NuMicro® Family

ARM926EJ-S™-based 32-bit Microprocessor

NuMaker-IIoT-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

List of Figures

List of Tables

1 OVERVIEW

The NuMaker-IIoT-NUC980 is an evaluation board based on Arm ARM926EJ-S microprocessor NUC980DK61Y which has rich peripherals to help users to design-in their products or application systems easily. In addition, the NuMaker-IIoT-NUC980 uses NUC980DK61Y microprocessor which runs up to 300 MHz with built-in 64 MB 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, SPI NAND Flash and SD/EMMC. All functions of the NUC980DK61Y are placed on the board, including peripheral interfaces such as memory (SPI NAND Flash, eMMC, SD), UART, Audio controller (NAU8822L), 10/100 Mbps Ethernet MAC controller, high speed USB (device, HOST), JTAG and EBI. Furthermore, the board provides Arduino Uno compatible interface for expansion. You can use it to develop and verify applications to emulate the real behavior.

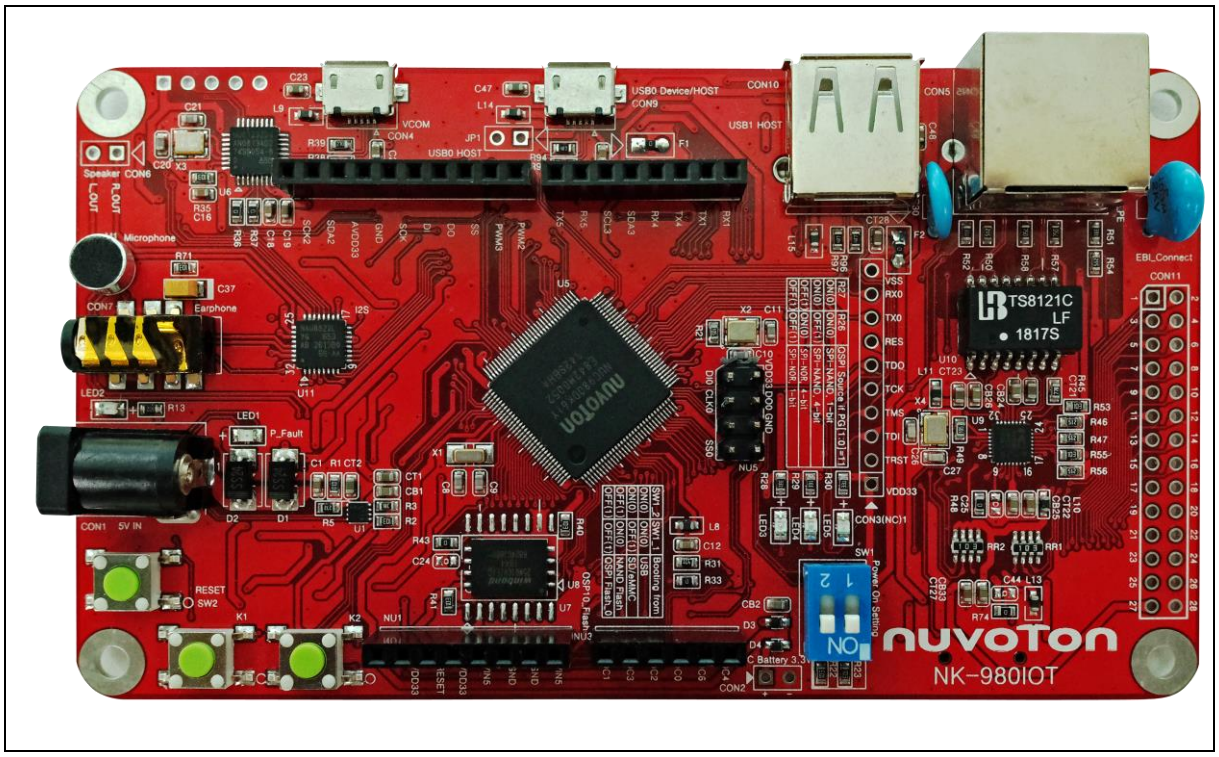


Figure 1-1 NuMaker-IIoT-NUC980 Evaluation Board

2 FEATURES

- NUC980DK61Y: LQFP128 pin MCP package with DDR2 (64 MB), which can run up to 300 MHz operating speed
- SPI Flash: Quad mode system booting or data storage, using W25N01GVZEIG SPI-NAND (128 MB)
- SD1/eMMC1: User SD/eMMC memory card for system booting, data storage or SDIO (Wi-Fi) device
- UART0: Connected to Virtual COM port for system development, debug message output
- Arduino Uno compatible interface connectors (NU1, NU2, NU3, NU4 and NU5)
- JTAG interface provided for software development
- RJ45 port with Ethernet 10/100 Mbps MAC (Ethernet0)
- EBI interface with pin header
- Microphone input and Earphone/Speaker output with 24-bit stereo Audio Codec (NAU8822L) for I^2S interface
- 3 sets of LED for status indication
- 2 sets of user-configurable push buttons
- 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

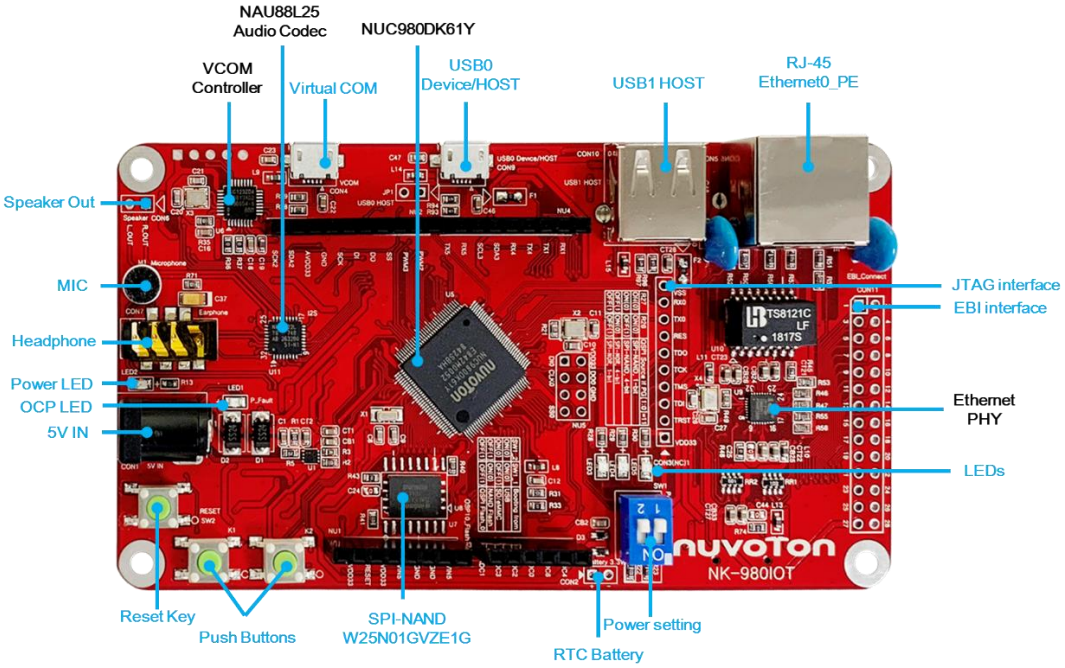


Figure 3-1 Front View of NuMaker-IIoT-NUC980

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

Power Model	CON4 USB Port (Micro-B)	CON9 USB Port (Micro-B)	CON1
Model 1	Connect to PC	-	-
Model 2	-	Connect to PC	-
Model 3	-	-	VDD5V Input

- Power indication LEDs (LED1, LED2):

LED	Color	Descriptions
LED1	Red	The system power will be terminated and LED1 lighting when the input voltage is over 5.7V or the current is over 1.7A.
LED2	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 (CON4, U6): NUC123ZD4AN0 microcontroller (U6), USB micro-B connector (CON4) to PC, for debug message output
- User indication LEDs (LED3, LED4, LED5):

LED	Color	GPIO pin of NUC980
LED3	Yellow	PB8
LED4	Green	PG15
LED5	Red	PB13

- SPI NAND Flash (U7, U8): Use Winbond W25N01GVZEIG 128MB (U8) for system booting; only one (U7 or U8) SPI Flash can be used, supporting dual / quad mode
- 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

- User Key SWs (K1 and K2)

Key	GPIO pin of NUC980
K1	GPE10
K2	GPE12

- Arduino UNO compatible interface (NU1, NU2, NU3, NU4 and NU5)

Connector	GPIO pin of NUN980	Function
NU1.1	-	-
NU1.2	-	VDD33
NU1.3	-	nRESET
NU1.4	-	VDD33
NU1.5	-	VIN
NU1.6	-	VSS

NU1.7	-	VSS
NU1.8	-	VIN

Connector	GPIO pin of NUN980	Function
NU2.1	GPF7	PWM2
NU2.2	GPF8	PWM3
NU2.3	GPG11	SPI1_SS
NU2.4	GPG14	SPI1_DO
NU2.5	GPG13	SPI1_DI
NU2.6	GPG12	SPI1_CLK
NU2.7	-	VSS
NU2.8	-	ADC VDD33
NU2.9	GPB7	I2C2_SDA
NU2.10	GPB5	I2C2_SCL

Connector	GPIO pin of NUN980	Function
NU3.1	GPB1	UART9_TXD
NU3.2	GPB3	UART9_RXD
NU3.3	GPB2	ADC_AIN[2]
NU3.4	GPB0	ADC_AIN[0]
NU3.5	GPB6	UART7_TXD
NU3.6	GPB4	UART7_RXD

Connector	GPIO pin of NUN980	Function
NU4.1	GPF9	UART1_RXD
NU4.2	GPF10	UART1_TXD
NU4.3	GPD12	UART4_TXD
NU4.4	GPD13	UART4_RXD
NU4.5	GPD15	I2C3_SDA
NU4.6	GPD14	I2C3_SCL
NU4.7	GPG6	UART5_RXD
NU4.8	GPG7	UART5_TXD

Connector	GPIO pin of NUN980	Function
NU5.1	GPD11	SPI0_DI
NU5.2	-	VDD33
NU5.3	GPD9	SPI0_CLK
NU5.4	GPD10	SPI0_DO
NU5.5	-	-
NU5.6	-	VSS
NU5.7	GPD8	SPI0_SS
NU5.8	-	-

- EBI port for use (CON11)

Connector	GPIO pin of NUN980	Function
CON11.1	GPC0	EBI_DATA0
CON11.2	GPC1	EBI_DATA1
CON11.3	GPC2	EBI_DATA2
CON11.4	GPC3	EBI_DATA3
CON11.5	GPC4	EBI_DATA4
CON11.6	GPC5	EBI_DATA5
CON11.7	GPC6	EBI_DATA6
CON11.8	GPC7	EBI_DATA7
CON11.9	GPC8	EBI_DATA8
CON11.10	GPC9	EBI_DATA9
CON11.11	GPC10	EBI_DATA10
CON11.12	GPC11	EBI_DATA11
CON11.13	GPC12	EBI_DATA12
CON11.14	GPC13	EBI_DATA13
CON11.15	GPC14	EBI_DATA14
CON11.16	GPC15	EBI_DATA15
CON11.17	GPA7	EBI_nWE
CON11.18	GPA8	EBI_nRE
CON11.19	GPA9	EBI_nCS0
CON11.20	GPA12	EBI_ADDR8

CON11.21	GPA11	EBI_ADDR9
CON11.22	GPA10	EBI_ADDR10
CON11.23	GPB0	ADC_AIN[0]
CON11.24	GPB2	ADC_AIN[2]
CON11.25	GPB4	ADC_AIN[4]
CON11.26	GPB6	ADC_AIN[6]
CON11.27	-	VDD33
CON11.28	-	VSS

- SD1/eMMC1 (CON8): Use Micro SD/eMMC memory card for system booting, data storage or SDIO (Wi-Fi) device
- Power on setting (SW1, R24~R27)

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/ ON	Boot from NAND Flash	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

Resistance	Status	Function	GPIO pin of NUC980
R27/R26	Solder R/ Solder R	SPI-NAND Flash boot with 1-bit mode	GPG9/GPG8
R27/R26	Solder R/ Remove	SPI-NAND Flash boot with 4-bit mode	GPG9/GPG8
R27/R26	Remove/ Solder R	SPI-NOR Flash boot with 4-bit mode	GPG9/GPG8

R27/R26	Remove/ Remove	SPI-NOR Flash boot with 1-bit mode	GPG9/GPG8

- Audio Codec (U11, M1, CON6, CON7, CN1): Nuvoton NAU8822L (U11) connects to NUC980 using I²S interface
  - Microphone (M1): Through the NAU8822L chip sound input
  - Speaker output (CON6): Through the NAU8822L chip sound output

Connector	Pin Name	Functions
CON6.1	SPKOUT_R	NAU8822L BTL Speaker Positive Output or Right high current output.
CON6.2	SPKOUT_L	NAU8822L BTL Speaker Negative Output or Left high current output.

  - Earphone output (CON7): Through the NAU8822L chip sound output

- USB0 Device/HOST (CON9, JP1): USB0 Device/HOST Micro-B connector, By JP1 status or defined by the ID pin of the USB cable
- USB1 HOST (CON10): USB1 for USB HOST with type-A connector
- Ethernet0_PE (CON5, U9): For Ethernet port, the NUC980 support RMII interface which add one Ethernet PHY IP101GR to RJ45 connector with LED indicator
- SOC CPU: NUC980DK61Y (U5)

3.2 Rear View

Figure 3-2 shows the main components and connectors from the rear side of NuMaker-IIoT-NUC980.

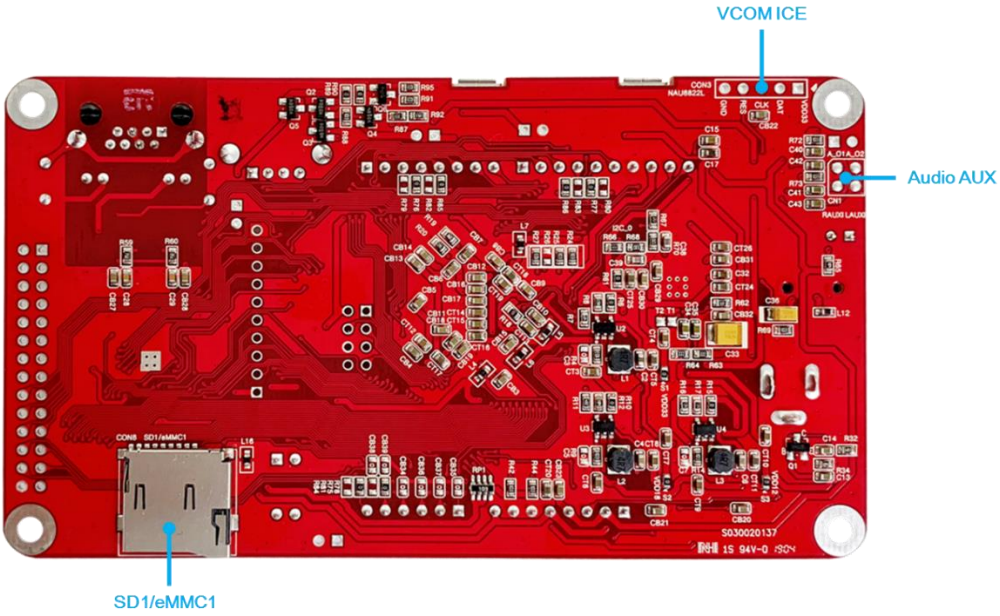


Figure 3-2 Rear View of NuMaker-IIoT-NUC980

- 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

- Audio Codec (U11, M1, CON6, CON7, CN1): Nuvoton NAU8822L (U11) connects to NUC980 using I²S interface
  - Auxiliary Input and Output(CN1)

Connector	Pin Name	Functions
CN1.1	AUXOUT1	Mono Mixed Output / Line Output
CN1.2	AUXOUT2	Line Output
CN1.3	AUXINR	Right Auxiliary Input
CN1.4	AUXINL	Left Auxiliary Input

- MicroSD Card Slot: T-Flash slot (CON8)

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 4-2.

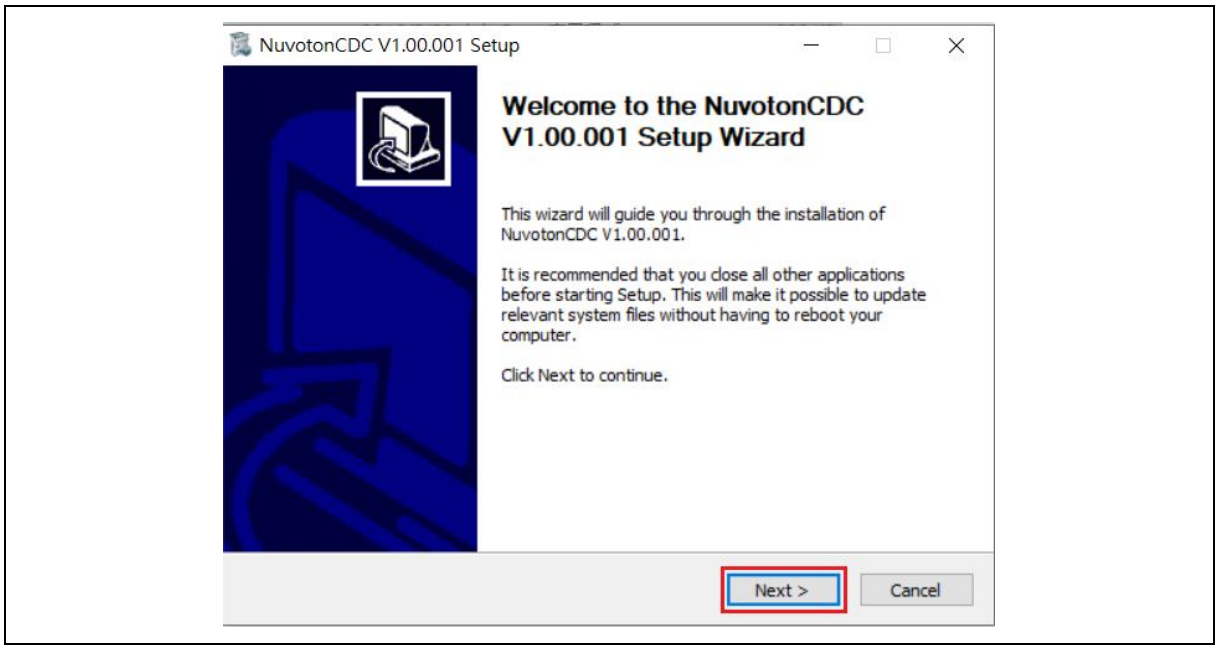
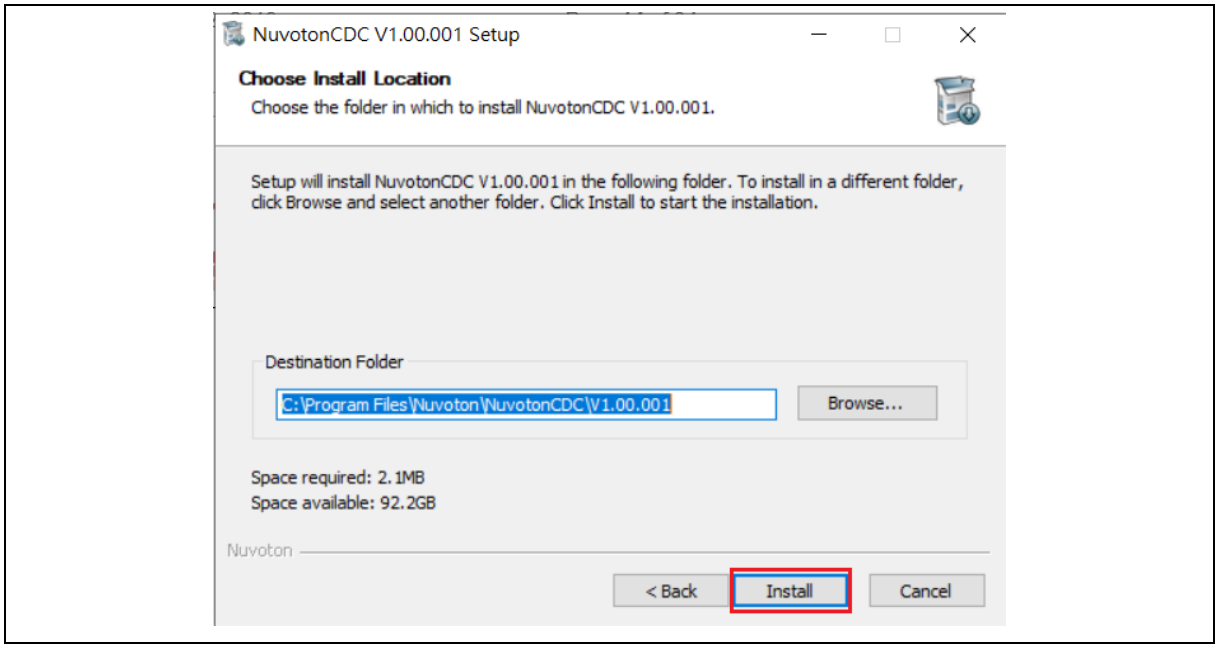


Figure 4-1 CDC Driver Installation Setup



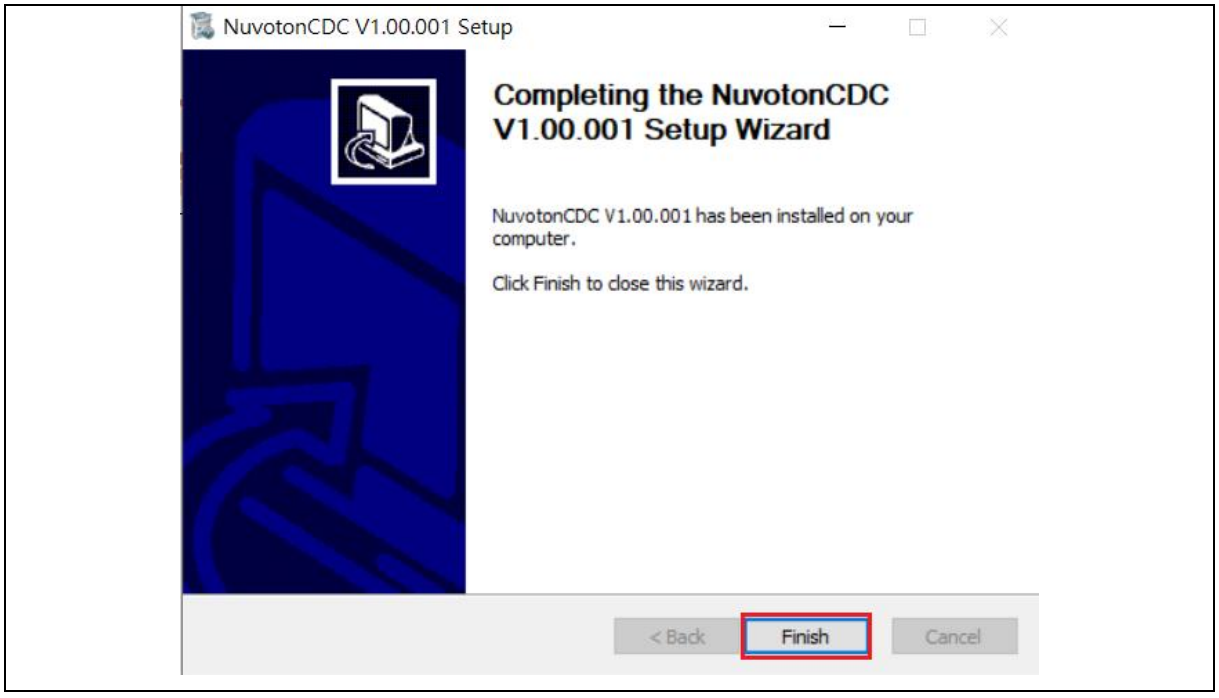


Figure 4-2 CDC Driver Installation

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

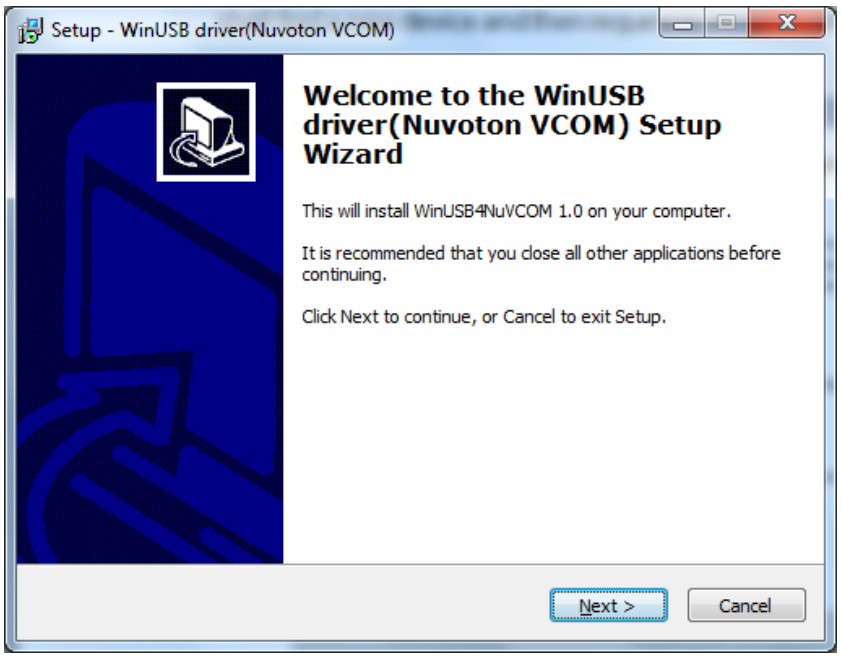
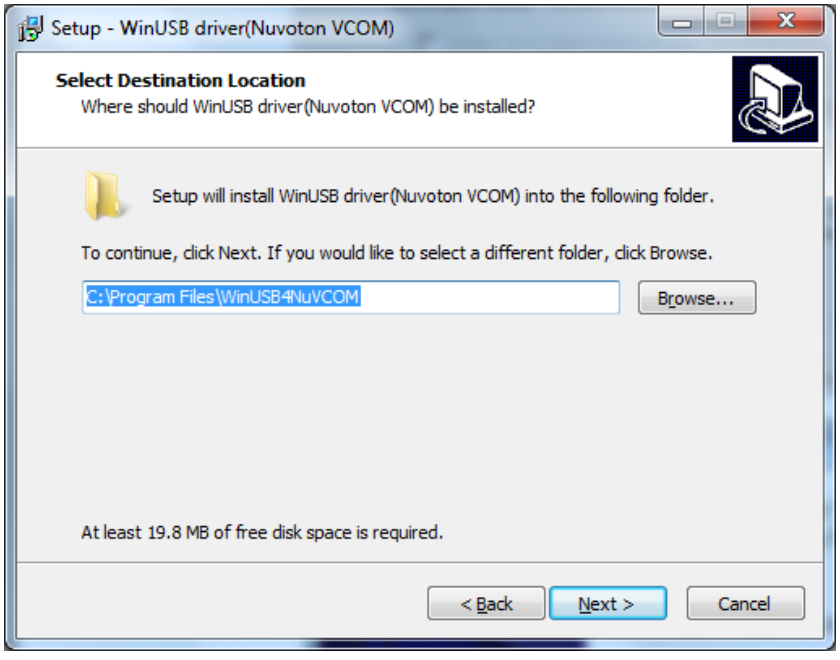
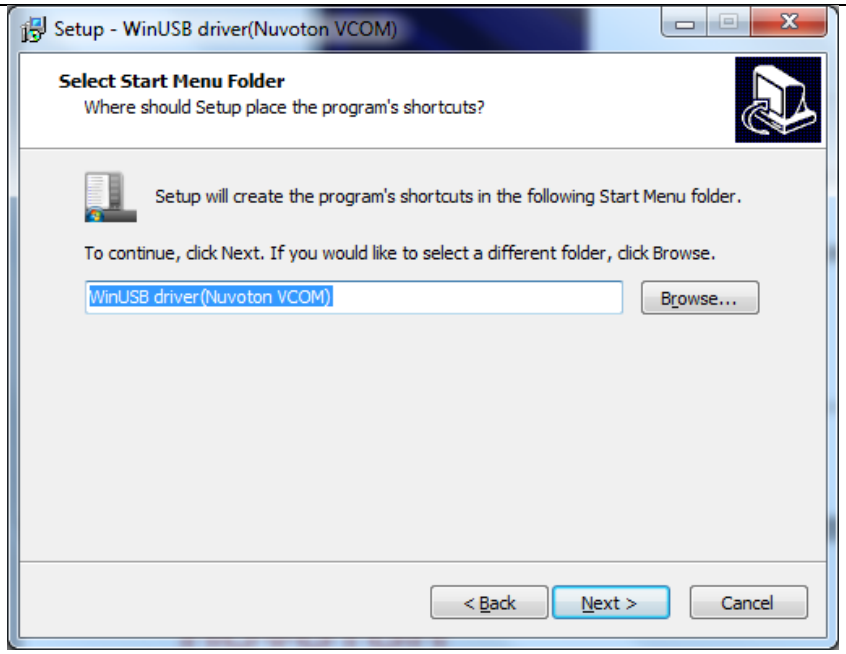
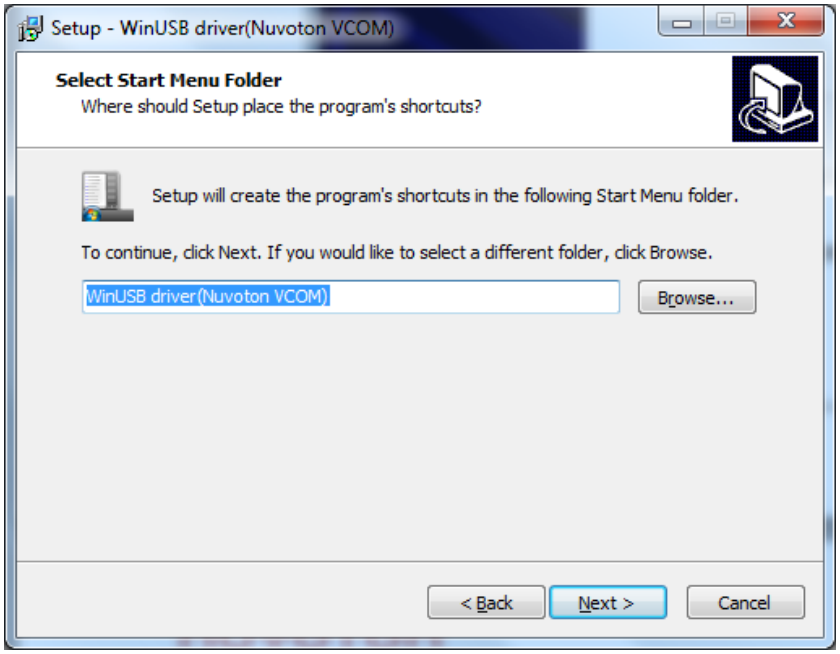
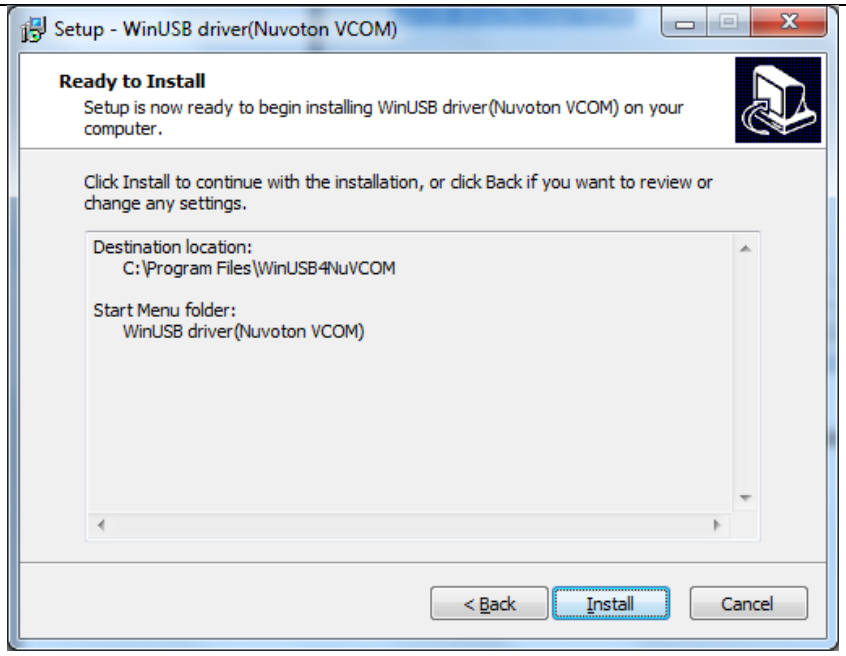


Figure 4-3 VCOM Driver Installation Setup




Setup - WinUSB driver(Nuvoton VCOM)
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
WinUSB driver(Nuvoton VCOM)
Browse...
< Back
Next >
Cancel


Setup - WinUSB driver(Nuvoton VCOM)
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
WinUSB driver(Nuvoton VCOM)
Browse...
< Back
Next >
Cancel



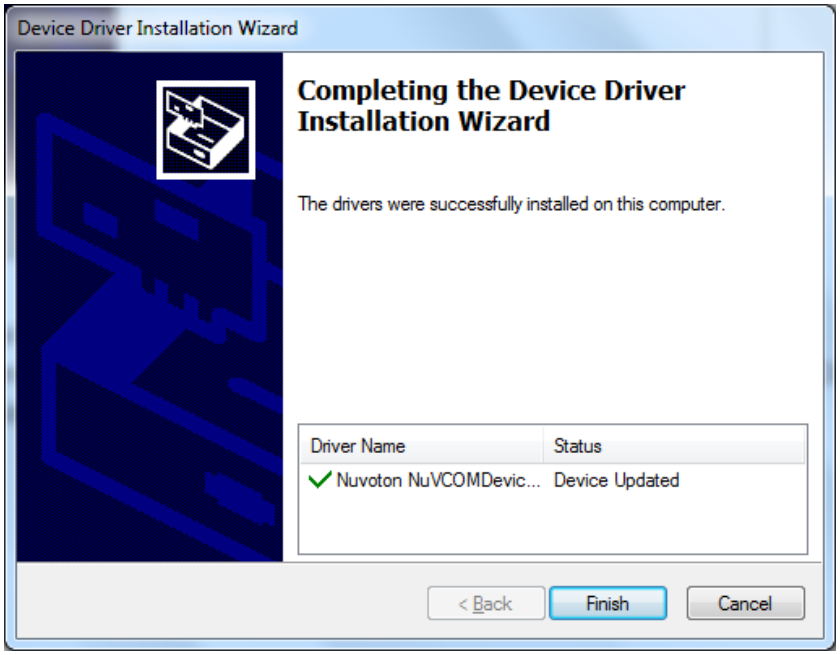


Figure 4-4 VCOM Driver Installation

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-iiot-nuc980/index.html

- VMware Linux Virtual machine image
  - An UBUNTU16.04 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 V4.4 BSP
  - Linux BSP, NuWriter tool and documents

Github:

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

For more details about NUC980 Linux BSP, please refer to *NUC980 Linux 4.4 BSP User Manual* in the "*BSP/Documents*" directory.

4.4 Hardware Setup

The NuMaker-IIoT-NUC980 provides jumpers to select boot-up conditions. To select USB ISP mode, the statuses of SW1.1 and SW1.2 are ON. For other boot selections, refer to Figure 4-5 and Table 4-1.

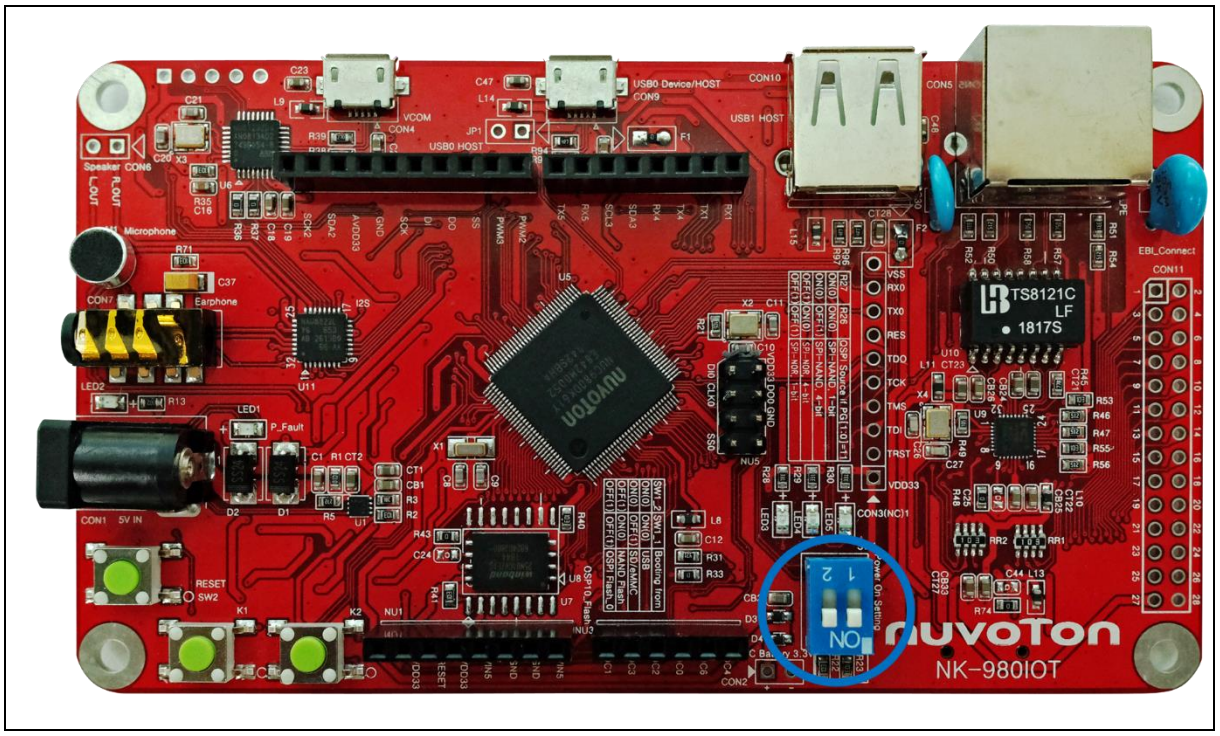


Figure 4-5 Boot Source Selection

Power-on setting	SW1.2	SW1.1
USB ISP	ON	ON
Boot from eMMC/SD	ON	OFF
Boot from NAND	OFF	ON
Boot from SPI	OFF	OFF

Table 4-1 Boot Source Selection Table

Refer to the following steps to use the debug console:

1. Connect USB-Serial connector shown in Figure 4-6 to the PC USB port through a USB cable.

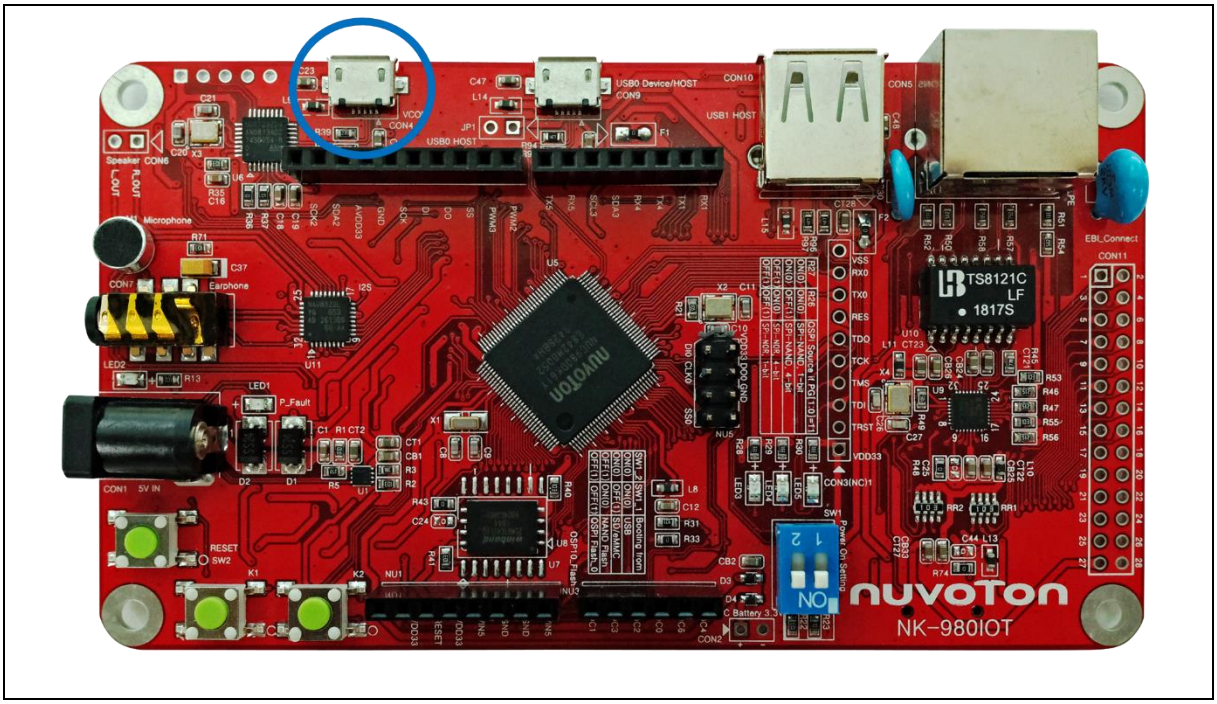


Figure 4-6 USB-Serial Debug Port

2. Find the “Nuvoton Virtual COM Port” on the Device Manger as Figure 4-7.

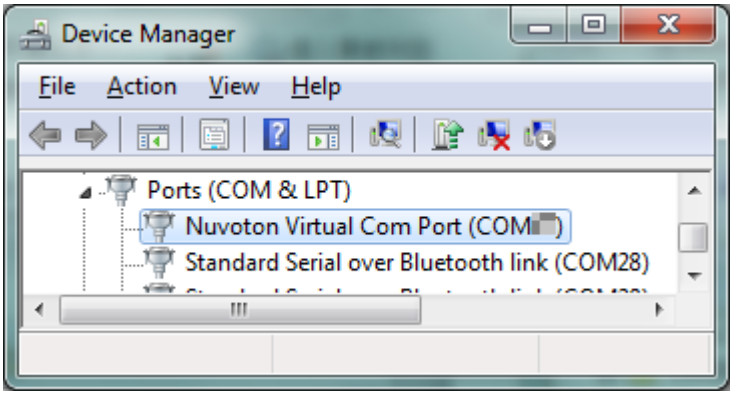


Figure 4-7 Device Manger

3. Open a serial port terminal, PuTTY for example, to print out debug message. Set the speed to 115200.

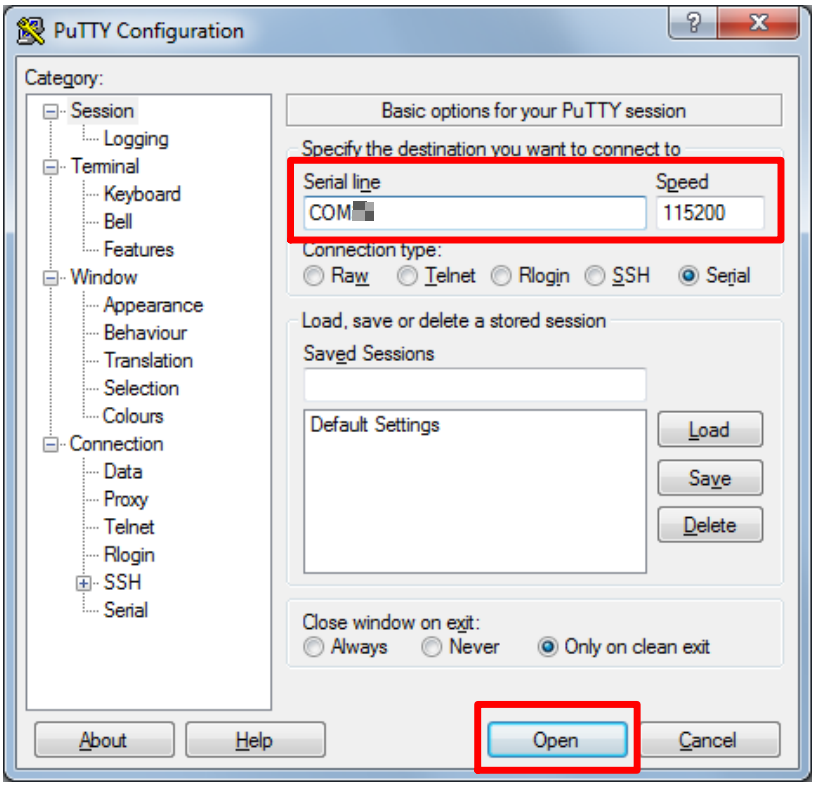


Figure 4-8 PuTTY Session Setting

Figure 4-9 is the log after booting from SPI NAND.

```
NUC980 IBR 20180813
Boot from SPI-NAND
DDR-OK
finish SPI download

 SPL load main U-Boot from SPI NAND Flash! (Feb 20 2019 09:12:29)


U-Boot 2016.11-g8127c47 (Feb 20 2019 - 09:12:26 +0800)

CPU: NUC980
Board: NUC980
DRAM:  64 MiB
SF: Detected W25N01GV with page size 2 KiB, erase size 128 KiB, total 128 MiB
In:    serial
Out:   serial
Err:   serial
Net:   Net Initialization Skipped
No ethernet found.
Hit any key to stop autoboot:  0
SF: Detected W25N01GV with page size 2 KiB, erase size 128 KiB, total 128 MiB
device 0 offset 0x200000, size 0x800000
SF: 8388608 bytes @ 0x200000 Read: OK
## Booting kernel from Legacy Image at 00007fc0 ...
   Image Name:   Linux-4.4.115+
   Image Type:   ARM Linux Kernel Image (uncompressed)
   Data Size:    6491128 Bytes = 6.2 MiB
   Load Address: 00008000
   Entry Point:  00008000
   Verifying Checksum ... OK
   XIP Kernel Image ... OK
```

Figure 4-9 Booting Log

4.5 NuWriter Tool

4.5.1 NuWriter Setup

1. Refer to section 4.3 to install NuWriter tool.
2. Connect USBD connector to the PC USB port through a USB cable.

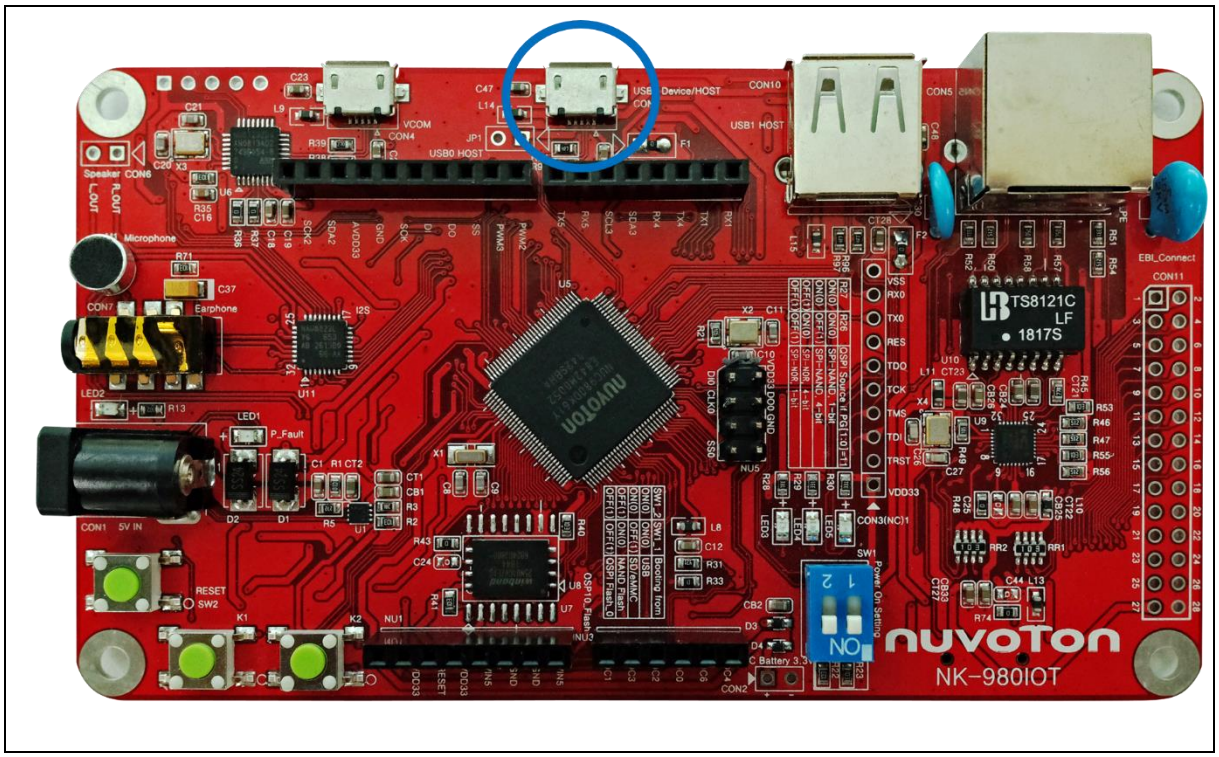


Figure 4-10 USBD Port

3. Boot NuMaker-IIoT-NUC980 from USB ISP mode.
4. Find the “WinUSB driver (Nuvoton VCOM)” on the Device Manger as Figure 4-11.

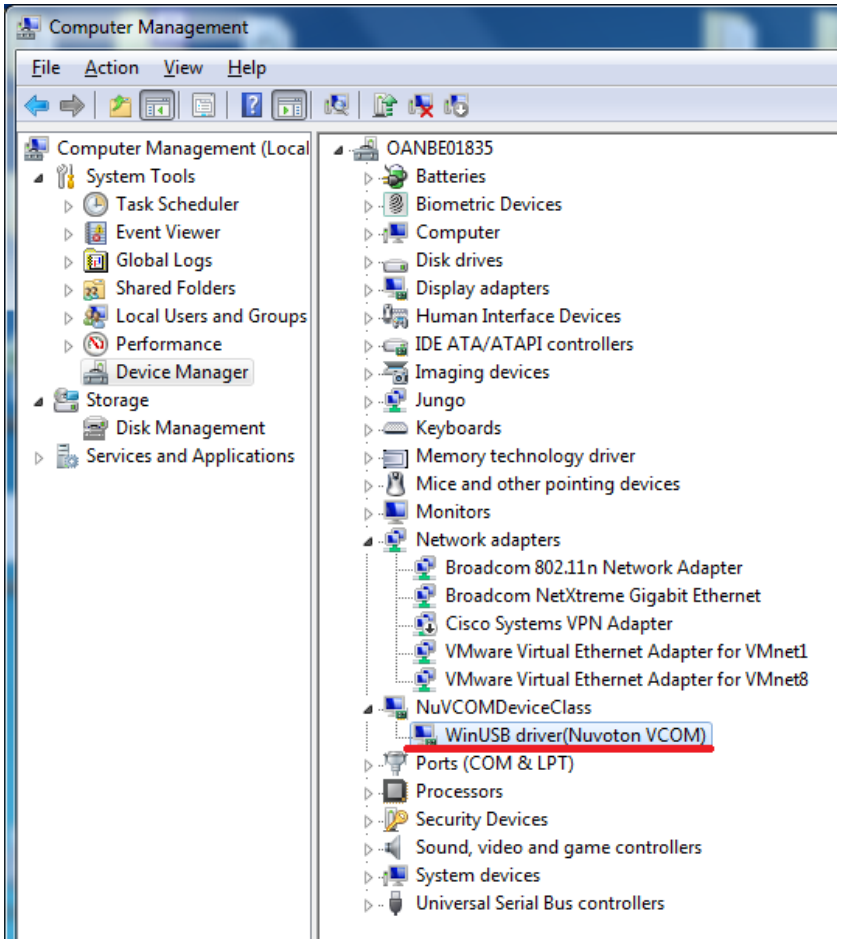


Figure 4-11 Device Manger(2)

Power on the NuMaker-IIoT-NUC980, and then open the programming tool "*NuWriter.exe*" on the PC. Note that the tool cannot work if the "**WinUSB4NuVCOM**" driver is not found.

First, double-click "*NuWriter.exe*" on PC. NuWriter will start and a window appears. Select target chip as NUC980 series and select DDR parameter as DDR initial files.

DDR initial is available to set register values. For example, the default GPIO output mode is push-pull. If you want to switch to open-drain mode, it is helpful to set register values via DDR initial files.

After selecting DDR parameter, click "**Continue**" to use NuWriter tool.

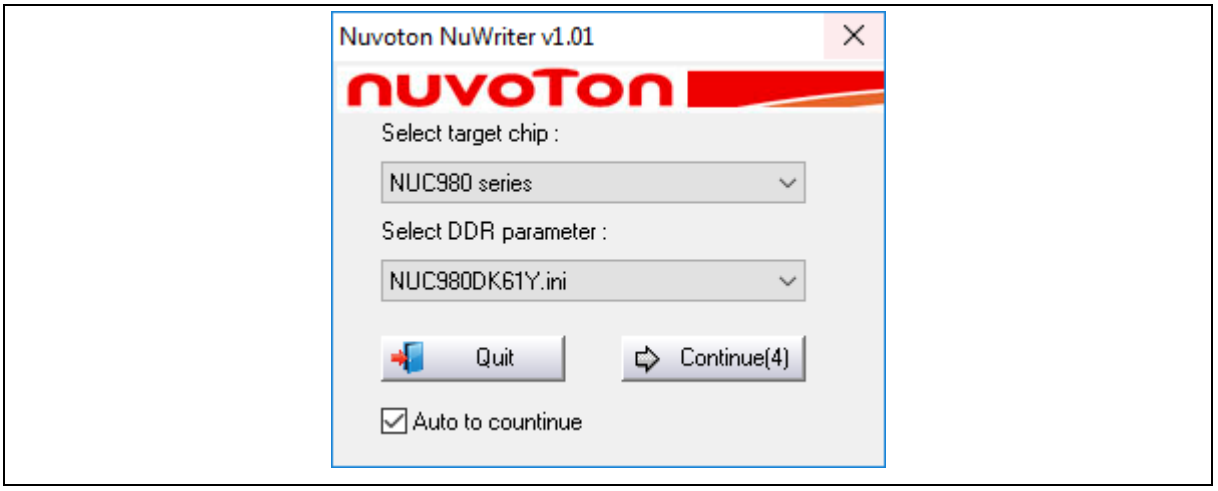


Figure 4-12 NuWriter Chip Setting

NuWriter provides 7 types of images to be downloaded including DDR/SRAM, SPI, NAND, eMMC/SD, SPI NAND, PACK and Mass Production. This chapter will guide you to download images to SPI NAND flash. If you want to choose others types to download images. For more details about NUC980 Linux BSP, please refer to *NUC980 NuWriter User Manual* in the "*BSP/Documents*" directory.

4.5.2 SPI NAND Mode

This mode can write a new image to SPI NAND flash and specify the type of the image. These types can be recognized by uboot or Linux. The Image type is set Loader, Data, Environment or Pack.

The NuMaker-IIoT-NUC980 default firmware consists of four images:

- u-boot-spl
- u-boot
- uImage
- environment variables

Please refer to *VMware Linux Virtual Machine Image User Manual* to generate these firmware images.

u-boot-spl

For the Linux system, Loader Type is used to boot the Linux kernel. To compile NUC980 U-Boot to get Main U-Boot and SPL U-Boot. The SPL U-Boot is a small binary file, which will move Main U-Boot into DDR execution. The SPL U-Boot is only for NAND/SPI NAND boot. The default link address of SPL U-Boot is 0x200. For the detailed introduction of Loader Type format, refer to *NUC980 NuWriter User Manual* in the "*BSP/Documents*" directory

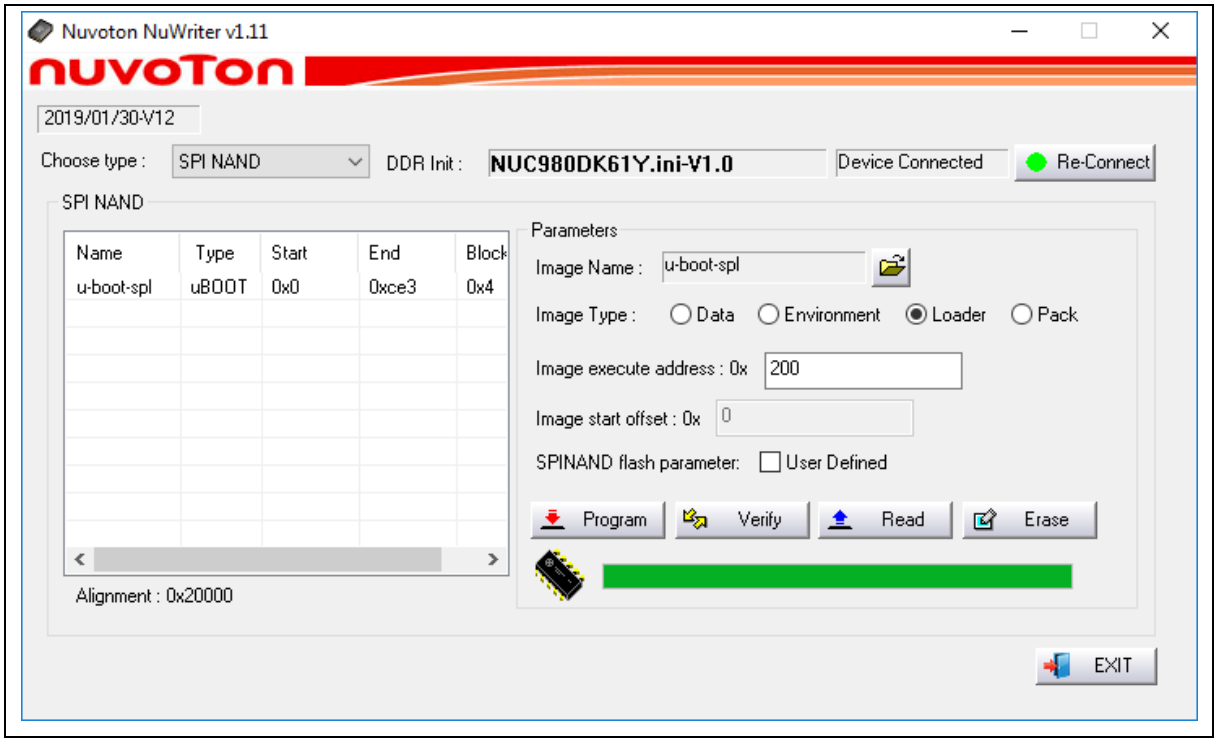


Figure 4-13 Download u-boot-spl to SPI NAND

u-boot

For the Linux system, Loader Type is used to boot the Linux kernel. Compile NUC980 U-Boot to get Main U-Boot and SPL U-Boot. The Main U-Boot is a fully featured version of U-Boot. In this case, the Main U-Boot need to set the address at 0x100000.

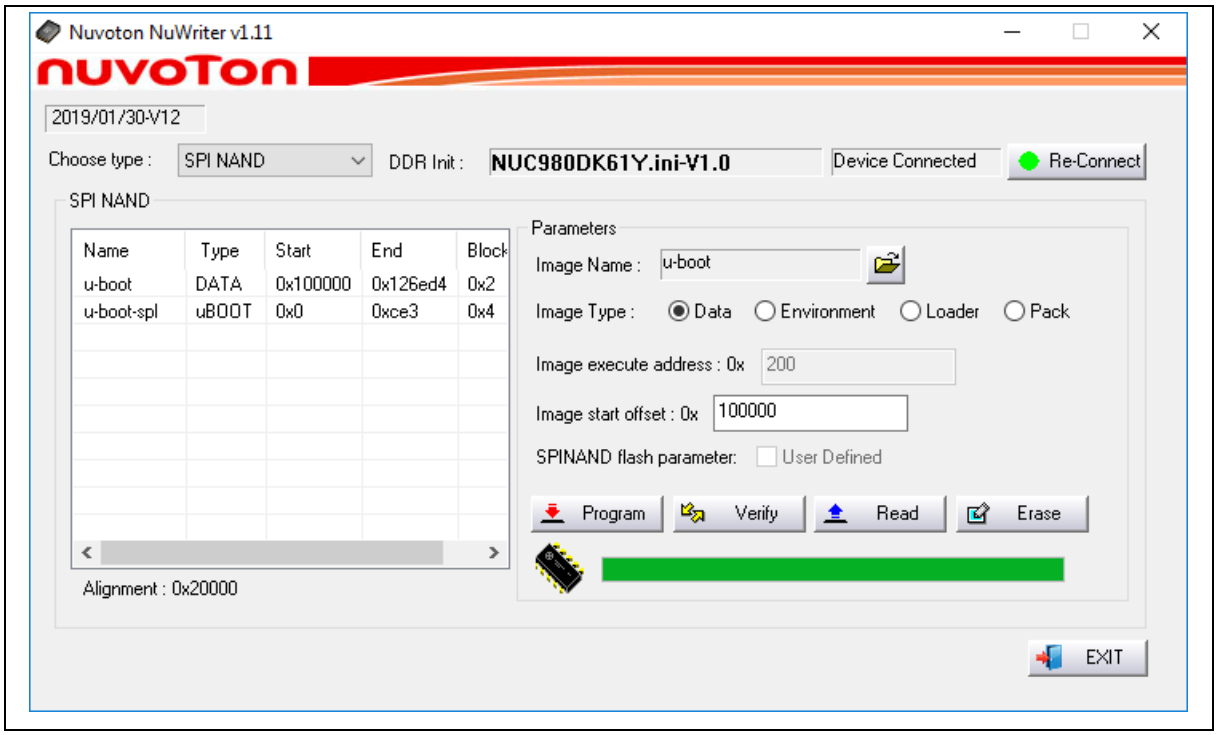


Figure 4-14 Download u-boot to SPI NAND

uImage

Download the image of data type into SPI NAND Flash at the specified address, depending on the value of image start offset (aligned on block size boundary, block size is based on SPI NAND specifications). If image start offset is equal to 0x200000, download the image of data into SPI NAND Flash at the address 0x200000.

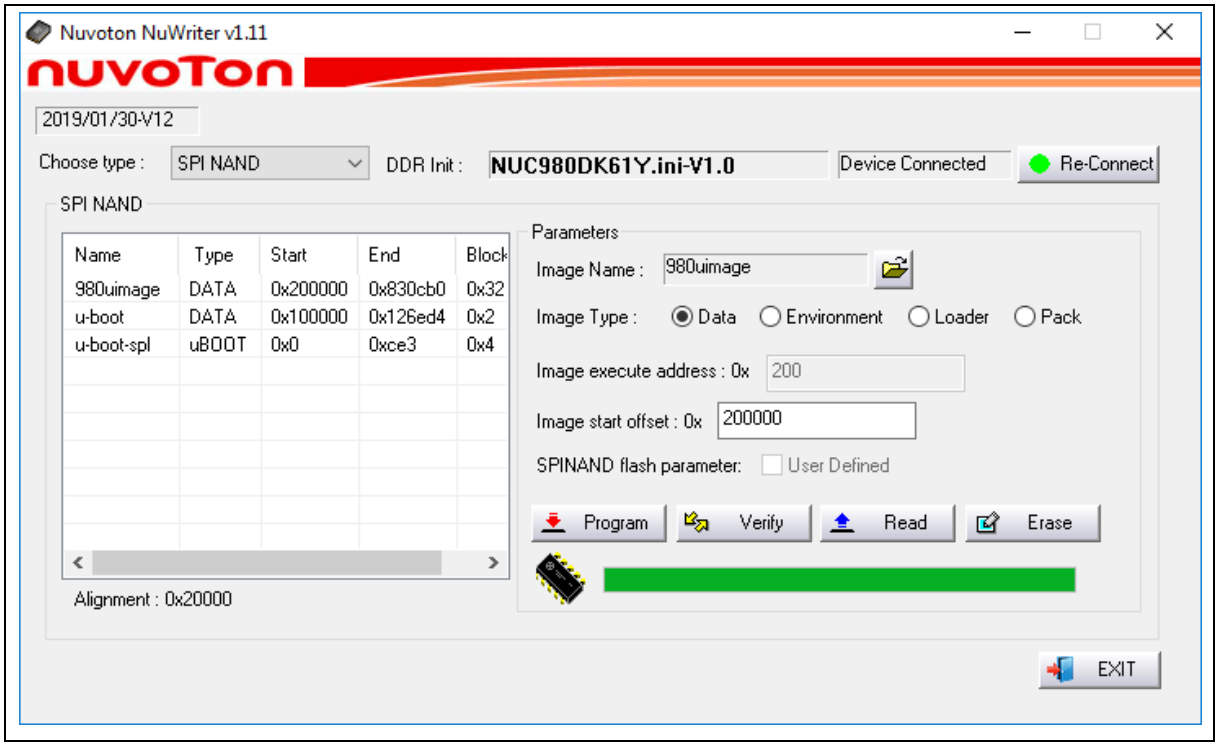


Figure 4-15 Download uImage to SPI NAND

environment

Loader Type is set uboot environment variables, the image of environment type into SPI NAND Flash in the specified address. U-Boot reads environment variables file to set the environment. If image start offset is equal to 0x80000, download the image of data into SPI NAND Flash at the address 0x80000.

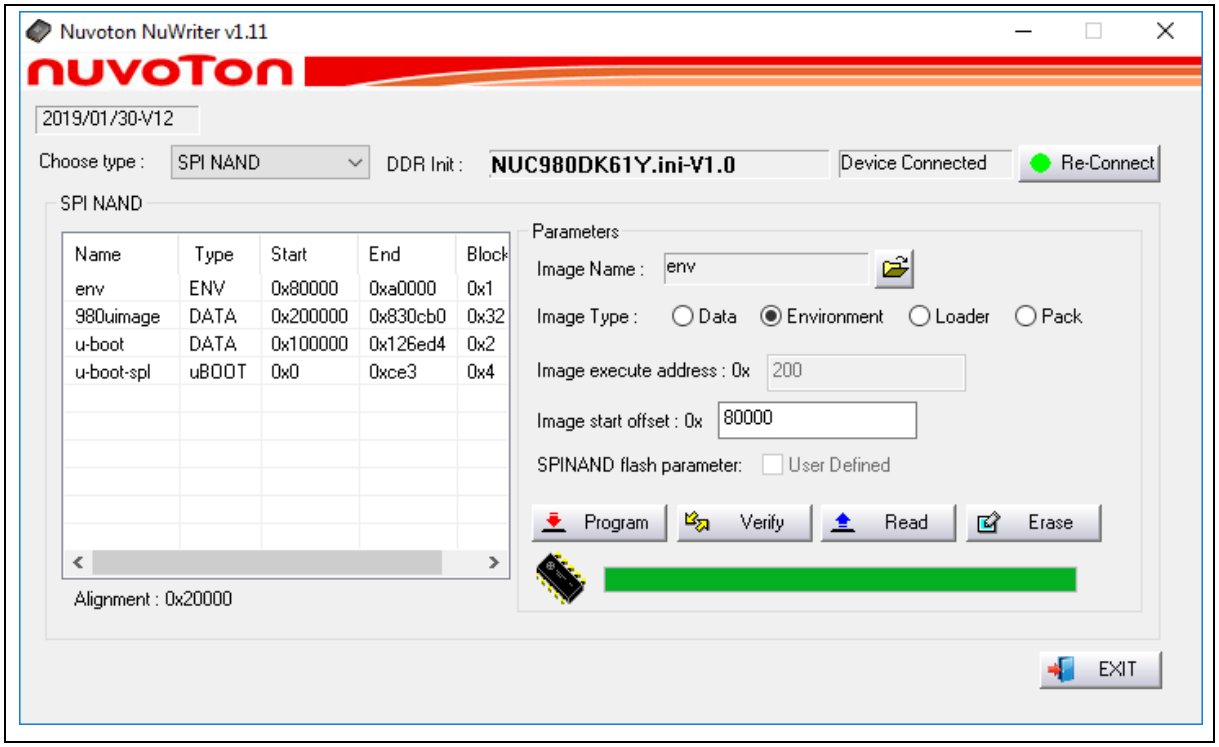


Figure 4-16 Download Environment to SPI NAND

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-IIoT-NUC980 environment variables:

```
baudrate=115200
bootdelay=1
stderr=serial
stdin=serial
stdout=serial
setspi=sf probe 0 30000000
loadkernel=sf read 0x7fc0 0x200000 0x800000
bootcmd=run setspi;run loadkernel;bootm 0x7fc0
```

5 NUMAKER-IIOT-NUC980 SCHEMATICS

5.1 Block Diagram Schematic

Figure 5-1 shows the Block Diagram of the NuMaker-IIoT-NUC980 board.

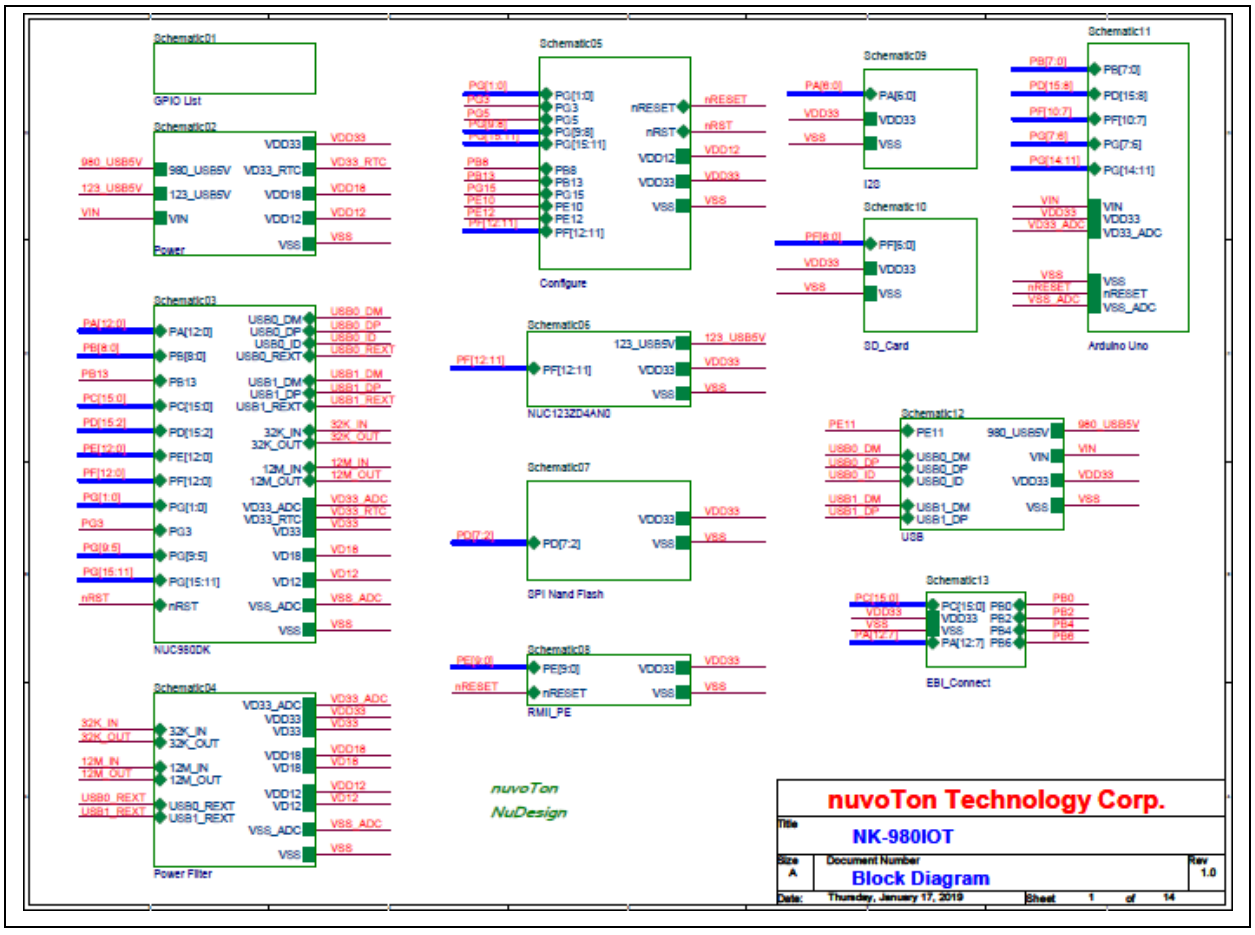
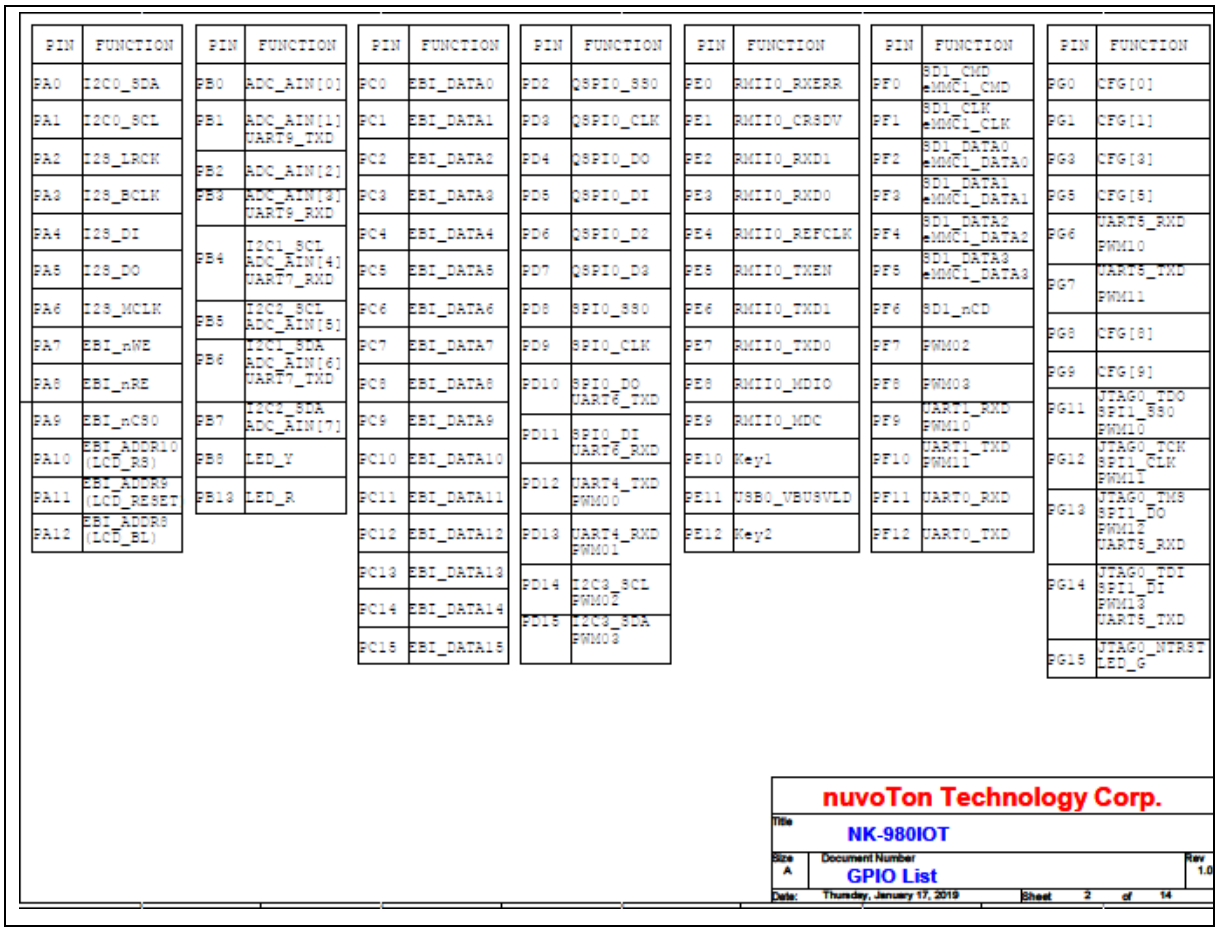


Figure 5-1 NuMaker-IIoT-NUC980 Board Block Diagram

5.2 GPIO List

Figure 5-2 shows the GPIO List of the NuMaker-IIoT-NUC980 board.



PIN	FUNCTION
PA0	I2C0_SDA
PA1	I2C0_SCL
PA2	I2S_LRCK
PA3	I2S_BCLK
PA4	I2S_DI
PA5	I2S_DO
PA6	I2S_MCLK
PA7	EBI_nWE
PA8	EBI_nRE
PA9	EBI_nCS0
PA10	EBI_ADDR10 (LCD_RS)
PA11	EBI_ADDR9 (LCD_RESET)
PA12	EBI_ADDR8 (LCD_BL)

PIN	FUNCTION
PB0	ADC_AIN[0]
PB1	ADC_AIN[1] UART9_TXD
PB2	ADC_AIN[2]
PB3	ADC_AIN[3] UART9_RXD
PB4	I2C1_SCL ADC_AIN[4] UART7_RXD
PB5	I2C2_SCL ADC_AIN[5]
PB6	I2C1_SDA ADC_AIN[6] UART7_TXD
PB7	I2C2_SDA ADC_AIN[7]
PB8	LED_Y
PB13	LED_R

PIN	FUNCTION
PC0	EBI_DATA0
PC1	EBI_DATA1
PC2	EBI_DATA2
PC3	EBI_DATA3
PC4	EBI_DATA4
PC5	EBI_DATA5
PC6	EBI_DATA6
PC7	EBI_DATA7
PC8	EBI_DATA8
PC9	EBI_DATA9
PC10	EBI_DATA10
PC11	EBI_DATA11
PC12	EBI_DATA12
PC13	EBI_DATA13
PC14	EBI_DATA14
PC15	EBI_DATA15

PIN	FUNCTION
PD2	QSPI0_SS0
PD3	QSPI0_CLK
PD4	QSPI0_D0
PD5	QSPI0_D1
PD6	QSPI0_D2
PD7	QSPI0_D3
PD8	SPI0_SS0
PD9	SPI0_CLK
PD10	SPI0_D0 UART6_TXD
PD11	SPI0_D1 UART6_RXD
PD12	UART4_TXD PWM00
PD13	UART4_RXD PWM01
PD14	I2C3_SCL PWM02
PD15	I2C3_SDA PWM03

PIN	FUNCTION
PE0	RMII0_RXERR
PE1	RMII0_CRSDV
PE2	RMII0_RXD1
PE3	RMII0_RXD0
PE4	RMII0_REFCLK
PE5	RMII0_TXEN
PE6	RMII0_TXD1
PE7	RMII0_TXD0
PE8	RMII0_MDIO
PE9	RMII0_MDC
PE10	Key1
PE11	USB0_VBUSVLD
PE12	Key2

PIN	FUNCTION
PF0	SD1_CMD eMMC1_CMD
PF1	SD1_CLK eMMC1_CLK
PF2	SD1_DATA0 eMMC1_DATA0
PF3	SD1_DATA1 eMMC1_DATA1
PF4	SD1_DATA2 eMMC1_DATA2
PF5	SD1_DATA3 eMMC1_DATA3
PF6	SD1_nCD
PF7	PWM02
PF8	PWM03
PF9	UART1_RXD PWM10
PF10	UART1_TXD PWM11
PF11	UART0_RXD
PF12	UART0_TXD

PIN	FUNCTION
PG0	CFG[0]
PG1	CFG[1]
PG3	CFG[3]
PG5	CFG[5]
PG6	UART5_RXD PWM10
PG7	UART5_TXD PWM11
PG8	CFG[8]
PG9	CFG[9]
PG11	JTAG0_TDO SPI1_SS0 PWM10
PG12	JTAG0_TCK SPI1_CLK PWM11
PG13	JTAG0_TMS SPI1_D0 PWM12 UART5_RXD
PG14	JTAG0_TDI SPI1_D1 PWM13 UART5_TXD
PG15	JTAG0_NTRST LED_G

nuvoTon Technology Corp.

Title: NK-980IOT

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

Date: Thursday, January 17, 2019 | Sheet 2 of 14

Figure 5-2 GPIO List

5.3 Power

Figure 5-3 shows the power circuit of the NuMaker-IIoT-NUC980 board.

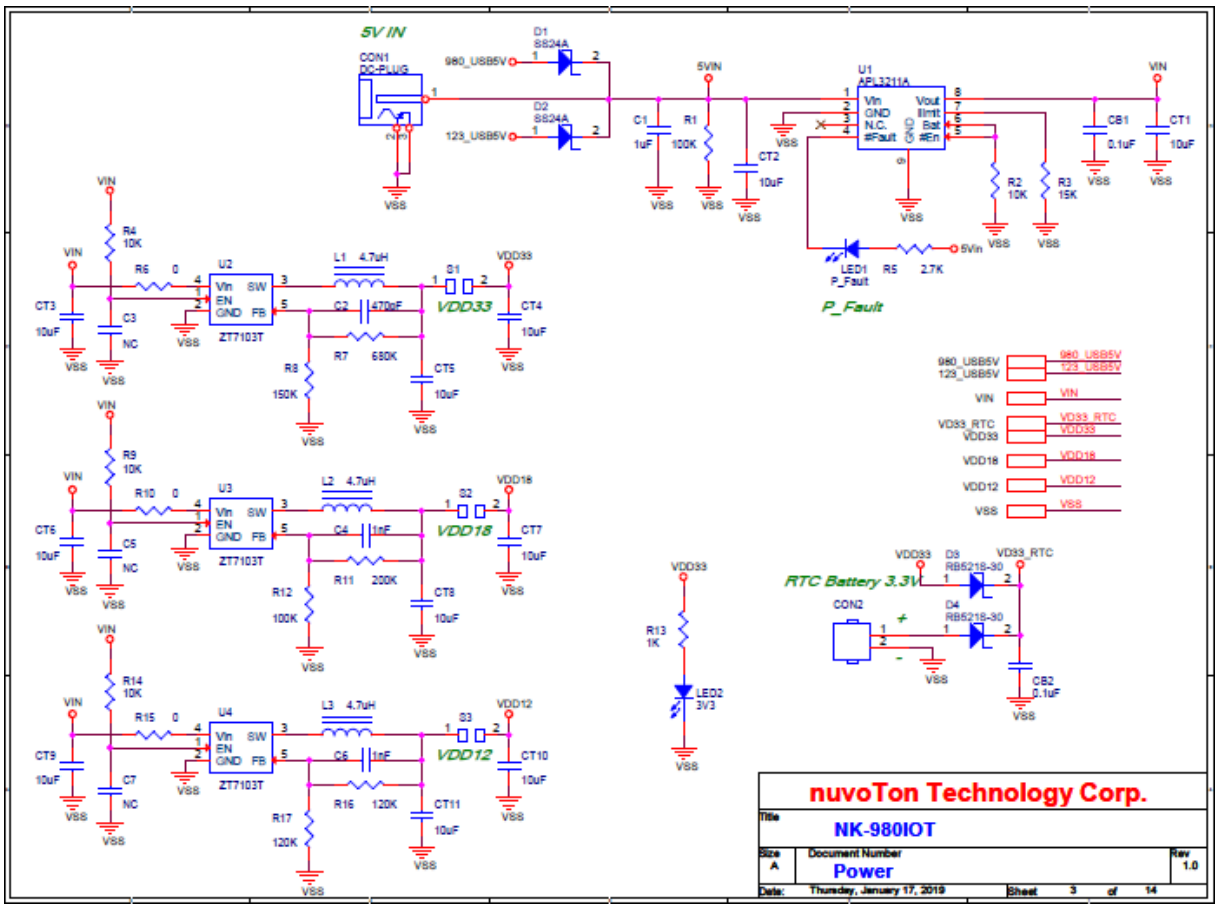


Figure 5-3 Power

5.4 NUC980DK

Figure 5-4 shows the NUC980DK net name of the NuMaker-IIoT-NUC980 board.

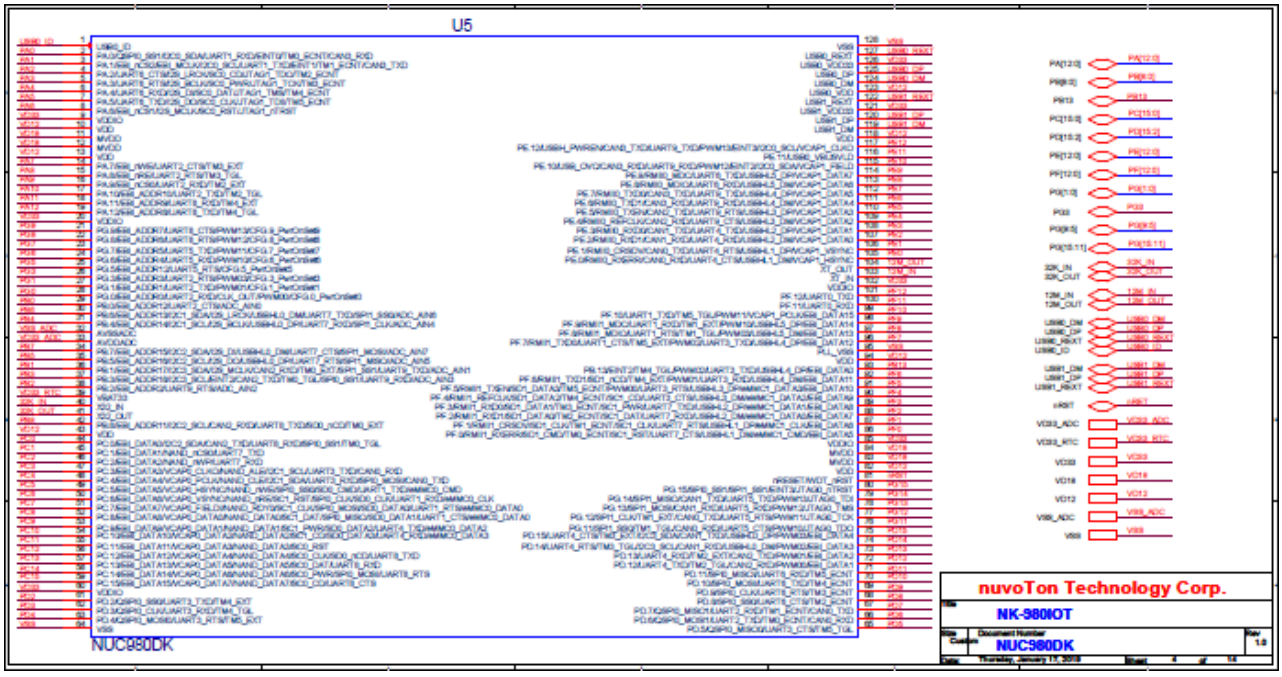


Figure 5-4 NUC980DK

5.5 Power Filter

Figure 5-5 shows the power filter of the NuMaker-IIoT-NUC980 board.

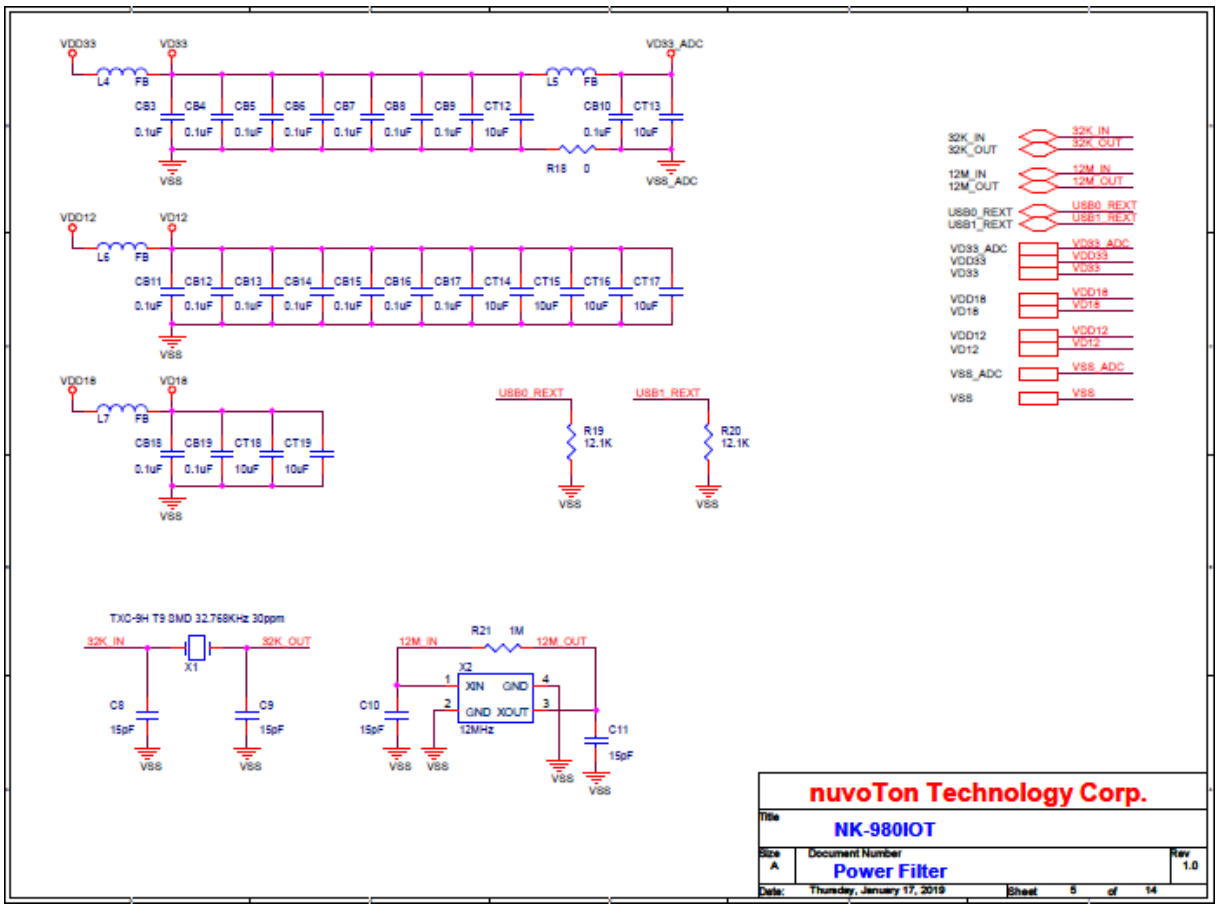


Figure 5-5 Power Filter

5.6 Power-on Setting

Figure 5-6 shows the power-on setting of the NuMaker-IIoT-NUC980 board.

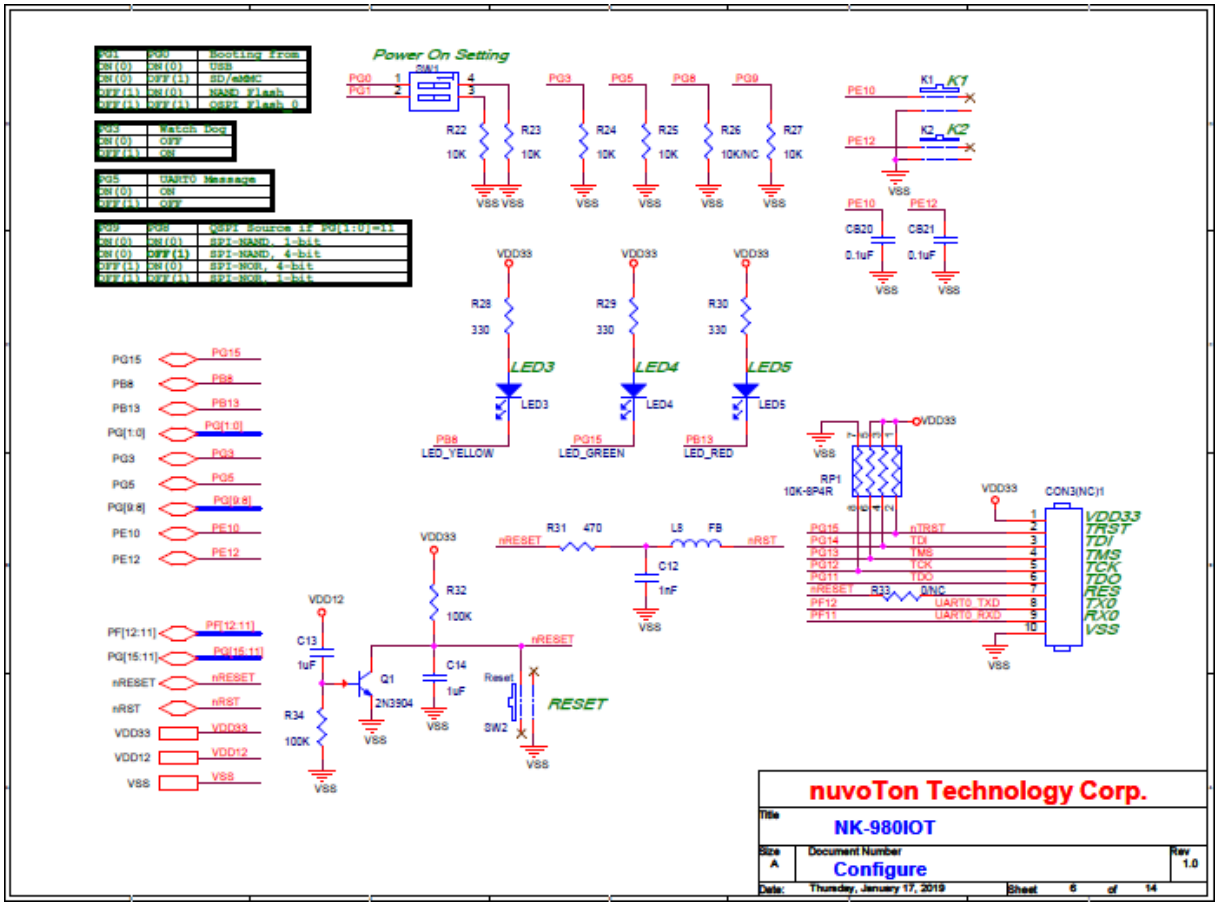


Figure 5-6 Power-on Setting

5.7 NUC123ZD4AN0

Figure 5-7 shows the NUC123 VCOM circuit of the NuMaker-IIoT-NUC980 board.

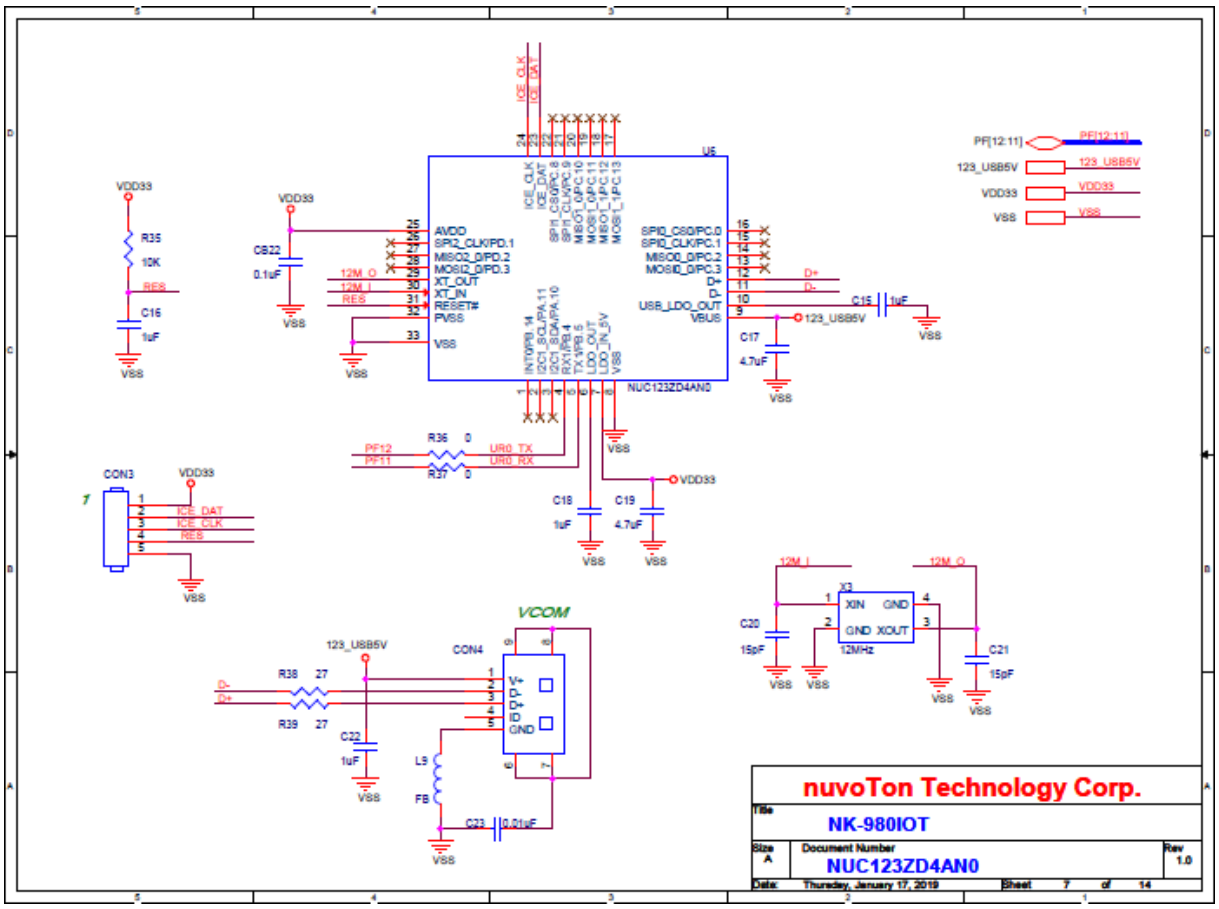


Figure 5-7 NUC123ZD4AN0

5.8 Memory

Figure 5-8 shows the QSPI0 (only SPI NAND Flash device mounted) circuit of the NuMaker-IIoT-NUC980 board.

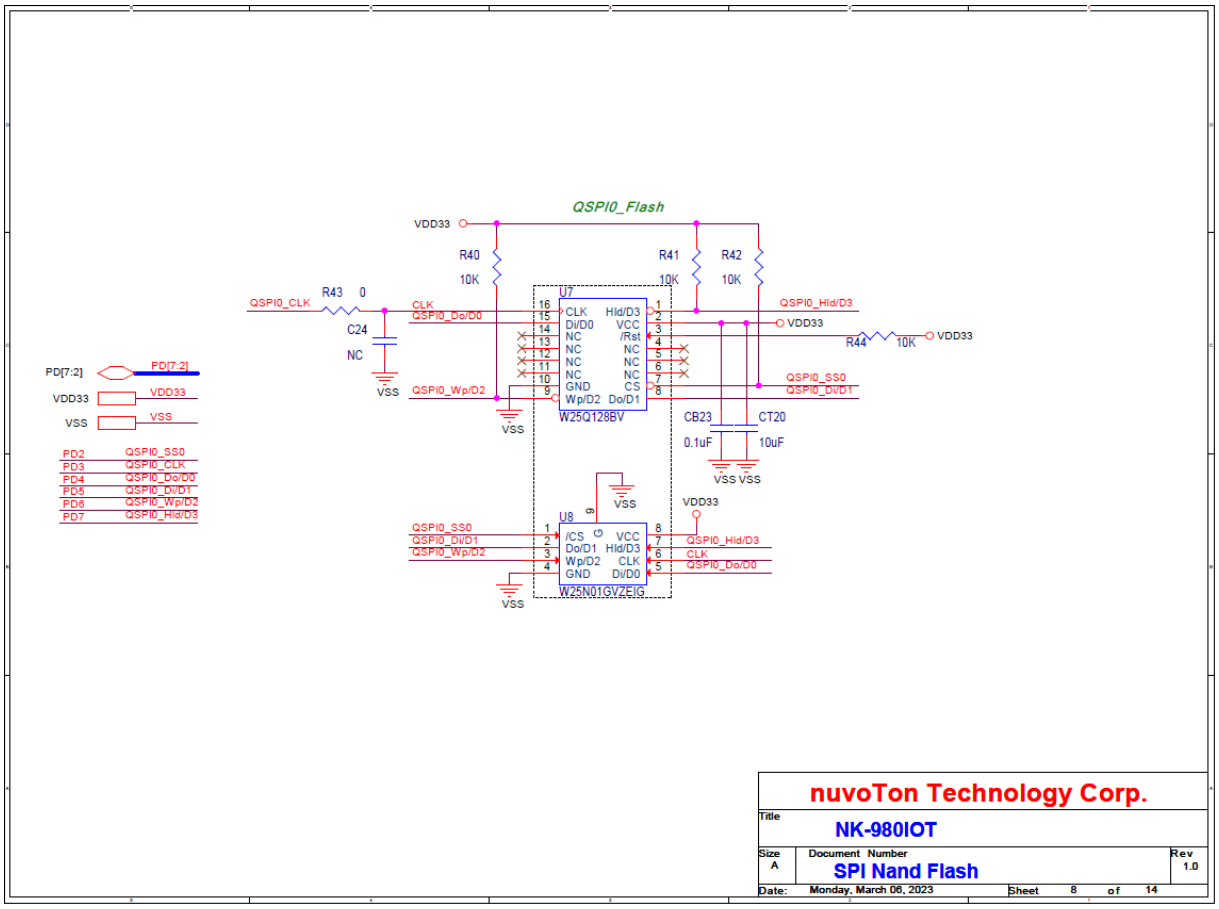


Figure 5-8 Memory

5.9 RMII_PE

Figure 5-9 shows the RMII_PE circuit of the NuMaker-IIoT-NUC980 board.

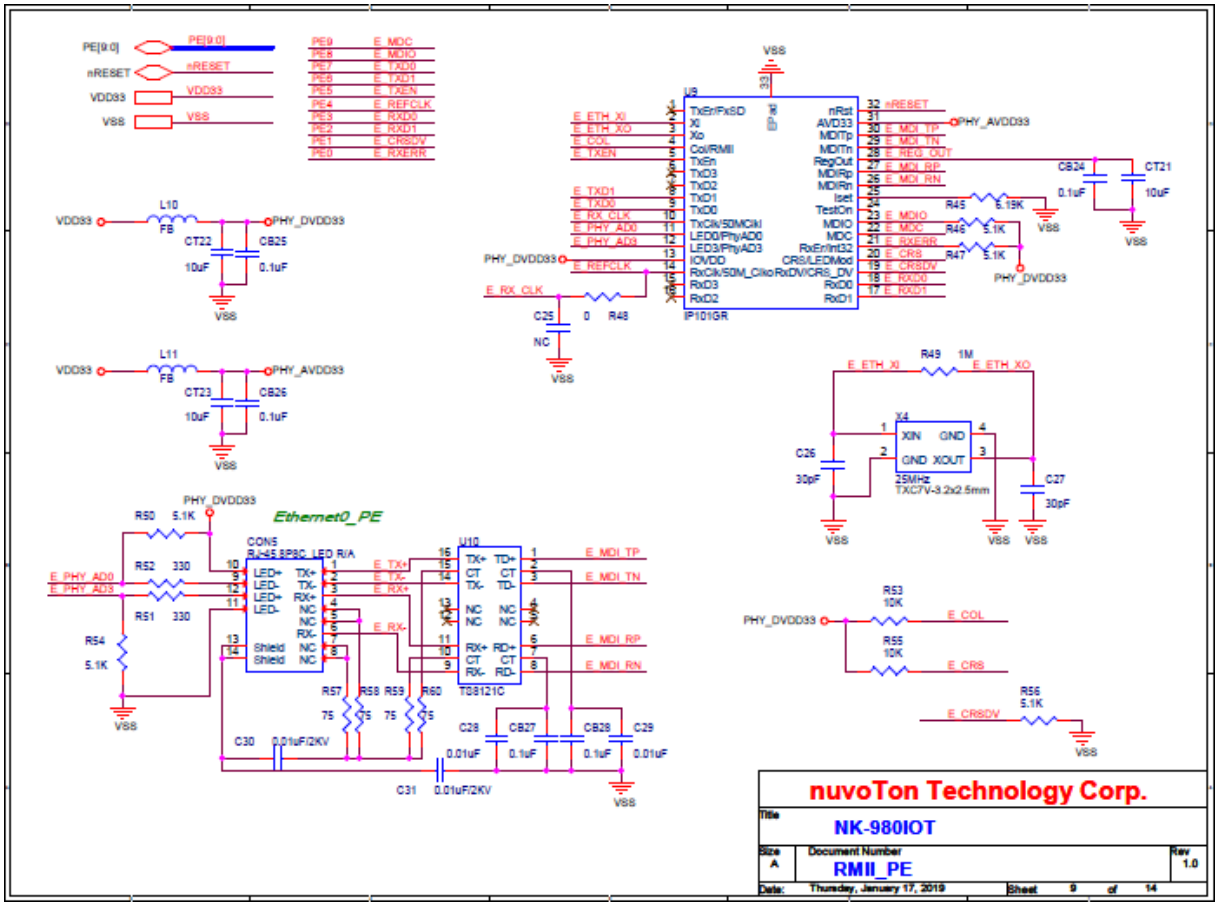


Figure 5-9 RMII_PE

5.10 Audio Codec

Figure 5-10 shows the NAU8822L Audio Codec circuit of the NuMaker-IIoT-NUC980 board.

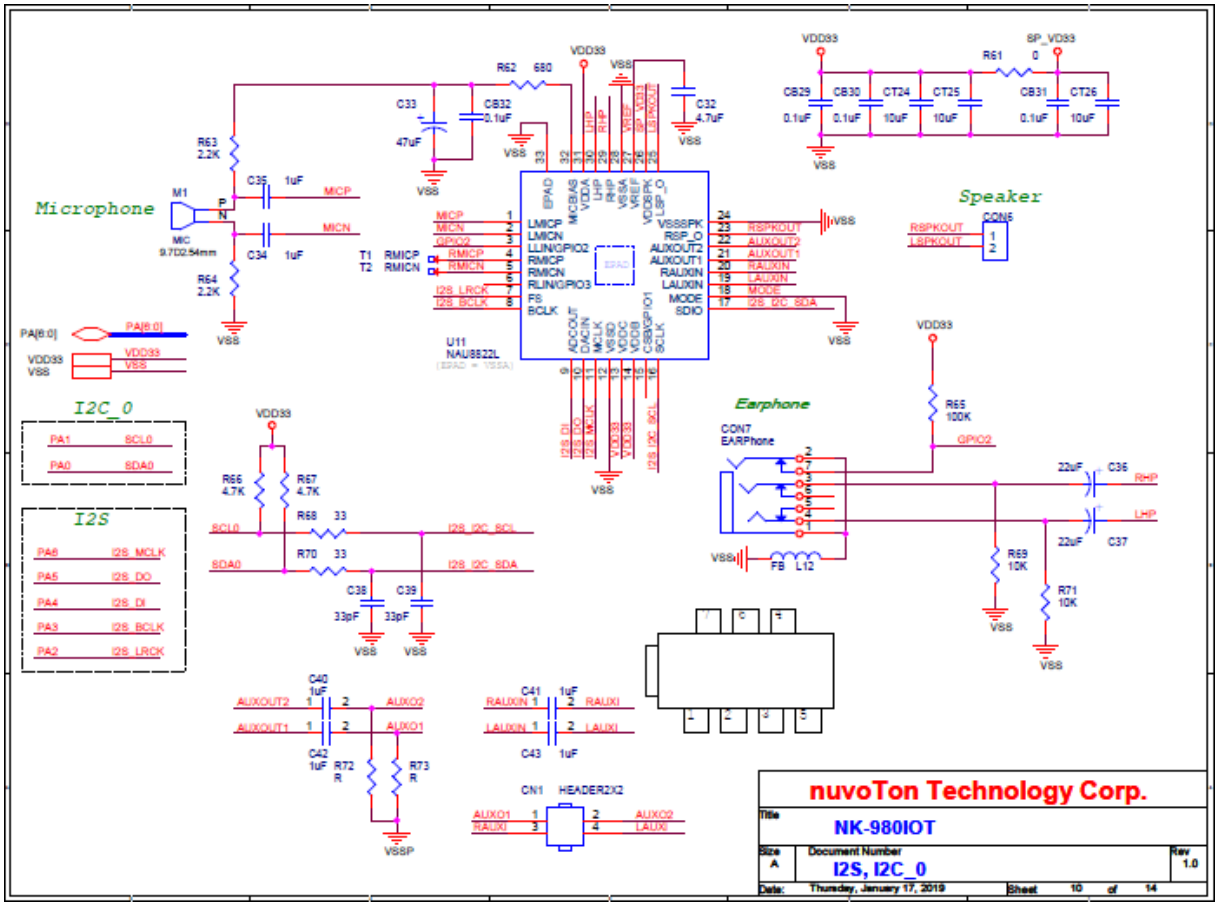


Figure 5-10 Audio Codec

5.11 SD1/eMMC1

Figure 5-11 shows the SD1 card slot circuit of the NuMaker-IIoT-NUC980 board.

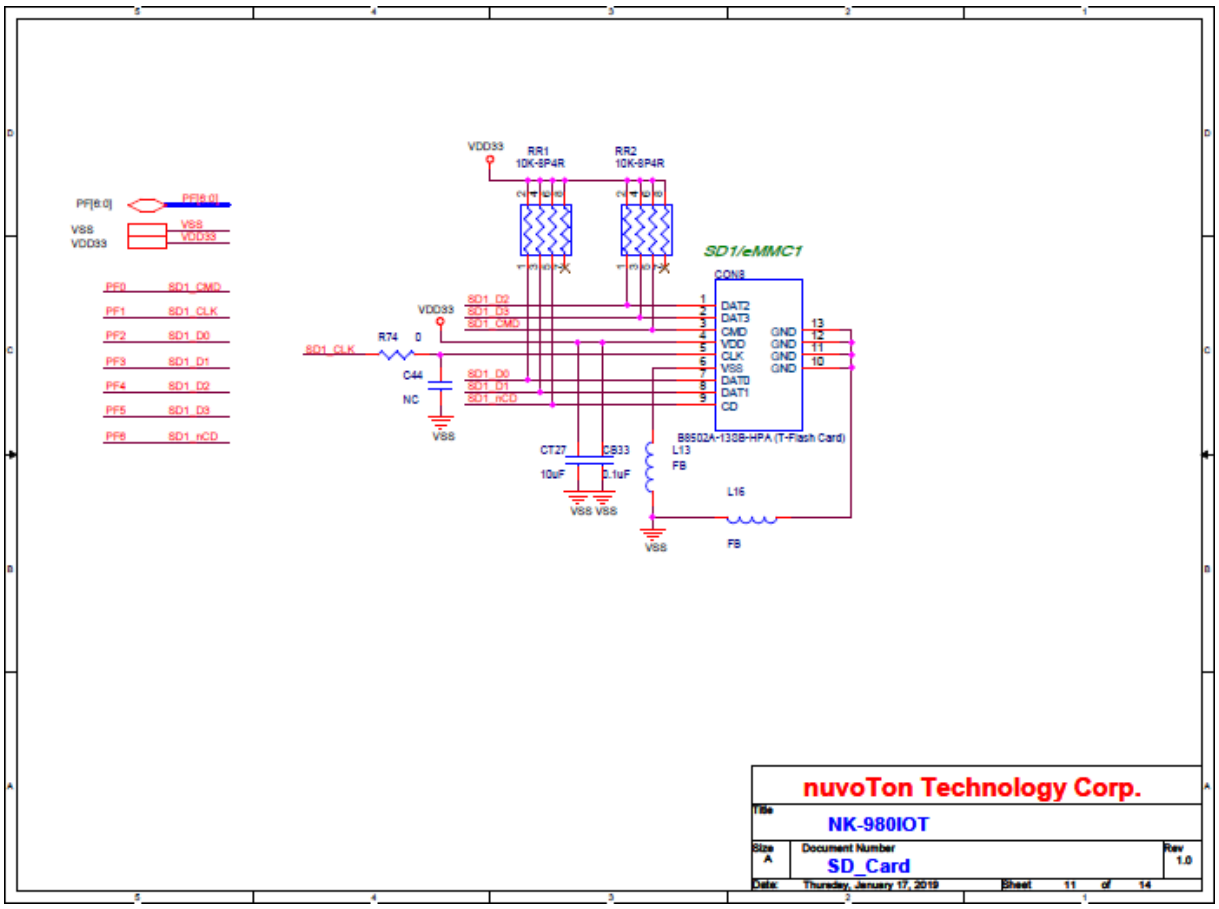


Figure 5-11 SD1/eMMC1

5.12 Arduino Uno Interface

Figure 5-12 shows the Arduino Uno interface of the NuMaker-IIoT-NUC980 board.

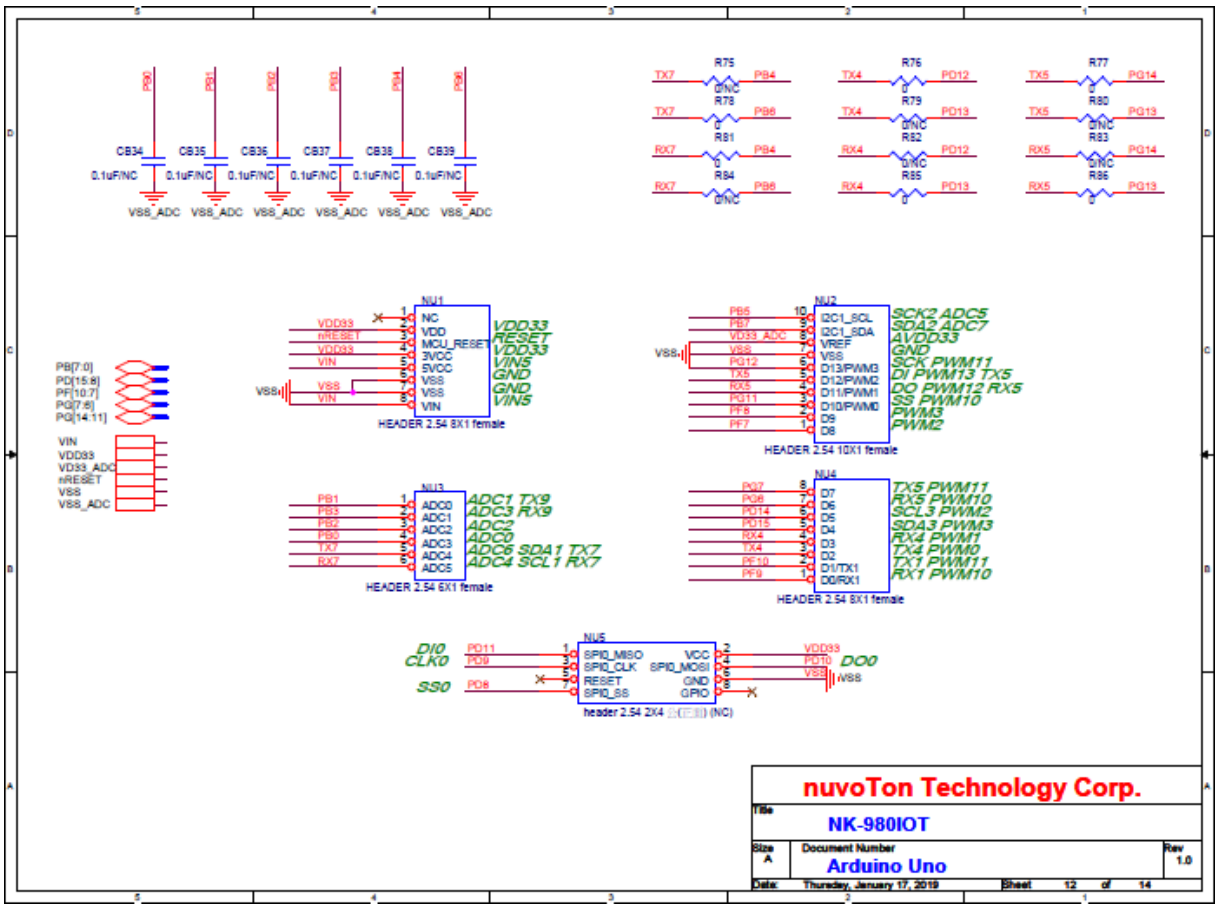


Figure 5-12 Arduino Uno interface

5.13 USB

Figure 5-13 shows the USB 0/1 circuit of the NuMaker-IIoT-NUC980 board.

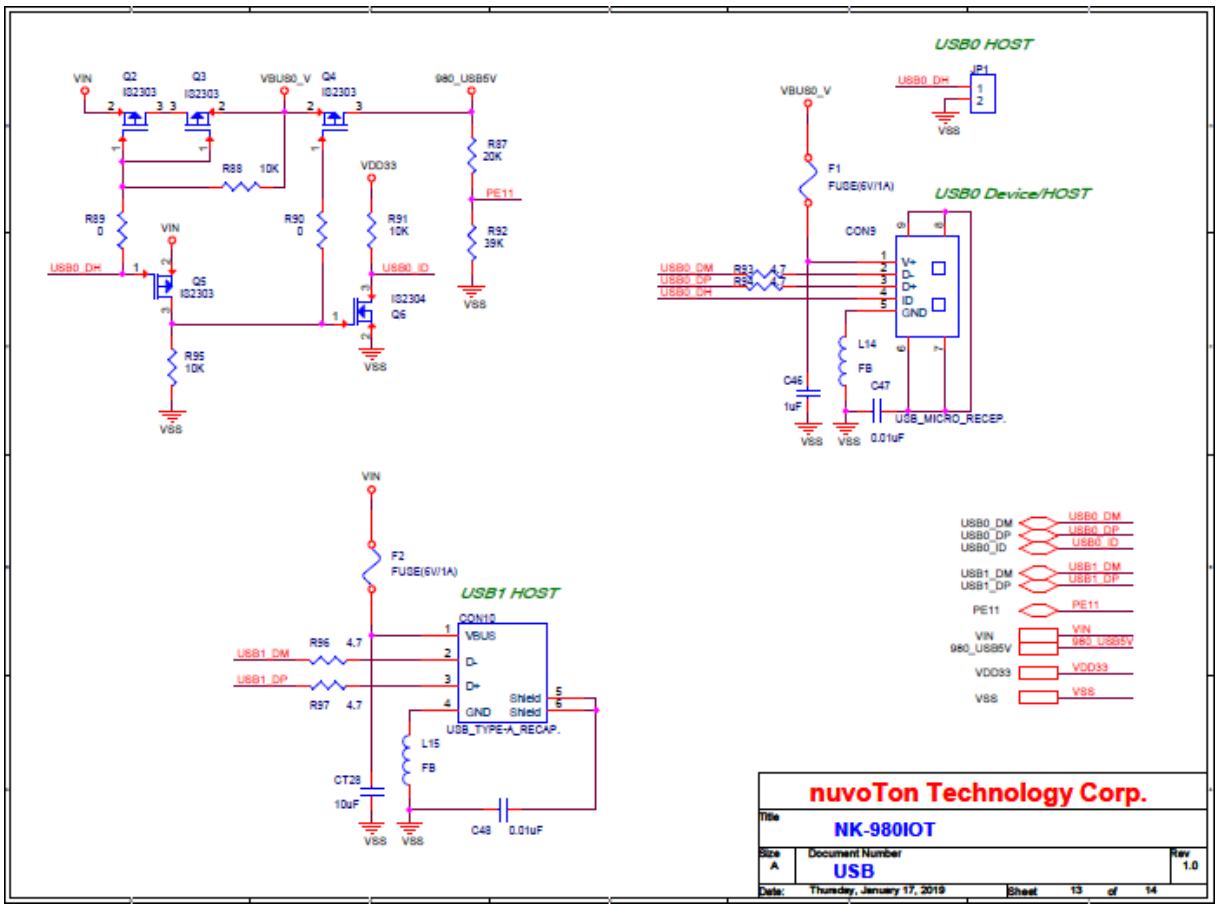


Figure 5-13 USB

5.14 Expand EBI Interface

Figure 5-14 shows the Expand EBI Interface circuit of the NuMaker-IIoT-NUC980 board.

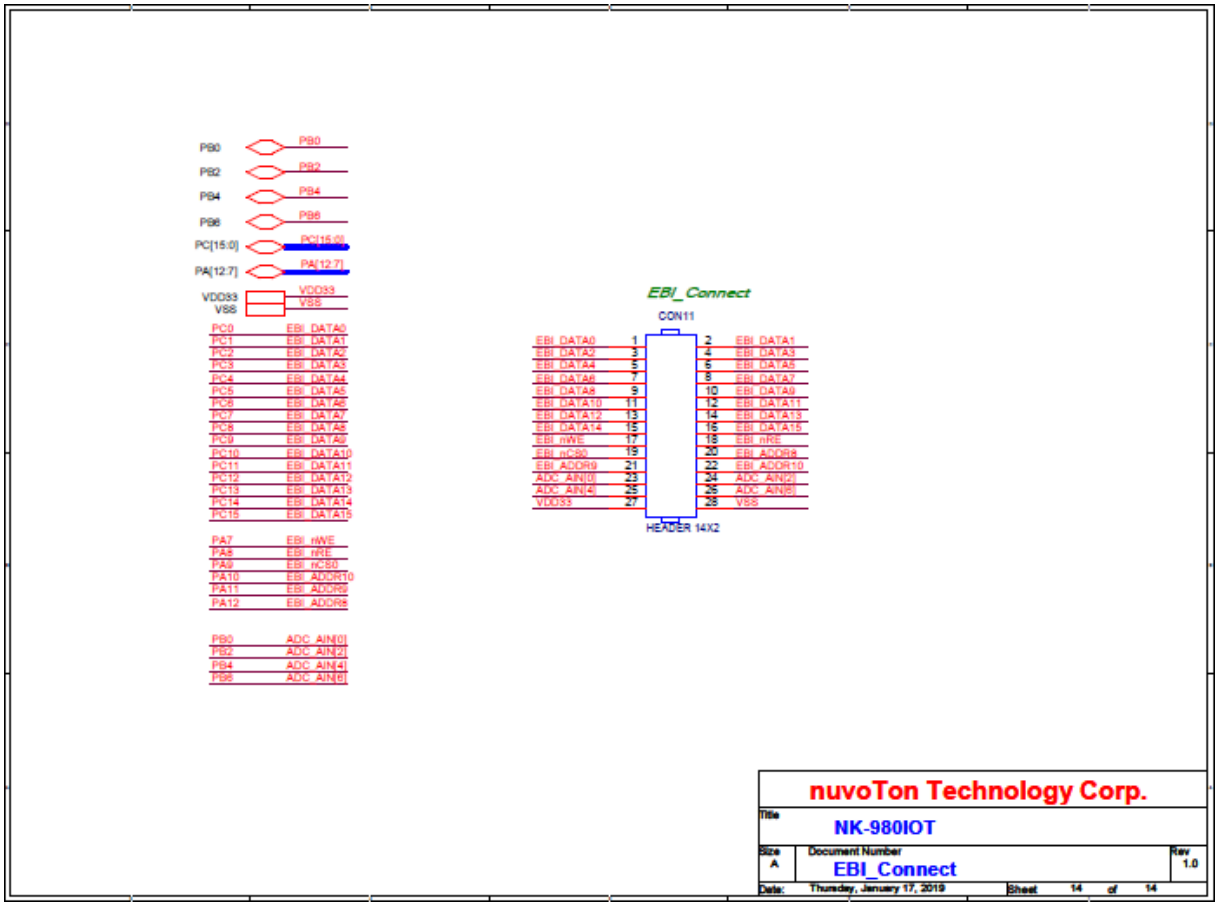


Figure 5-14 Expand EBI Interface

5.15 PCB Placement

Figure 5-15 and Figure 5-16 show the front and rear placement of NuMaker-IIoT-NUC980.

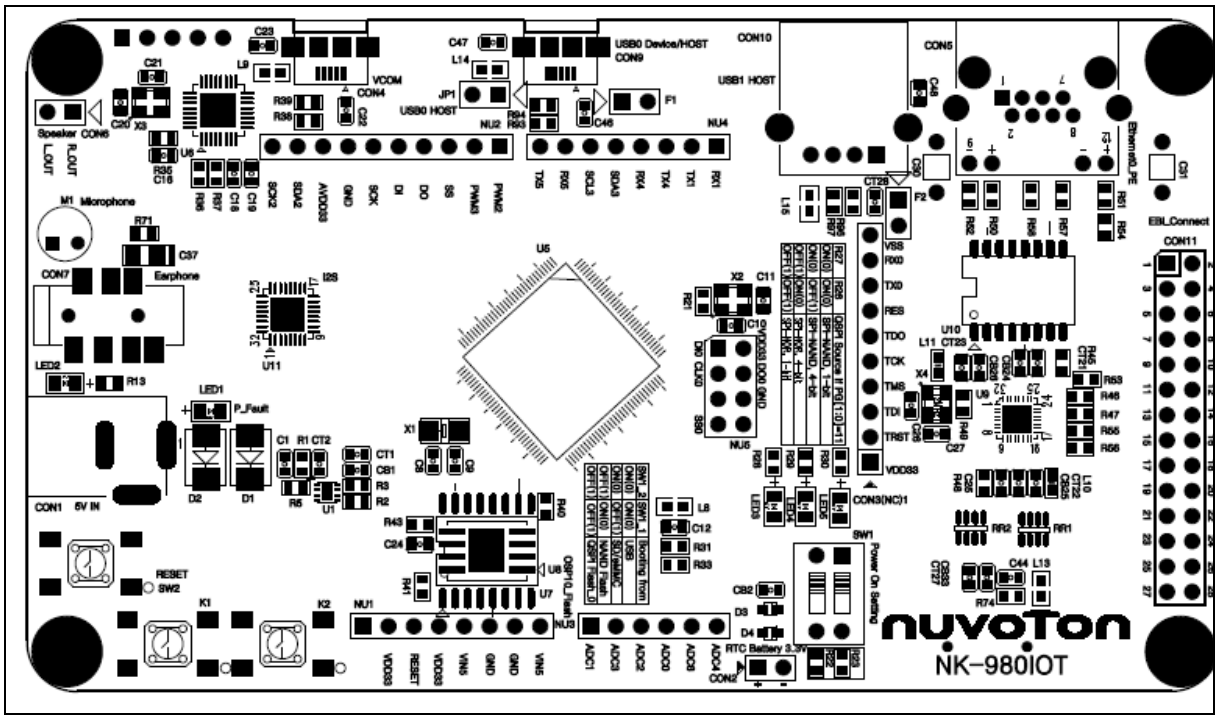


Figure 5-15 Front Placement

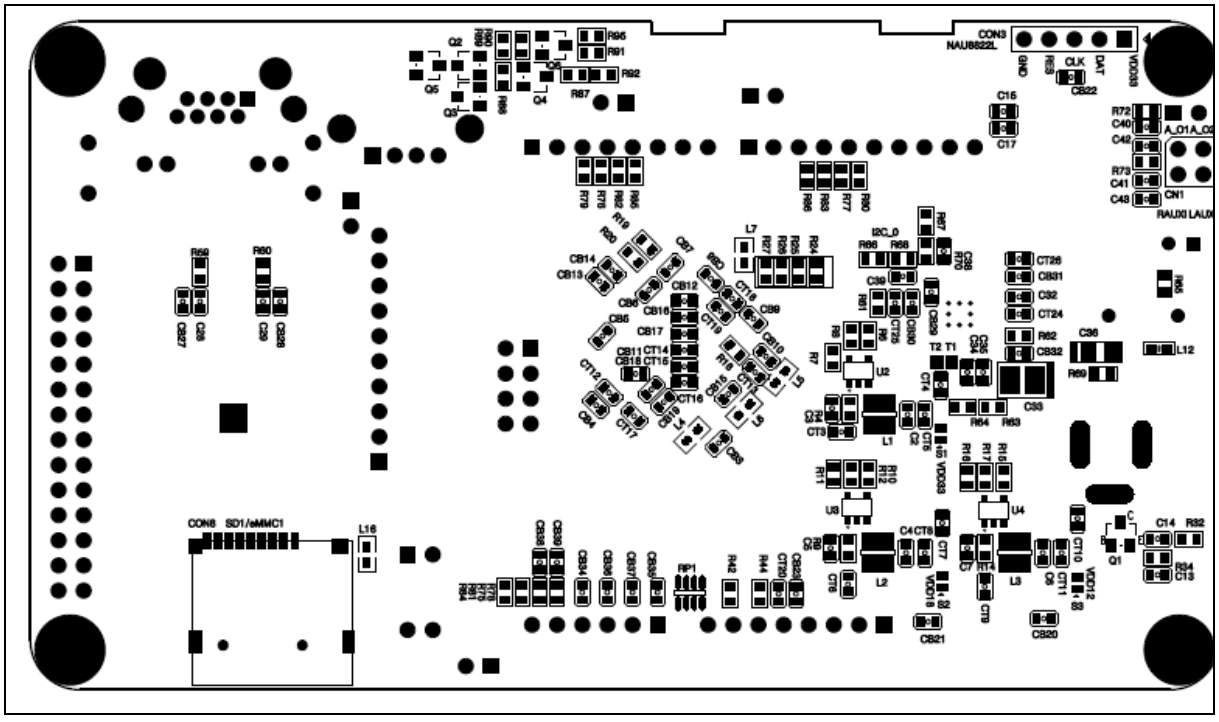


Figure 5-16 Rear Placement

6 REVISION HISTORY

Date	Revision	Description
2021.05.20	1.00	● Initial version.
2023.03.14	1.01	● Updated document format ● Modified SPI-NAND part number W25N01GVZE1G to W25N01GVZEIG ● Modified Figure 5-8 Memory

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.