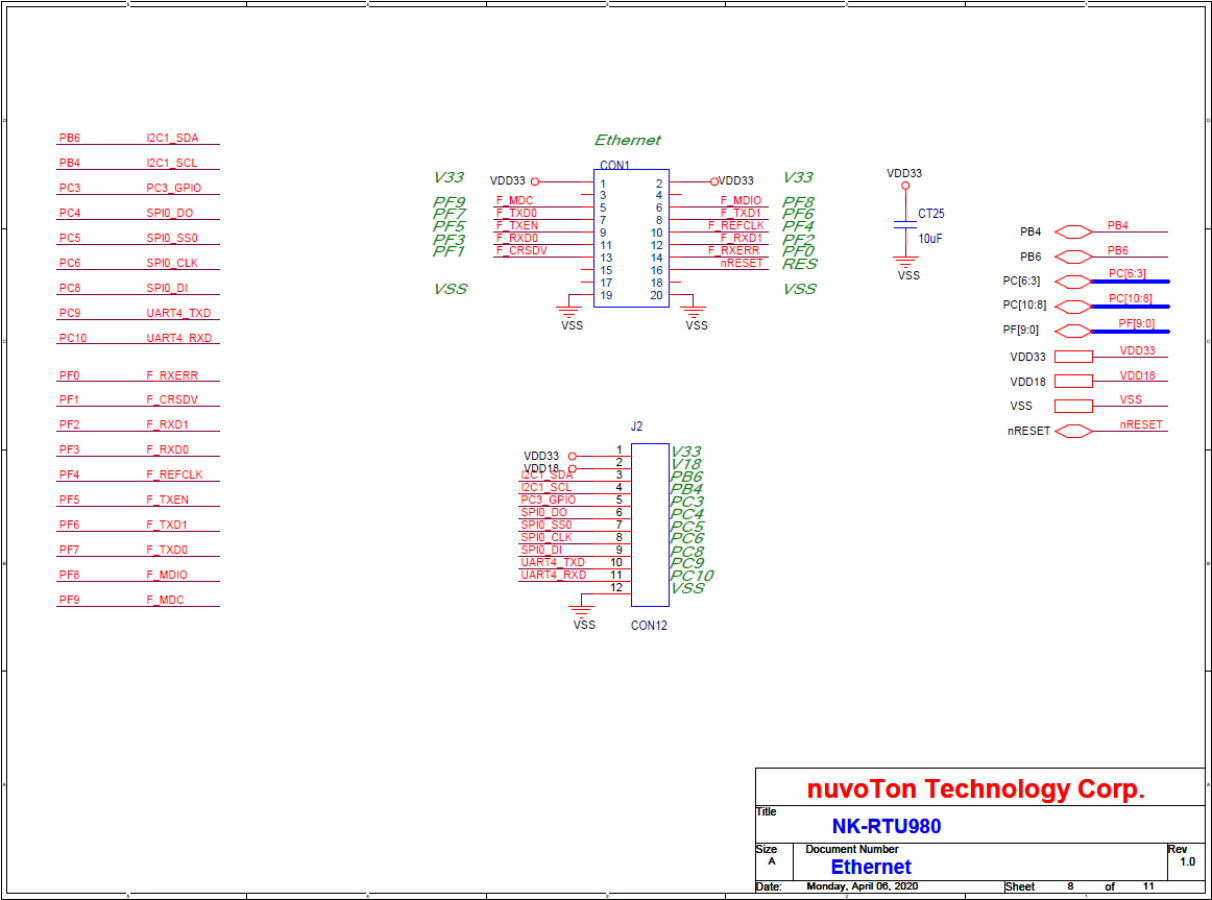


Figure 5-8 RMII_PF connector Schematic

5.9 RS485 and CAN Schematic

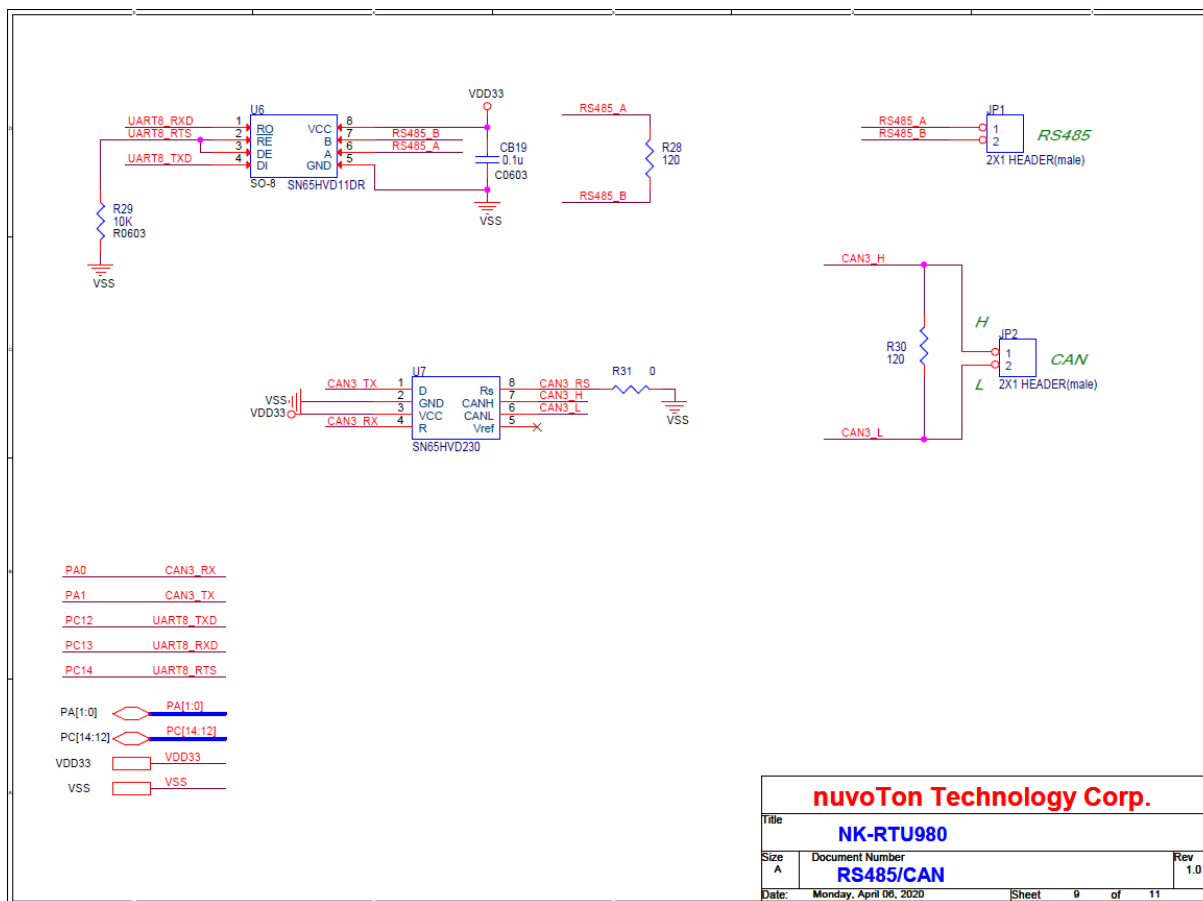


Figure 5-9 RS485 and CAN Schematic

5.10 USB Schematic

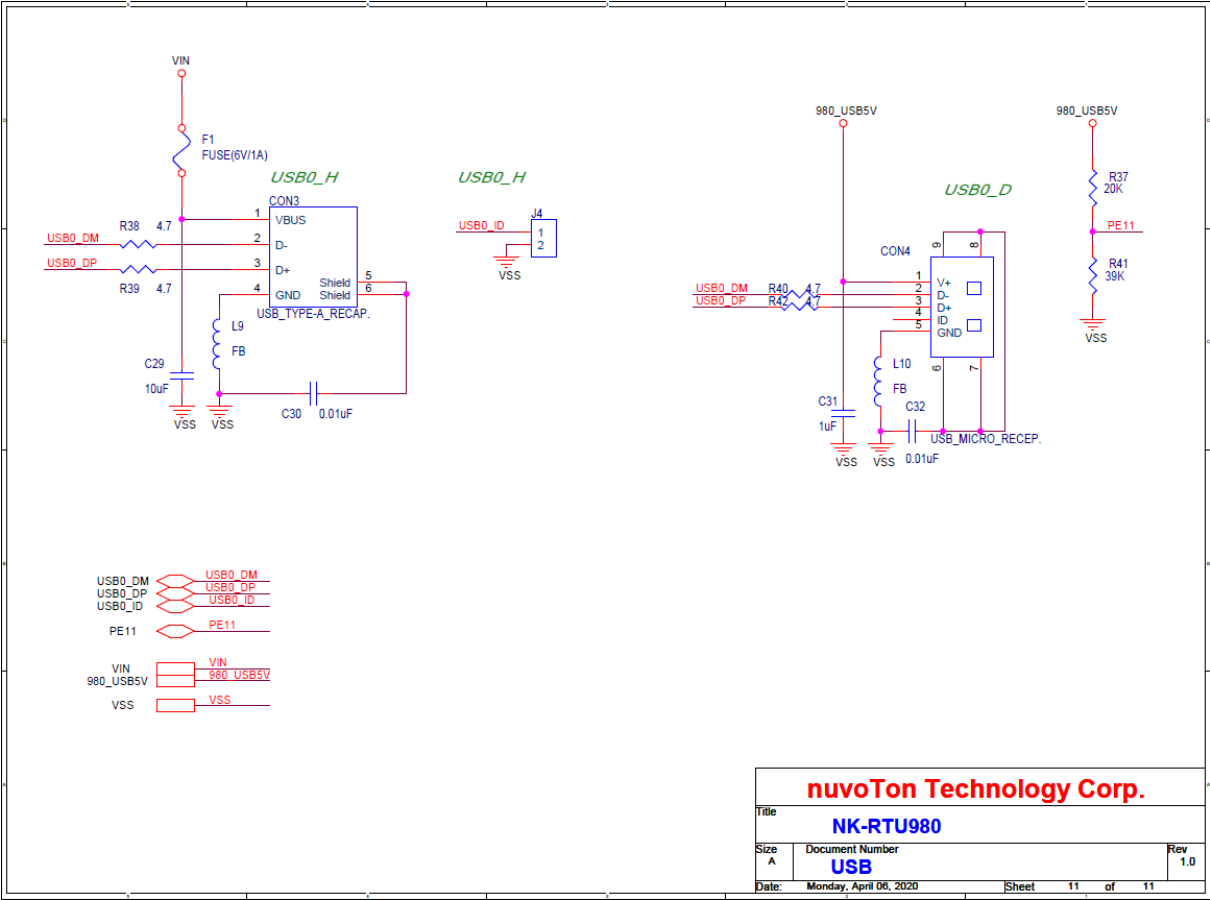


Figure 5-10 USB Schematic

5.11 PCB Placement

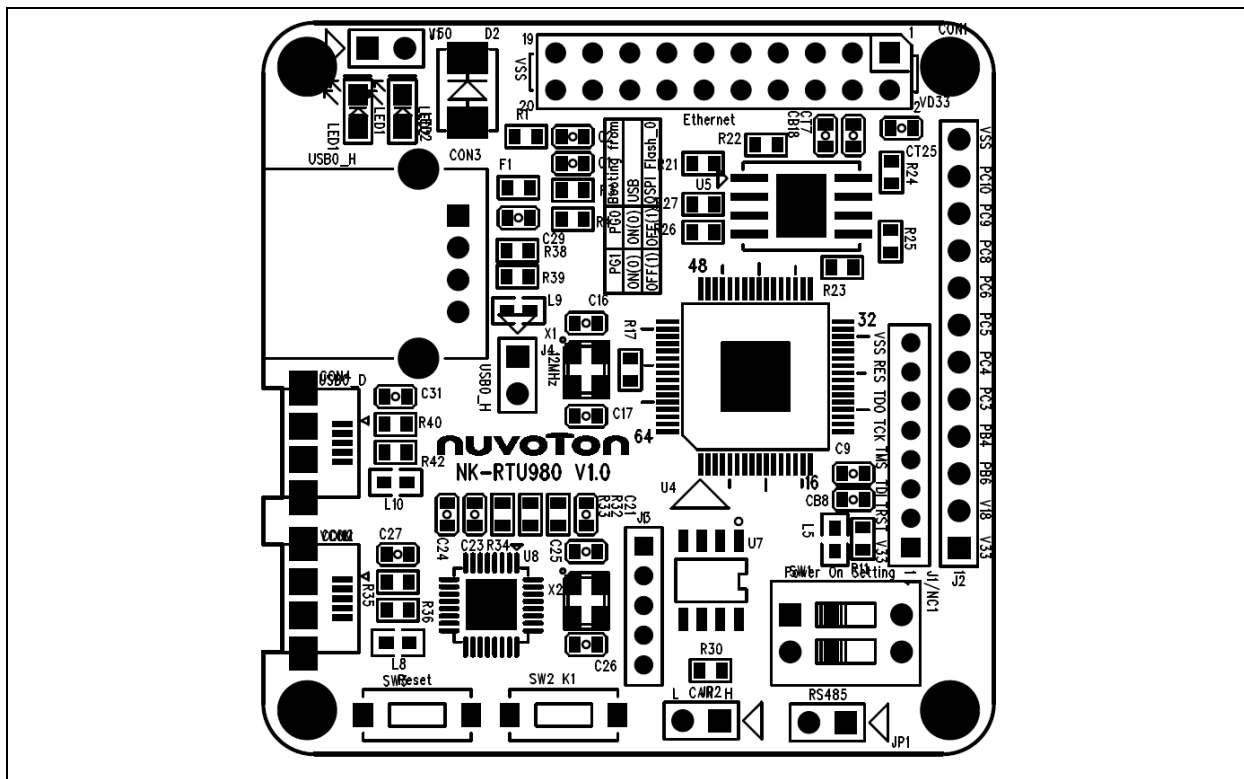


Figure 5-11 Front PCB Placement

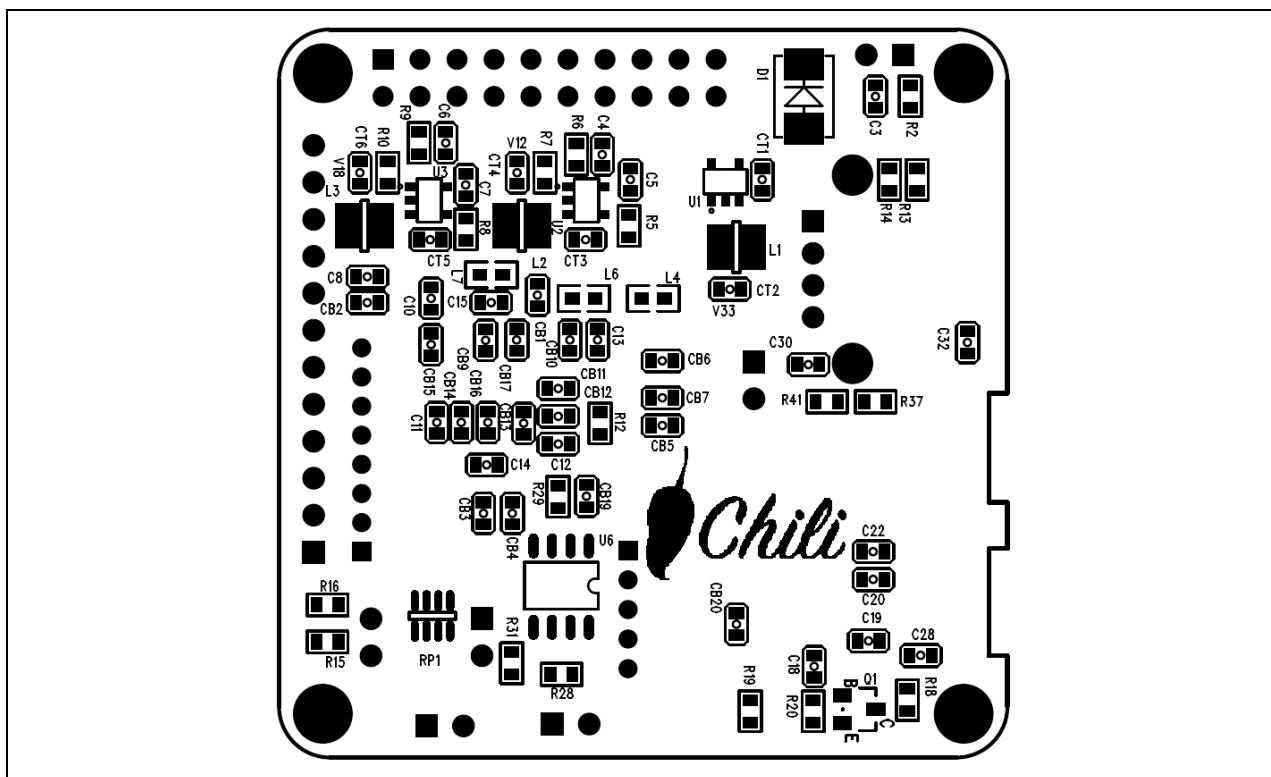


Figure 5-12 Back PCB Placement

6 REVISION HISTORY

Date	Revision	Description
2020.05.22	1.00	1. Initial version

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.*