ARM® Cortex®-M0

32-bit Microcontroller

NuMicro® Family
NuMaker UNO
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

Arduino is an open-source electronics platform based on easy-to-use hardware and software. The NuMicro® NuMaker UNO Evaluation Board is an Arduino compatible hardware using NuMicro® microcontroller (MCU) as the MCU. Its function can be extended with Arduino add-ons. With the Arduino IDE, users can develop their applications and leverage large number of open source samples.

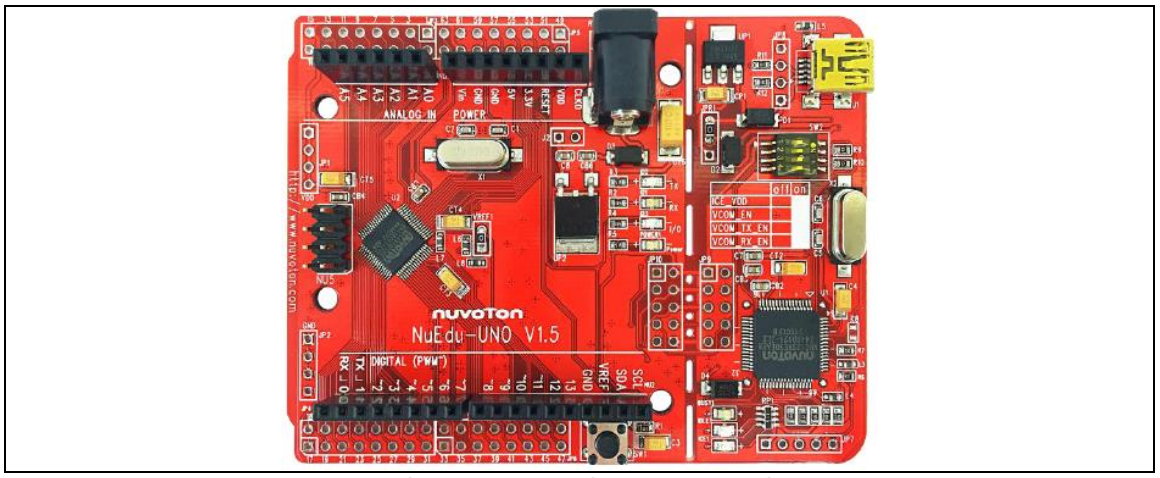


Figure 1-1 NuMaker UNO Board

The NuMaker UNO is a specific development tool for NuMicro® NUC131 series by which users can develop and verify the application program easily. Its purpose is to provide a platform for development and learning. With ADC, PWM, I²C, SPI, UART and other peripheral functions, user can set different functions on the NuMaker UNO development kit, or increase the peripheral functions according to the user needs on the development kit. The NuMaker UNO includes two portions: NuMaker UNO (an evaluation board) and Nu-Link-Me (Debug Adaptor). With the NuMaker UNO, users do not need additional ICE or debug equipment.

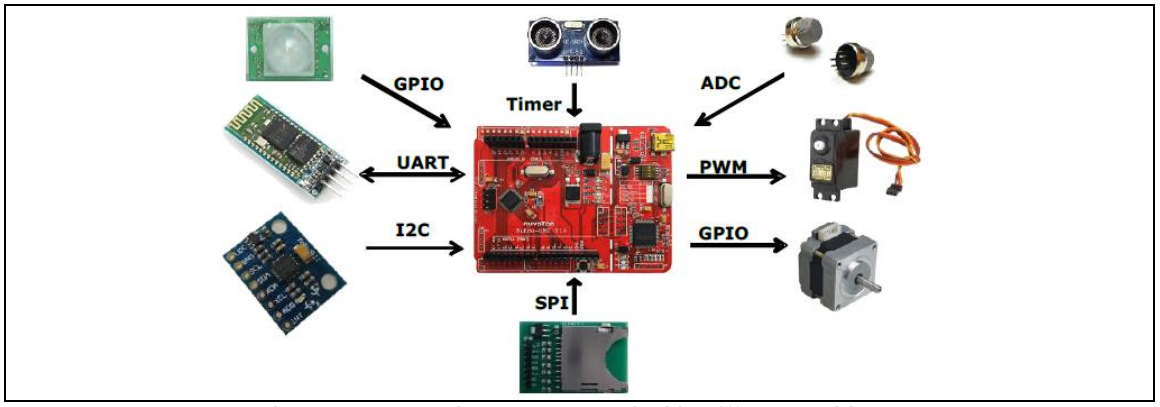


Figure 1-2 NuMaker UNO Board with Different Add-ons

The NuMaker UNO is pin to pin compatible with Arduino UNO board.

Digital pins provide UART, I2C, LED, INT, and 10-channels PWM. In addition, the extended pins of MCU NUC131SD2AE provide 24-channels PWM and 6-channels UART. Clock Output (CLKO) is also available on extra pin.

2 FEATURES

The NuMaker UNO development board provides the following features:

- Wide range of development tools for Learning/applications/debug
- Portable package development and debugging tools
- Rich MCU peripheral functions, such as ADC, PWM, I²C, SPI, and UART
- Able to connect to different application modules due to high scalability
- Supports pin to pin compatible with Arduino UNO revision 3
- Supports USB virtual serial port (VCOM)
- Supported by a wide choice of Integrated Development Environments (IDEs) including Arduino IDE, IAR EWARM and Keil RVMDK IDE
- Extension resources:
  - NuMicro® Morpho extension pin headers for full access to all MCU NUC131 I/O
  - On-board Nu-Link debugger/programmer with SWD connector
- Flexible board power supply:
  - USB VBUS (jumper JPR1 can be used to select 5V or 3.3V)
  - External VIN (7V ~ 12V) supply voltage from transformer converted into 5V
  - External 2.5 ~ 5.5V supply voltage from other power source input to V_{DD} pin
- LEDs Status
  - Power, TX, RX, ICE and user status
- One push button: RESET

3 INTRODUCTION TO NUMAKER UNO DEVELOPMENT BOARD

Figure 3-1 and Figure 3-2 show the NuMaker UNO development board, in which the left portion is called NuMaker UNO target board and the right portion is Debug Adaptor called Nu-Link-Me.

The NuMaker UNO is similar to other development boards. Users can use it to develop and verify applications to emulate the real behavior. NuMaker UNO can be a real system controller to design user's target system. NuMaker UNO is pin to pin compatible with Arduino UNO revision 3. The left portion is NuMaker UNO target board where the NUC131 series MCU is mounted on. The NUC131 series provides 24-ch PWM as well as 6-ch UART, and commonly used peripherals such as Timer, WDT, SPI, I^2C and ADC. The NUC131 series also provides CAN communication interface.

The right portion is the debug adaptor called Nu-Link-Me which connects the USB port from PC to user's target system (via the Serial Wired Debug port) and allows users to program and debug embedded programs on the target hardware. In addition to loading external application, it also provides a virtual serial port (VCOM) functions. The debug messages through user-friendly Nu-Link–Me is displayed on the computer screen. For detailed description, please refer to section 3.1.2. The NuMaker UNO supports Arduino IDE, Keil, and IAR. For detailed installation and setup, please refer to section 4.1.

The NuMaker UNO development board provides the Arduino pin-out definition and extended connectors for each pin from the NUC131SD2AE MCU. It can be used to connect the application circuit board. The target board also supports a wide range of power supplies, such as from ICE, V_{DD} & GND (JP1 & JP2) or from 7V ~ 12V transformer. LED status is for power, I/O, TX, RX and ICE, as shown in Figure 3-1.

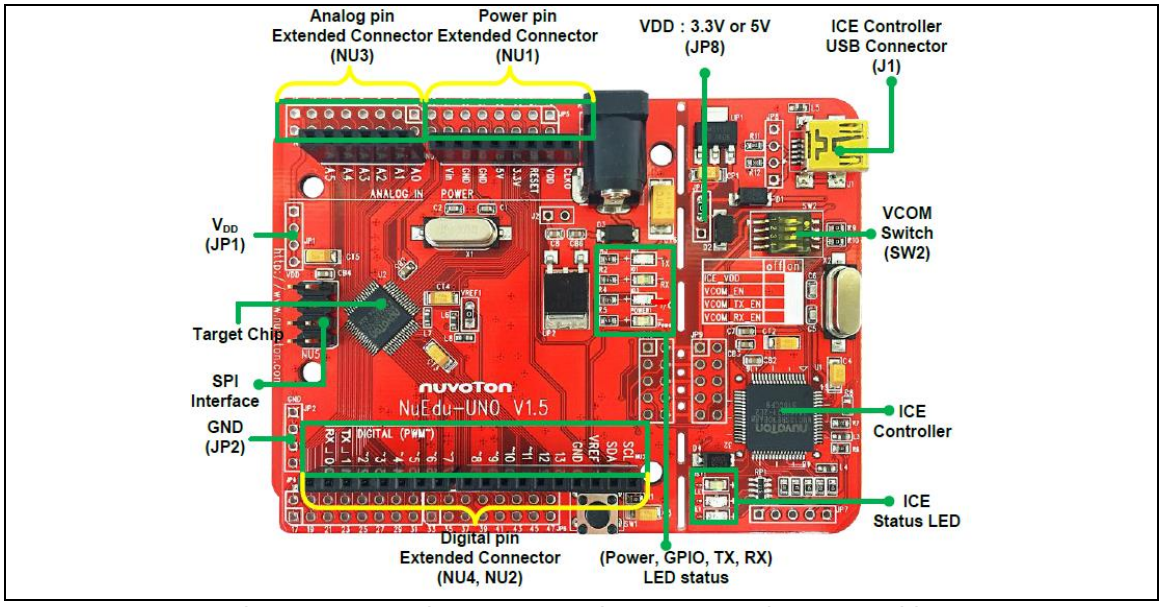


Figure 3-1 NuMaker UNO Development Board – Upper Side

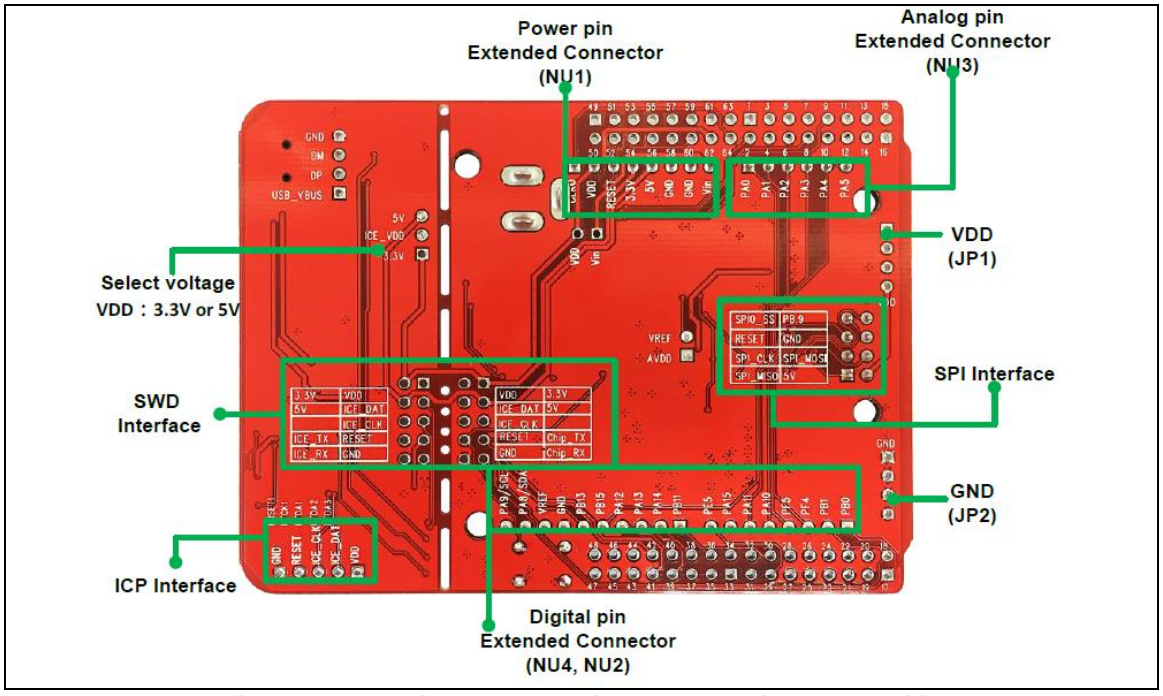


Figure 3-2 NuMaker UNO Development Board – Bottom Side

3.1 NuMaker UNO Jumper Description

3.1.1 Power Settings

There are three methods to use the NuMaker UNO board to provide power to V_{DD}. The first method is through the Nu-Link-Me USB interface. This power will go through LDO voltage regulator to 3.3V, JPR1 can be used to adjust V_{DD} power to 5V or 3.3V. The second method is through the JP1 on development board to V_{DD} by DC 2.5V ~ 5.5V power supply. The third method is through transformer (7V ~ 12V) and then the voltage is converted into 5V through the step-down circuit. Please refer to the table below.

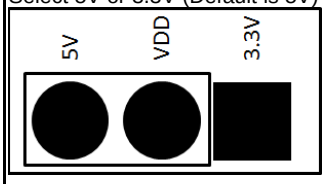
Model	JPR1 (Selection Voltage)	JP1 (V_{DD} Provided Voltage)	MCU Voltage
USB	Select 5V or 3.3V (Default is 5V) 	USB, target board will be able to get the supply voltage. (SW2 pin1 need on)	DC 5V or 3.3V
Other power for V_{DD} pin	X	External power supply DC 2.5 V ~ 5.5 V.	Voltage by JP1 input
Transformer	X	Transformer DC 7 V ~ 12 V, after pass the step-down circuit, it provides 5V supply to target chip (J2 need short)	DC 5V

Table 3-1 Power Settings

3.1.2 USB Virtual COM Function Setting

- **SW2**: Open the Virtual COM mode for the debug message and supply power to NuMaker UNO.

The Virtual COM function can be used for Arduino IDE, Keil and IAR. To enable VCOM function on Nu-Link Me, turn on all SW2 pins. To enable the UART0 function, turn off pin 2 ~ pin 4 on SW2. Please refer to the table below.

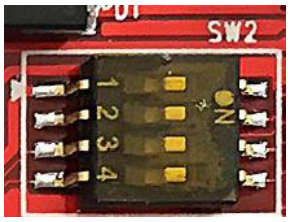
Switch Pin Number	Function Name	UART0 Mode	VCOM Mode	
1	ICE_VCC	On	On	
2	VCOM_En	Off	On	
3	VCOM_TX	Off	On	
4	VCOM_RX	Off	On	

Table 3-2 USB VCOM Setting (Default as VCOM Mode)

3.1.3 Header Description

JP1	V_{DD} connector on NuMaker UNO, can be accessed by an external power supply DC 2.5V ~ 5.5V, in order to provide NuMaker UNO power supply
JP2	GND connector on NuMaker UNO, can be accessed by external power supply GND
J1	Mini USB connector on Nu-Link-Me connected to the PC USB port
SW2	Pin 1 is ICE V_{DD} connected to target chip V_{DD}
CON1	7 V ~ 12 V transformer
J2	Power JACK is connected to V_{DD} transformer, after the step-down circuit then to provide V_{DD} power supply to target Chip
SW1	Reset button on NuMaker UNO
JP10	Connector on target board (NuMaker UNO) for connecting to Nuvoton ICE adaptor (Nu-Link-Me)
JP9	Connector on ICE adaptor (Nu-Link-Me) for connecting to the target board (NuMaker UNO)
JP3 ~ JP6	Show all NUC131SD2AE MCU pin-out on NuMaker UNO
X	Unused

Table 3-3 Header Description

Note: When using the transformer power supply, please turn off all the SW2 pins.

3.2 Pin Assignment for NUC131SD2AE Extended Connectors

The NuMaker UNO provides the NUC131SD2AE target chip on board and the NUC131SD2AE extended connectors (**JP3**, **JP4**, **JP5** and **JP6**) for LQFP64-pin. Table 3-4 shows the pin assignment for NUC131SD2AE.

Pin No	Pin Name	Pin No	Pin Name
01	PB.14,INT0	33	PC.11,PWM1_BRAKE1
02	PB.13	34	PC.10,PWM1_BRAKE0
03	PB.12,CLKO,BPWM1_CH3	35	PC.9,PWM0_BRAKE1
04	PF.5,I2C0_SCL,PWM1_CH5	36	PC.8,PWM0_BRAKE0
05	PF.4,I2C0_SDA,PWM1_CH4	37	PA.15,PWM0_CH3
06	PA.11,I2C1_SCL,PWM1_CH3	38	PA.14,PWM0_CH2
07	PA.10,I2C1_SDA,PWM1_CH2	39	PA.13,PWM0_CH1,UART5_TXD
08	PA.9,I2C0_SCL,UART1_nCTS	40	PA.12,PWM0_CH0,UART5_RXD
09	PA.8,I2C0_SDA,UART1_nRTS	41	PF.7,ICE_DAT
10	PB.4,UART1_RXD	42	PF.6,ICE_CLK
11	PB.5,UART1_TXD	43	AVSS
12	PB.6,UART1_nRTS	44	PA.0,PWM0_CH4,ADC0,I2C1_SCL,UART5_TXD
13	PB.7,UART1_nCTS	45	PA.1,PWM0_CH5,ADC1,I2C1_SDA,UART5_RXD
14	LDO_CAP	46	PA.2,PWM1_CH0,ADC2,UART3_TXD
15	V_{DD}	47	PA.3,PWM1_CH1,ADC3,UART3_RXD
16	V_{SS}	48	PA.4,ADC4
17	PB.0,UART0_RXD	49	PA.5,UART3_RXD,ADC5
18	PB.1,UART0_TXD	50	PA.6,UART3_TXD,ADC6

19	PB.2,UART0_nRTS,TM2_EXT,TM2,PWM1_BRAK	51	PA.7,Vref,ADC7
20	PB.3,UART0_nCTS,TM3_EXT,TM3,PWM1_BRAK	52	AV_{DD}
21	PD.6,BPWM1_CH1,CAN0_RXD	53	PC.7,PWM0_BRAKE1,I2C0_SCL,UART4_RXD
22	PD.7,BPWM1_CH0,CAN0_TXD	54	PC.6,PWM0_BRAKE0,I2C0_SDA,UART4_TXD
23	PD.14,BPWM0_CH5,UART2_RXD	55	PC.15
24	PD.15,BPWM0_CH4,UART2_TXD	56	PC.14
25	PC.3,BPWM0_CH3,SPI0_MOSI0	57	PB.15, ,BPWM1_CH5TM0,TM0_EXT,INT1
26	PC.2,BPWM0_CH2,SPI0_MISO0	58	PF.0,XT1_OUT
27	PC.1,BPWM0_CH1,SPI0_CLK	59	PF.1,XT1_IN
28	PC.0,BPWM0_CH0,SPI0_SS0	60	nRESET
29	PE.5,PWM0_CH5,TM1_EXT,TM1	61	V_{SS}
30	PB.11,TM3,PWM0_CH4	62	V_{DD}
31	PB.10,TM2	63	PF.8, PWM1_CH4,CLKO
32	PB.9,TM1	64	PB.8,BPWM1_CH2,CLKO,TM0,STADC

Table 3-4 Pin Assignment for NUC131SD2AE

3.3 NuMaker UNO NUC131SD2AE Extended Connectors Layout

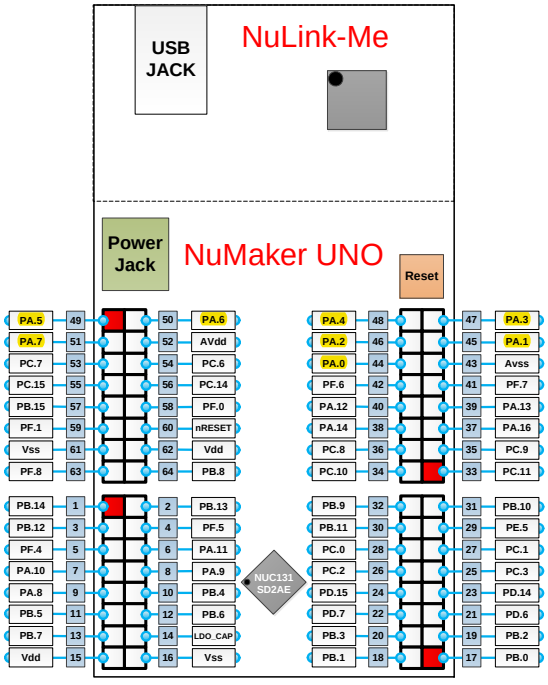


Figure 3-3 NuMaker UNO NUC131SD2AE Extended Connectors Layout

3.4 Arduino UNO Pin Assignment for NuMaker UNO

The NuMaker UNO provides the NUC131SD2AE target chip on board and the Arduino UNO extended connectors (**NU1**, **NU2**, **NU3, NU4** and **NU5**) for LQFP64-pin. Table 3-5 shows the pin assignment for NuMaker UNO.

	Pin No	Pin Name	Pin No	Pin Name
Clock Output	1	CLKO	5	5V
Power	2	VCC	6	GND
	3	RESET	7	GND
	4	3.3V	8	Vin
Analog	1	A0	4	A3
	2	A1	5	A4
	3	A2	6	A5
Digital	0	PB.0/UART_RX0	9	PA.14/PWM0_CH2
	1	PB.1/UART_TX0	10	PA.13/PWM0_CH1
	2	PF.4/PWM1_CH4	11	PA.12/PWM0_CH0
	3	PF.5/PWM1_CH5	12	PB.15/TM0/INT1
	4	PA.10/PWM1_CH2	13	PB.13(LED)
	5	PA.11/PWM1_CH3	VSS	VSS
	6	PA.15/PWM0_CH3	VREF	VREF
	7	PE.5/PWM0_CH5	I2C	PA.8/SDA
	8	PB.11/PWM0_CH4	I2C	PA.9/SCL
SPI Interface	1	PC.2/SPI0_MISO	5	RESET
	2	V_{DD}	6	GND
	3	PC.1/SPI0_CLK	7	PC.0/SPI0_SS
	4	PC.3/SPI0_MOSI	8	PB.9/TM1

Table 3-5 Pin Assignment for NuMaker UNO

3.5 NuMaker UNO Arduino Layout

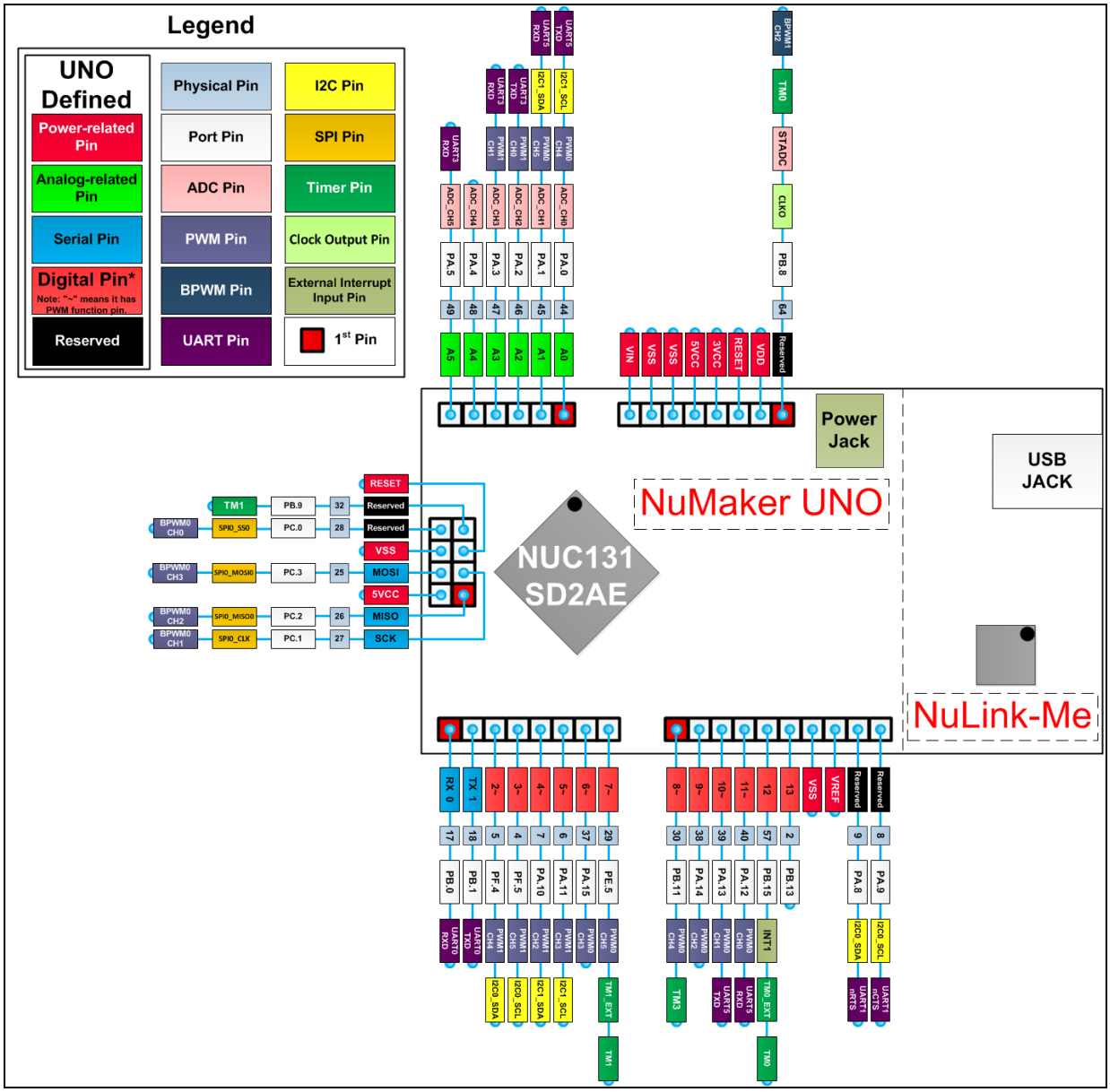


Figure 3-4 NuMaker UNO Pin Design for Arduino

3.6 NuMaker UNO PCB Placement

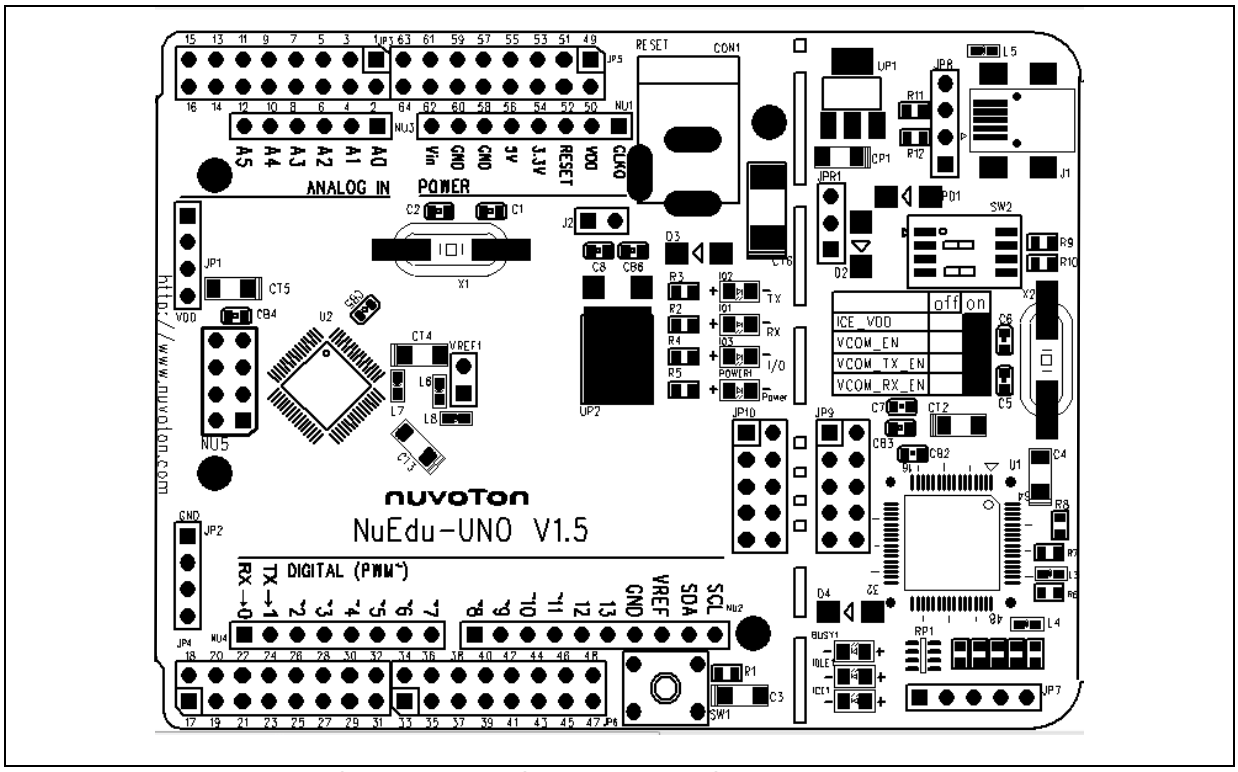


Figure 3-5 NuMaker UNO PCB Placement

4 DOWNLOADING AND INSTALLING ARDUINO IDE 1.8.5 AND NUMAKER UNO SOFTWARE

4.1 Downloading and Installing Arduino IDE 1.8.5 Software

Please visit the Arduino official website (http://www.arduino.cc/en/Main/software) to download and install the Arduino IDE 1.8.5. Currently it is recommended to install Arduino IDE 1.8.5 for Windows version, since the other operating system has not been verified.

The Arduino IDE 1.8.5 installation steps are as follows:

4.1.1 Step 1: Download Arduino IDE 1.8.5 software

Visit https://www.arduino.cc/en/Main/Software, and download the latest Arduino.

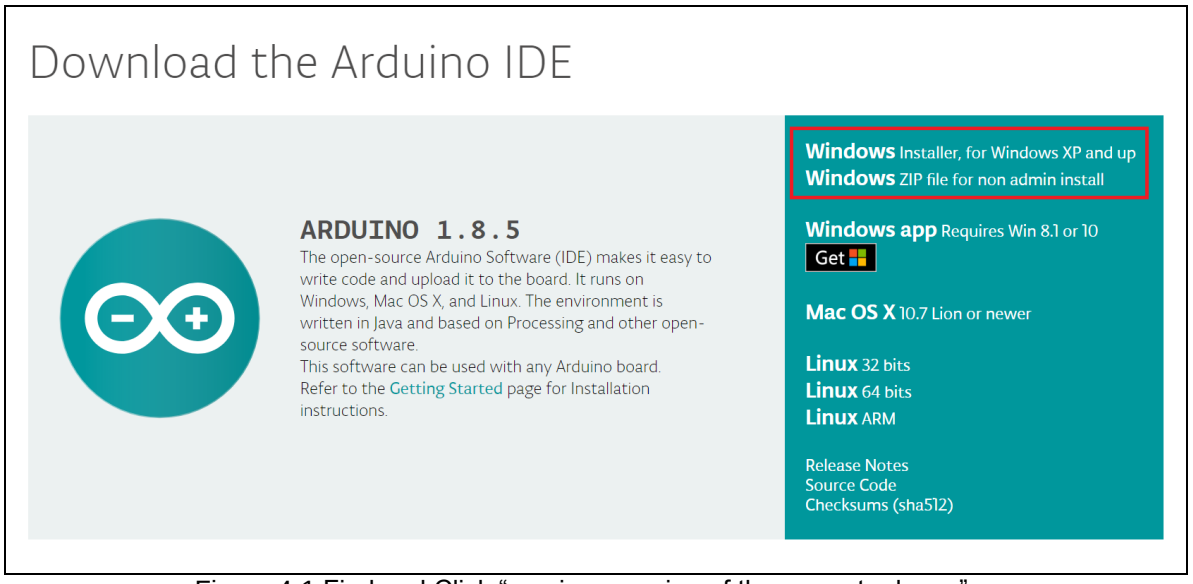


Figure 4-1 Find and Click “previous version of the current release”

4.1.2 Step 2: Check if download file is version 1.8.5

Click on download.

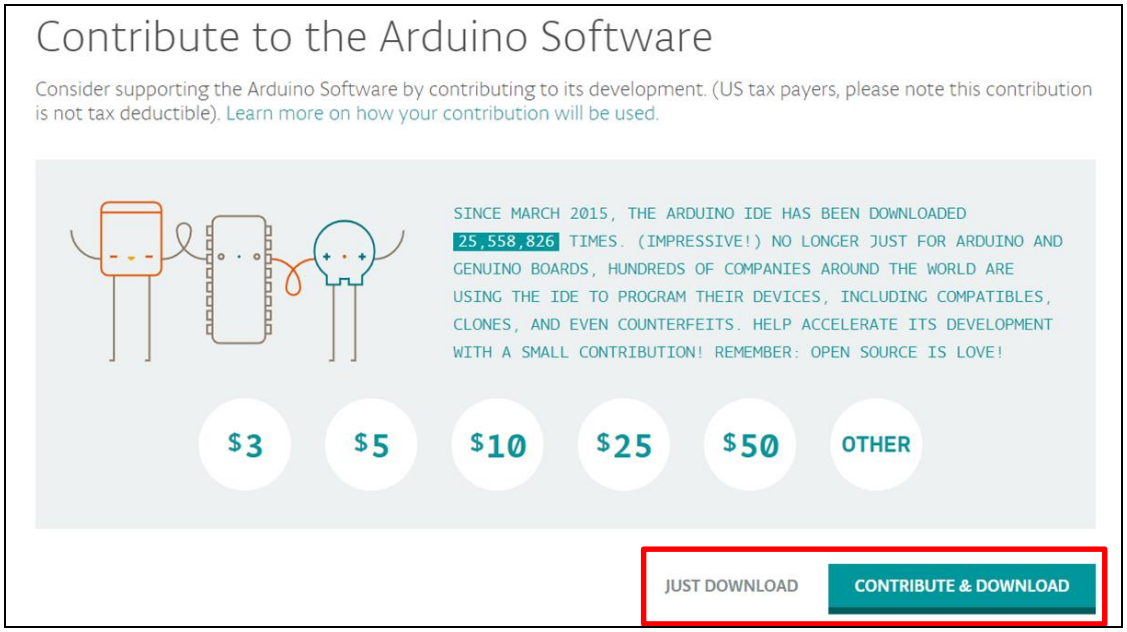


Figure 4-2 Find and Download Arduino 1.8.5

4.1.3 Step 4: Install the "Arduino-1.8.5-windows.exe" file

Double click on the "arduino-1.8.5-windows.exe" and click "Run".

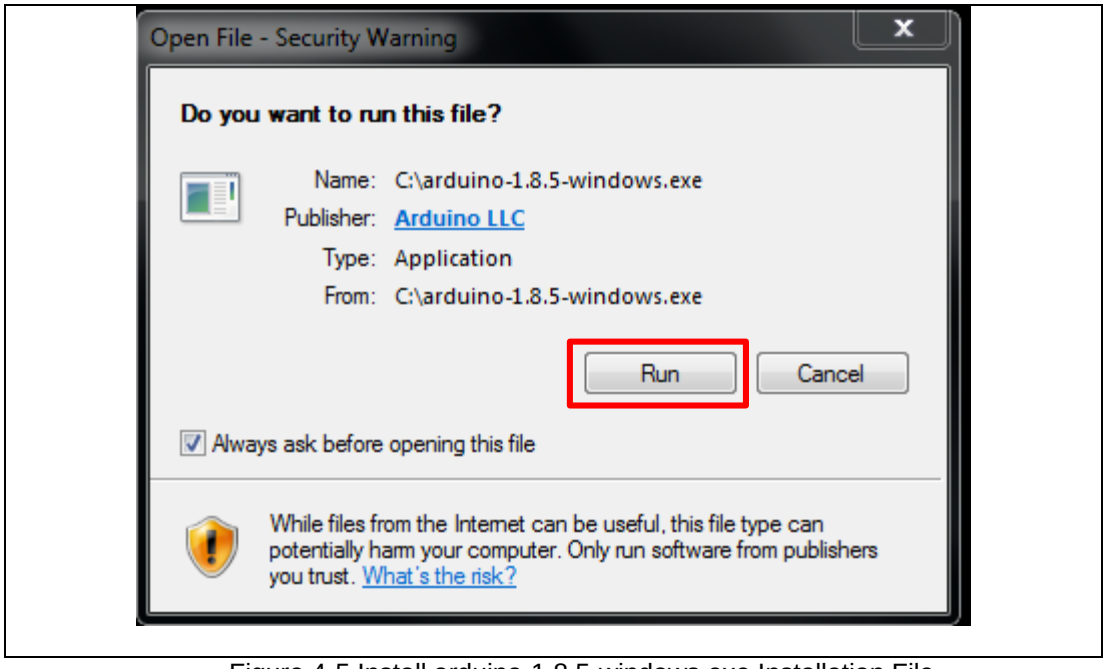


Figure 4-5 Install arduino-1.8.5-windows.exe Installation File

4.1.4 Step 5: Select the installation folder

Select the installation folder and click “Install”.

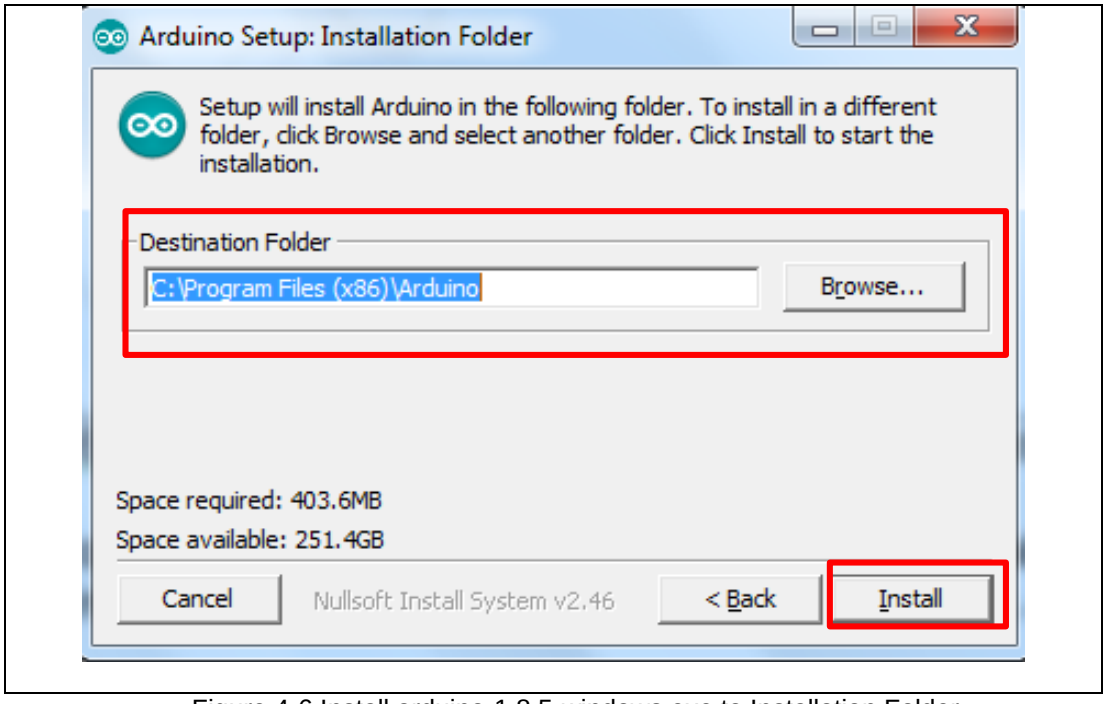


Figure 4-6 Install arduino-1.8.5-windows.exe to Installation Folder

4.1.5 Step 6: Wait for the installation is complete

Wait the installation process until it is finished. It will take about 2 minutes to install arduino-1.8.5-windows file to the installation folder.

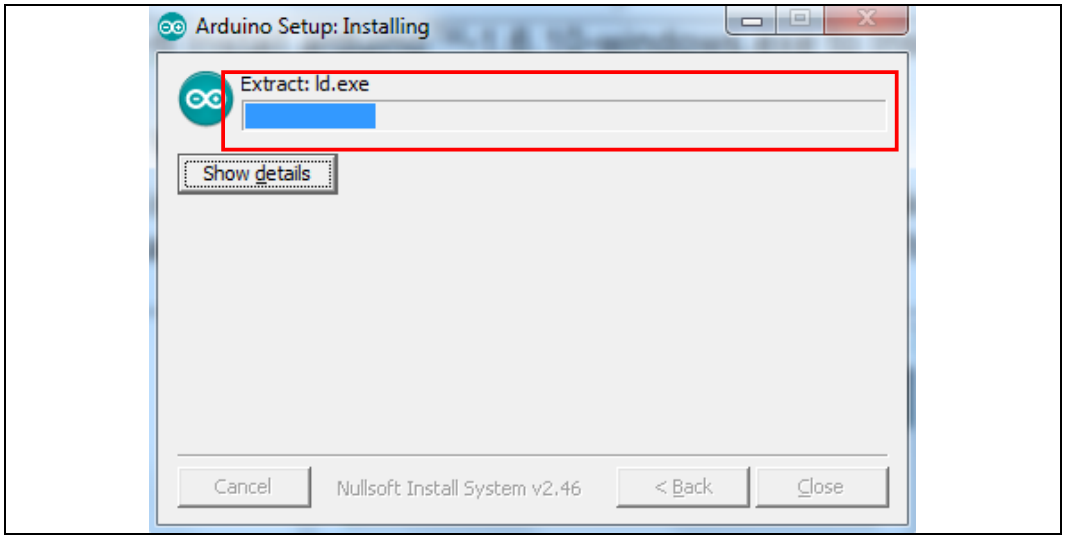


Figure 4-7 Install arduino-1.8.5-windows to Installation Folder

After installation is finished, two executable files can be found in the Arduino 1.8.5 installation directory, including arduino.exe and arduino_debug.exe. Both files are able to start the Arduino

IDE program. As for the arduino_debug.exe, you can open the debug window. When the program is executed or compiled, the debug window will open for users to check which part of the problem occurs.

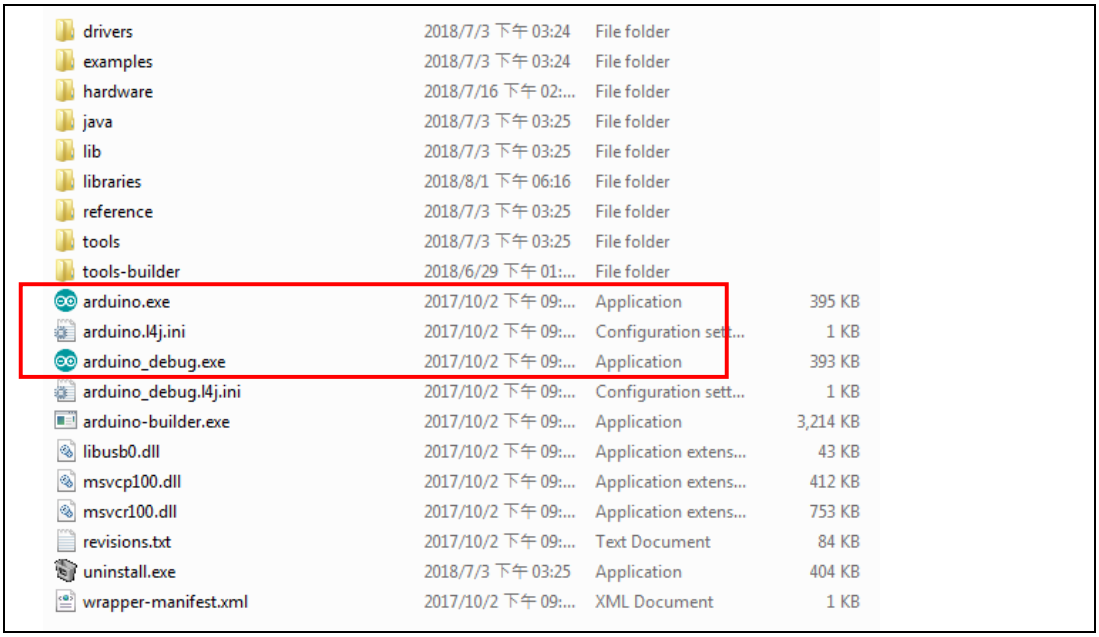


Figure 4-8 Arduino 1.8.5 Installation Directory

4.2 Installing Nu-Link USB Driver for Arduino IDE 1.8.5

Please visit the Nuvoton NuMaker UNO official website (www.nuvoton.com/NuMaker_UNO) to download "Nu-Link USB Driver".

Note: It is recommended to use Arduino IDE version 1.8.5.

The Nu-Link USB Driver installation steps are as follows:

4.2.1 Step 1: Download Nu-Link USB Driver

Visit www.nuvoton.com/NuMaker_UNO and find Resources. Click on "Nu-Link USB Driver".

Figure 4-9 Find and Click Nu-Link USB Driver

4.2.2 Step 2: Select the installation folder

Select the installation folder and click “Next” to install.

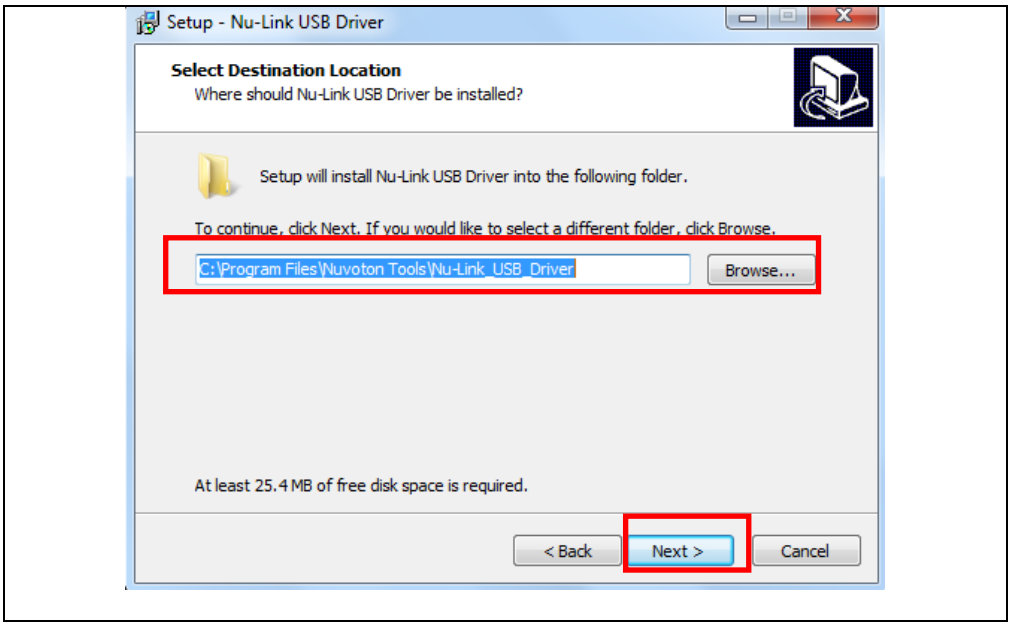


Figure 4-10 Installing Nu-Link USB Driver

4.2.3 Step 3: Run Arduino IDE 1.8.5

Go to **File** → **Preferences**, enter the following URL to textbox of ‘Additional Board Manager URLs’
https://raw.githubusercontent.com/OpenNuvoton/NuMaker-UNO/master/package_nuvoton_index.json

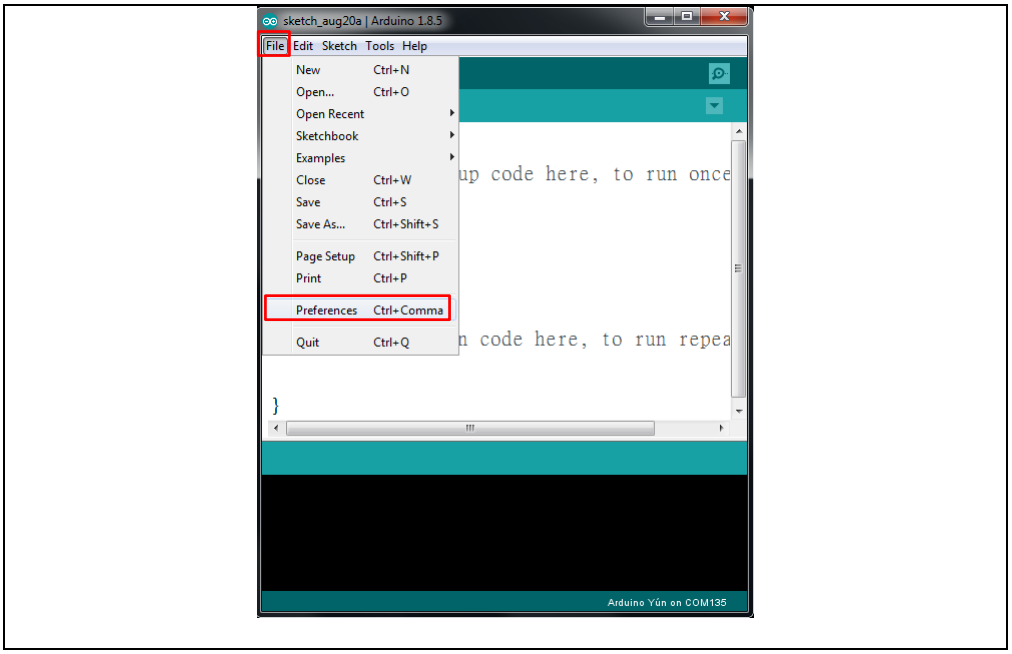


Figure 4-11 Arduino IDE 1.8.5 Select File and Click Preferences

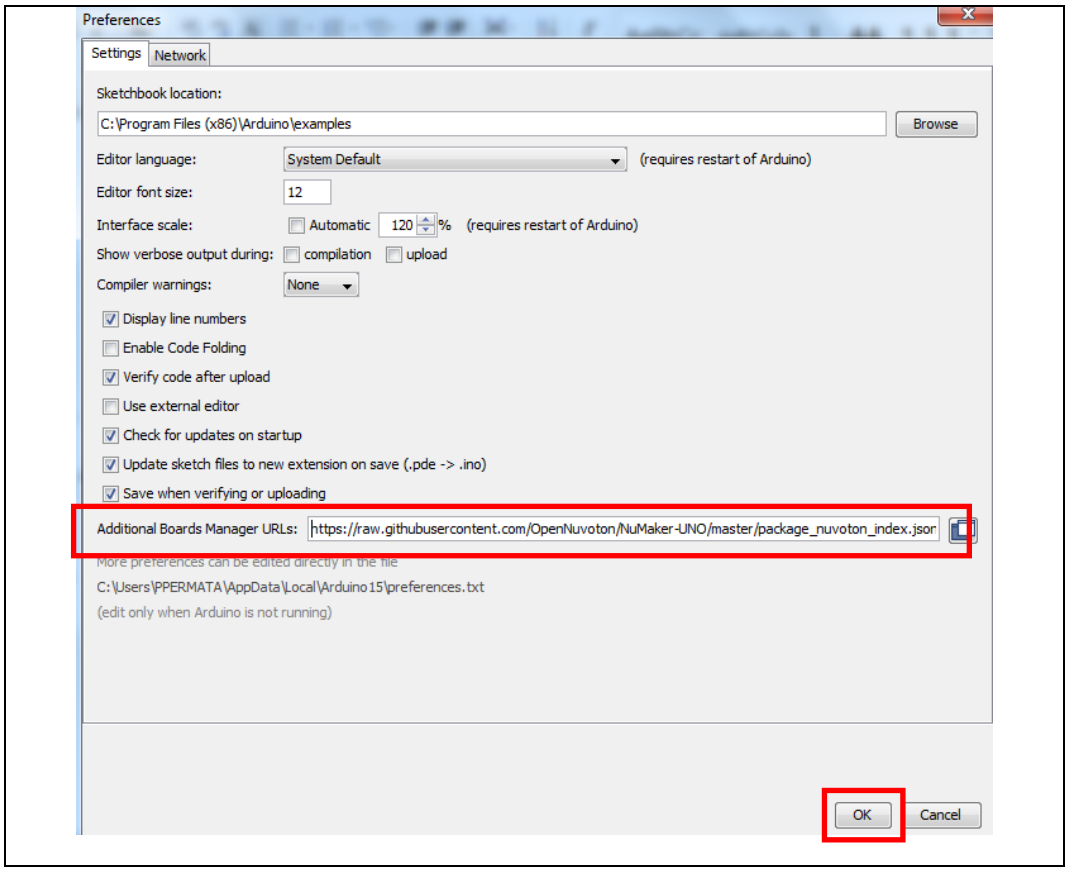


Figure 4-12 Enter the NuMaker UNO Board's URL to 'Additional Board Manager URLs'

4.2.4 Step 4: Install NuMaker UNO on Arduino IDE 1.8.5 Boards Manager

Go to **Tools → Board → Boards Manager** on Arduino IDE 1.8.5.

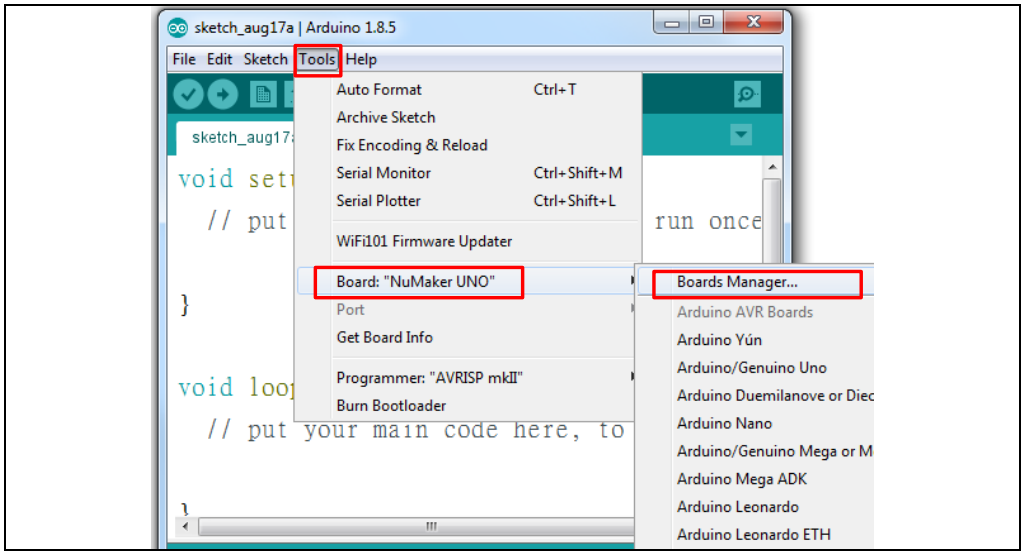


Figure 4-13 Click Boards Manager on Arduino IDE 1.8.5

Search “MuMaker” and the NuMaker UNO will show up on the list. Select NuMaker UNO and click install.

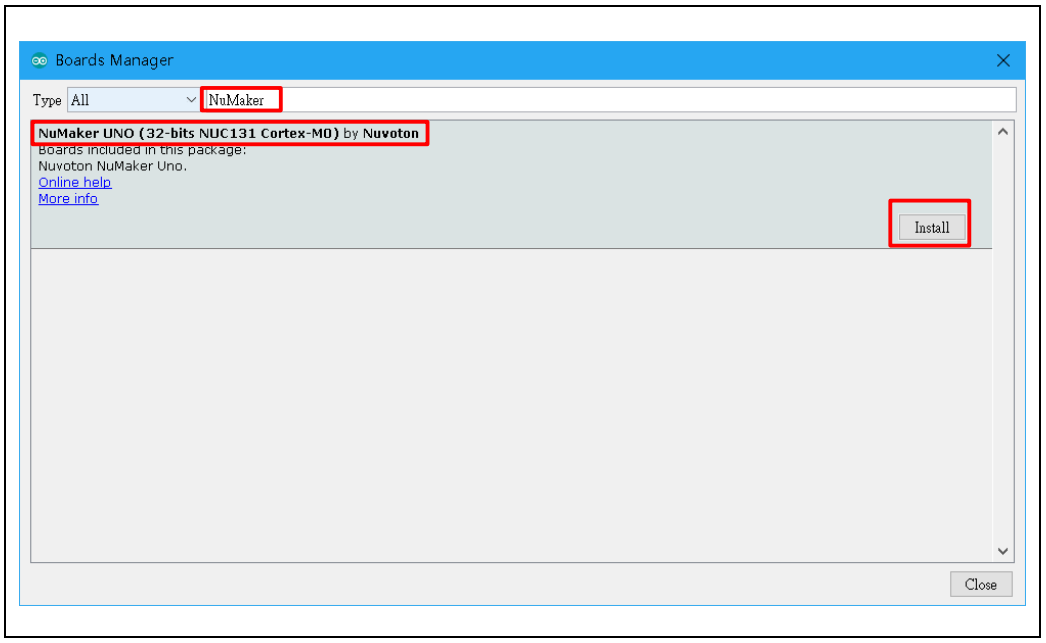


Figure 4-14 Select NuMaker UNO and Click Install

4.2.5 Step 5: Select NuMaker UNO on Arduino IDE 1.8.5 Board Selection

After the package download is finished, go to **Tools** → **Board** and select NuMaker UNO to use it.

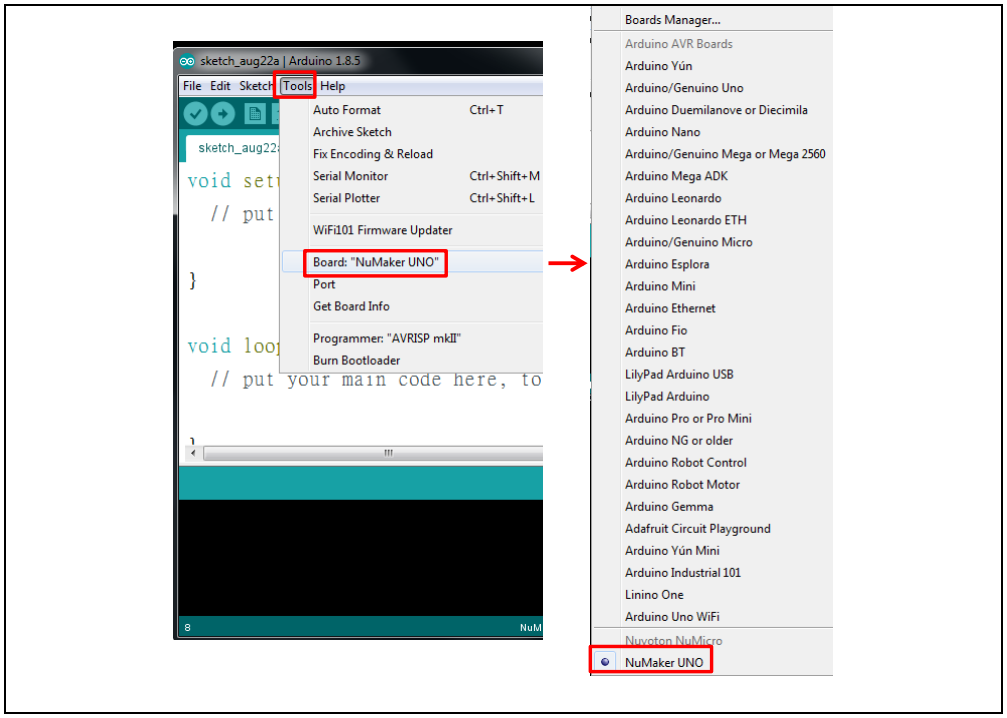


Figure 4-15 Select NuMaker UNO on Board Selection to Use It

4.3 Hardware Setup

Hardware connections are shown in the figure below. Use a mini USB to connect to a computer.

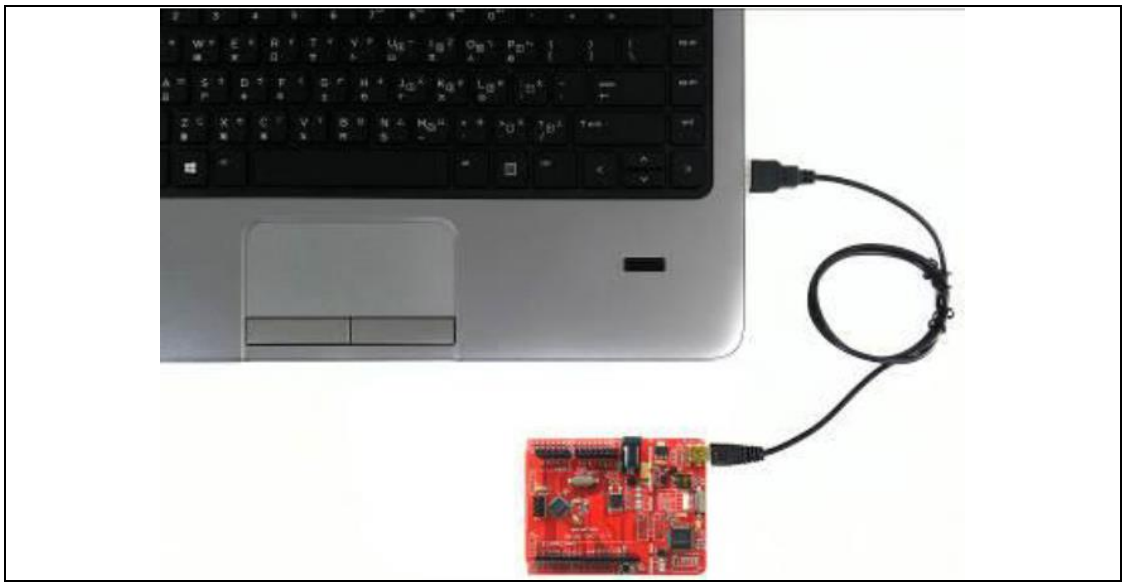


Figure 4-16 NuMaker UNO Connected to Computer

4.4 Testing USB and VCOM Device in the Device Manager

Before connecting the NuMaker UNO development board to the computer, please enable SW2 of VCOM Function. All the SW2 pins shall be turned on. Please refer to section 3.1.2. Open the device manager and check if the USB is detected. If it is not detected, please reinstall the Nu-Link USB Driver.

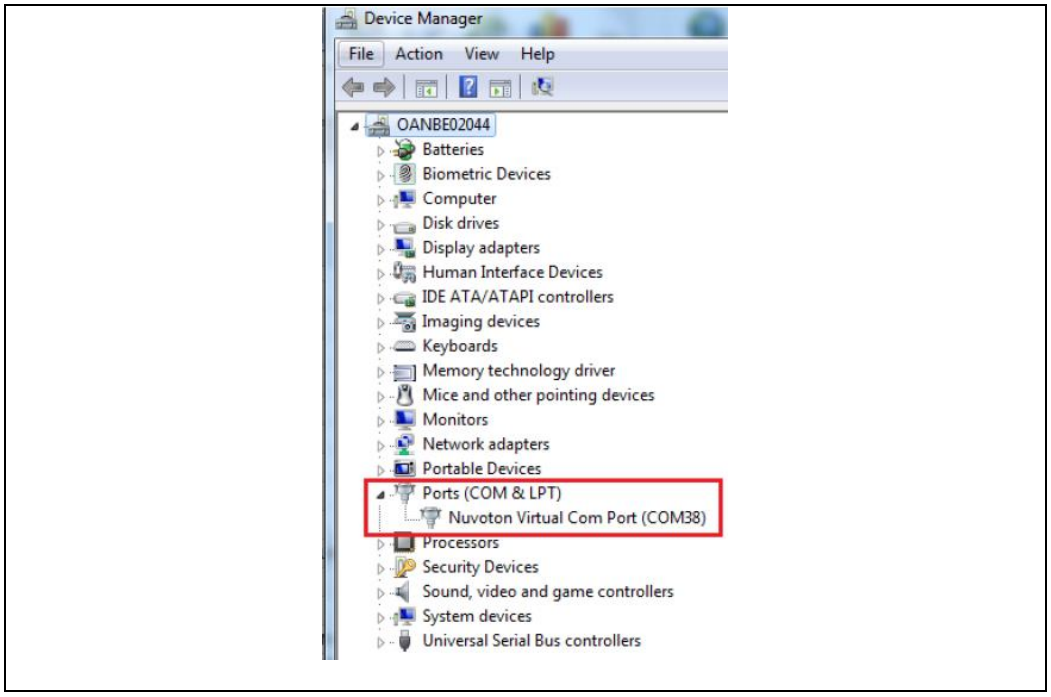


Figure 4-17 USB Device Detected by Device Manager

5 STARTING TO USE NUMAKER UNO ON ARDUINO IDE 1.8.5

5.1 Compiling and Executing Example Program

This example demonstrates how to download an application and debug using virtual serial port on a NuMaker UNO board. When users install the Nu-Link USB Driver, the “NuMaker UNO” can be found in Arduino IDE on **Tools → Board → NuMaker UNO** as shown in the figure below.

Note: Execute the “arduino_debug.exe” file in the installation path.

5.1.1 Step 1: Select the NuMaker UNO Board

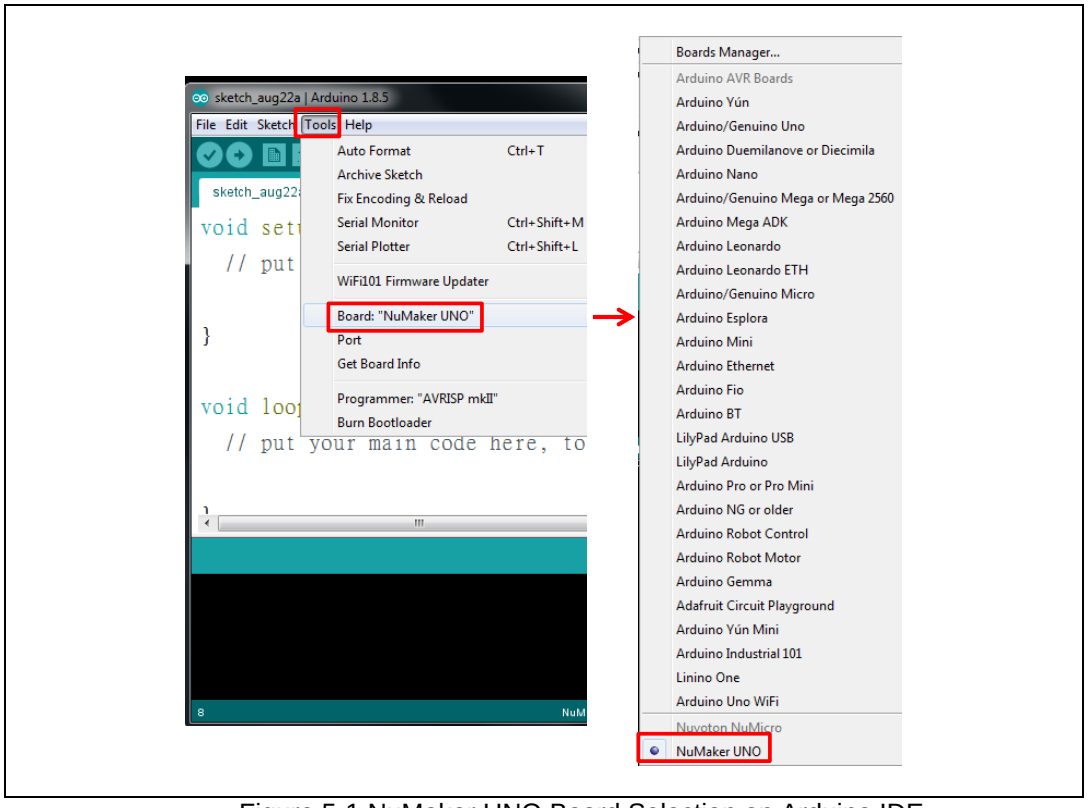


Figure 5-1 NuMaker UNO Board Selection on Arduino IDE

5.1.2 Step 2: Open virtual serial modes and select Debug Serial Port

SW2: Supply power to NuMaker UNO. Before NuMaker UNO is connected to PC, all the SW2 pins shall be turned on. To activate UART0 function, SW2 pin2 ~ pin4 shall be turned off.

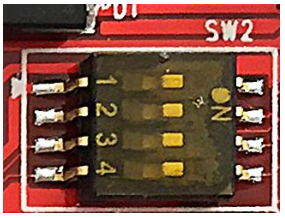
Switch Pin Number	Function Name	UART0 Mode	VCOM Mode	
1	ICE_VCC	On	On	
2	VCOM_En	Off	On	
3	VCOM_TX	Off	On	
4	VCOM_RX	Off	On	

Table 5-1 SW2 Mode Setting (Default as VCOM Mode)

After SW2 setting is completed, you will be able to see the serial COM port number.

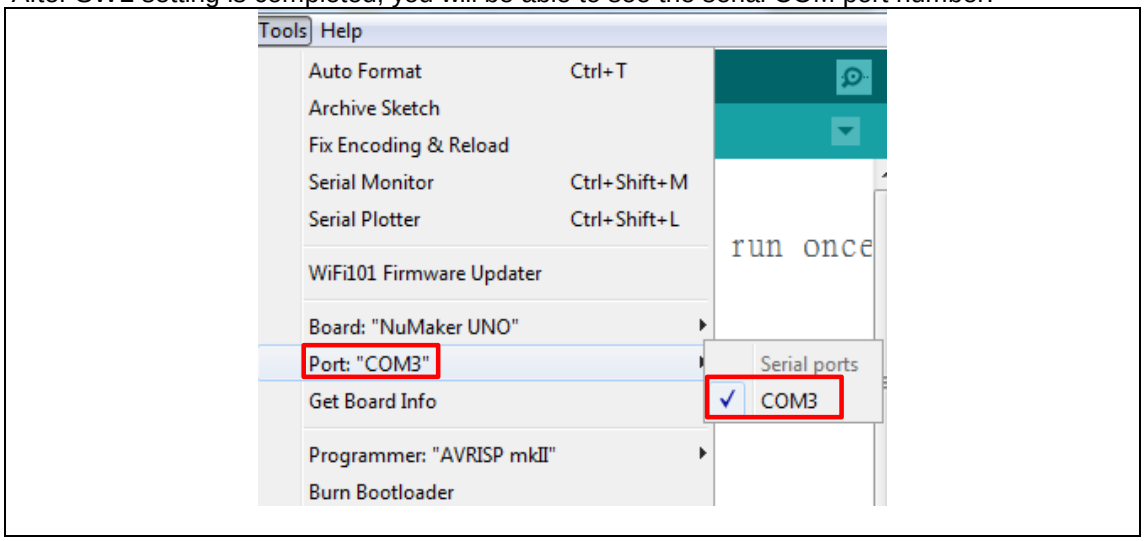


Figure 5-2 Select NuMaker-UNO Serial COM Port

Note 1: If the SW2 pin 2 of VCOM function is not enabled, **Tools → Port** will become gray (disabled), unless the board is re-connected to the computer. However, before re-connecting to PC, you must open the VCOM function.

Note 2: If the serial port number is not selected, you will not be able to use the serial monitor.

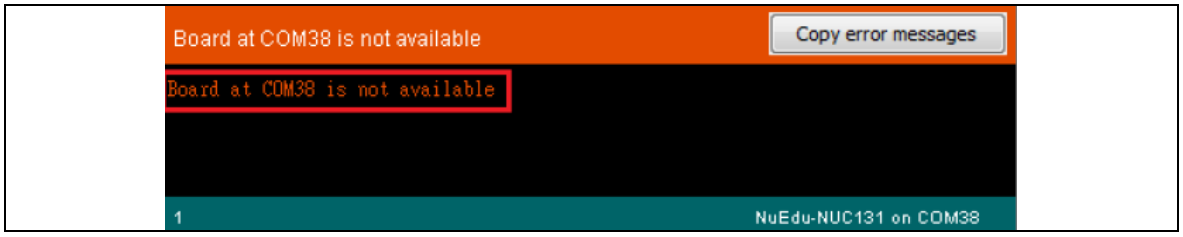


Figure 5-3 COM port Not Detected and Unable to Use Serial Monitor

5.1.3 Step 3: Open the sample code

Open the sample code through **File → Examples → 03.Analog → AnalogInOutSerial**.

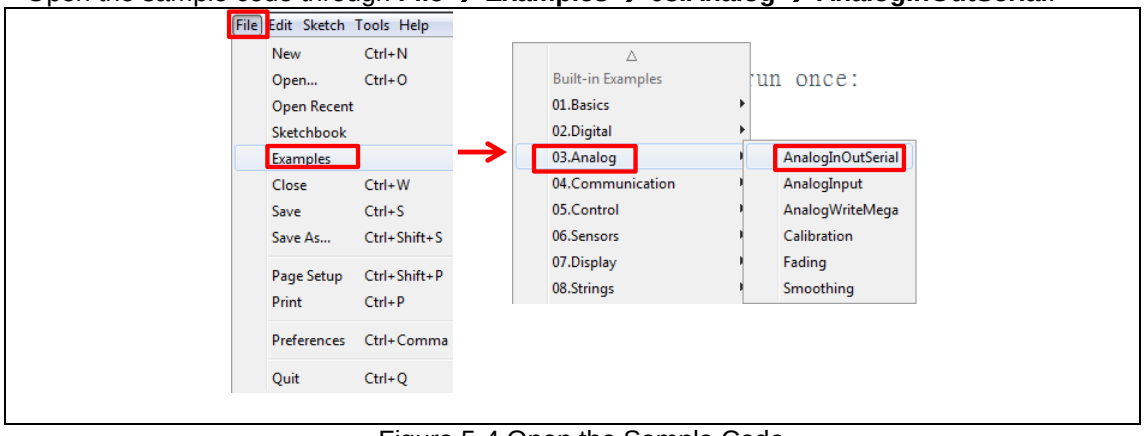


Figure 5-4 Open the Sample Code

5.1.4 Step 4: Download the sample code

Use the Upload button to compile and load code to target board or use the Verify button to compile code.

1.  Upload: This button can compile and load code to a target board.
2.  Verify: This button can compile code.
3.  Serial Monitor: This button can open the serial monitor.

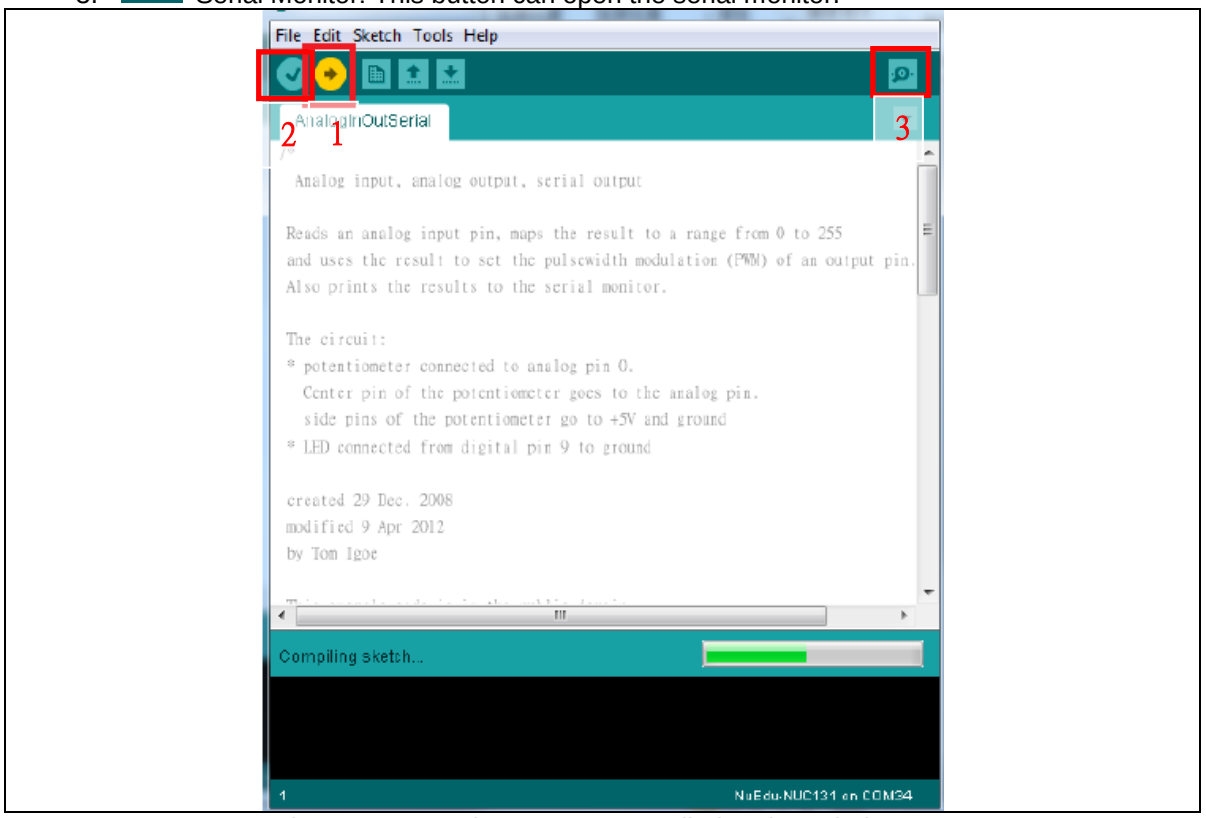


Figure 5-5 Sample Program Compiled and Loaded

5.1.5 Step 5: Check the Correctness of the VCOM Baud Rate

User needs to check if the virtual serial port baud rate setting is the same as the program setting. When the program is set to 9600, the virtual serial port monitor will be set as 9600.

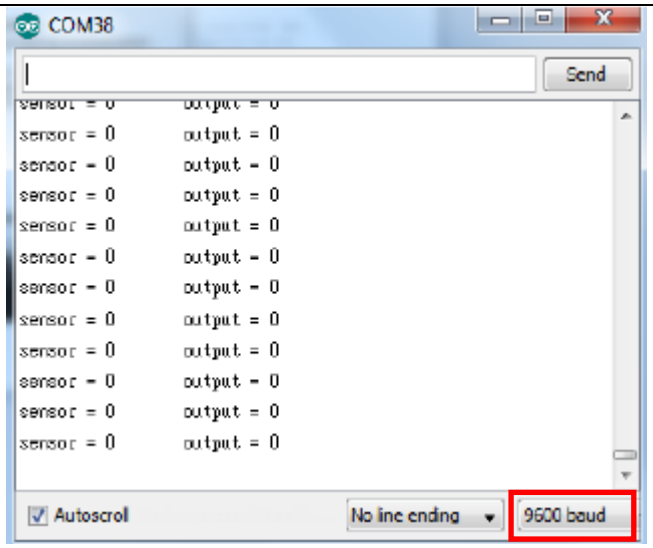


Figure 5-6 Serial Monitor

This example uses ADC0 to perform ADC conversion into a digital value which is then displayed by the virtual serial port out. To set a COM port baud rate and then display the virtual serial conversion results, use the ADC0 pin connected to V_{DD} or V_{SS}. Then, whether the ADC conversion result is correct can be displayed.

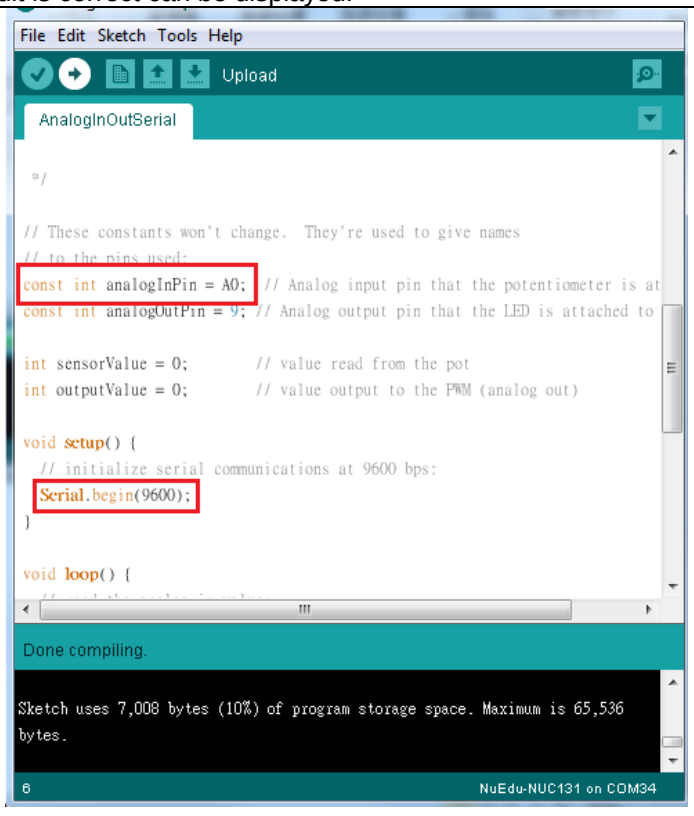


Figure 5-7 Sample Program Baud Rate Setting

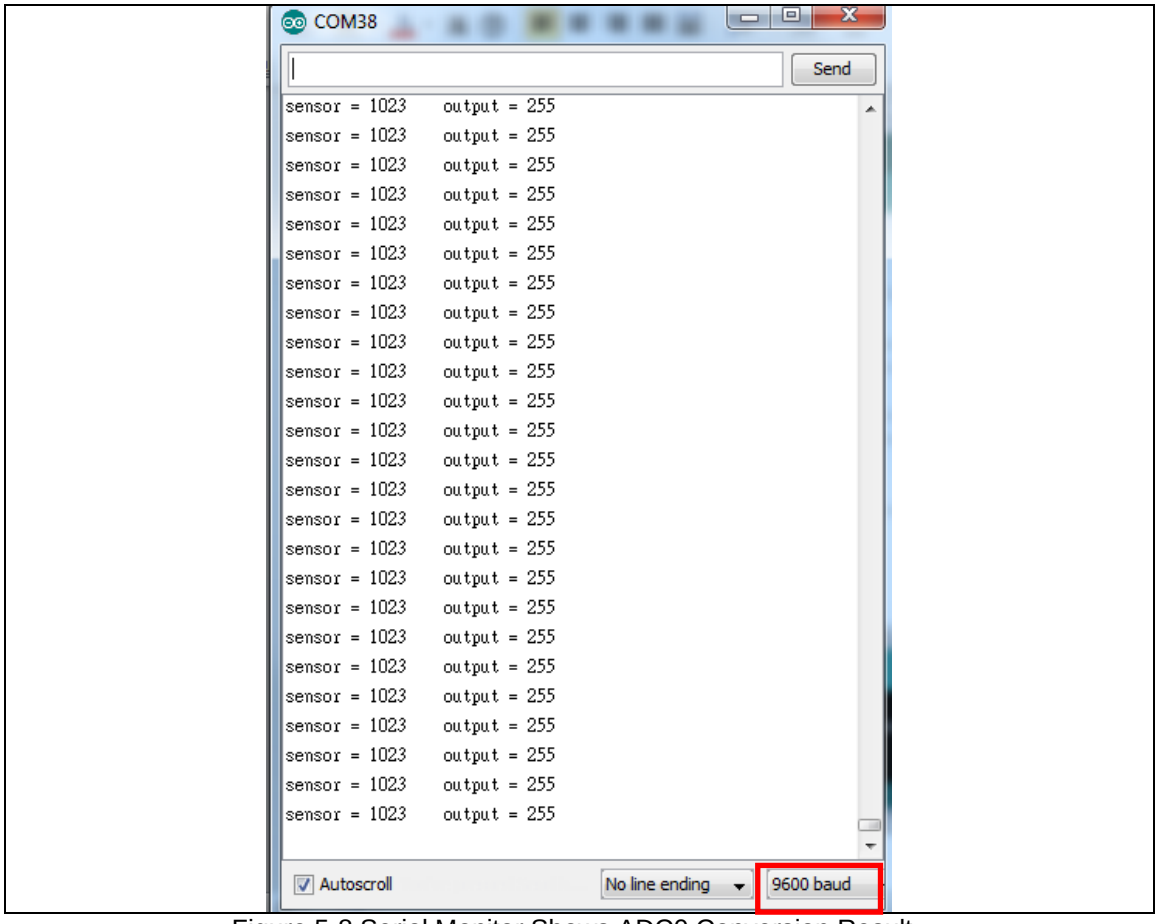


Figure 5-8 Serial Monitor Shows ADC0 Conversion Result

6 STARTING TO USE NUMAKER UNO ON KEIL µVISION® IDE

6.1 Downloading and Installing Keil µVision® IDE Software

Please visit the Keil official website (http://www.keil.com) to download the Keil µVision® IDE and install the RVMDK.

6.2 Downloading and Installing Nu-Link Keil Driver

Please visit Nuvoton NuMicro® official website (http://www.nuvoton.com/NuMicro) to download the "*NuMicro Nu-Link Keil Drive*r" file. Please refer to section 8.1 for the detailed download flow. After the Nu-Link driver is downloaded, please unzip the file and execute the "Nu-Link_Keil_Driver.exe" to install the driver.

6.3 Hardware Setup

The hardware setup is shown in the figure 6-1. If users want to use the VCOM function, turn on all the SW2 pins (refer to section 3.1.2).

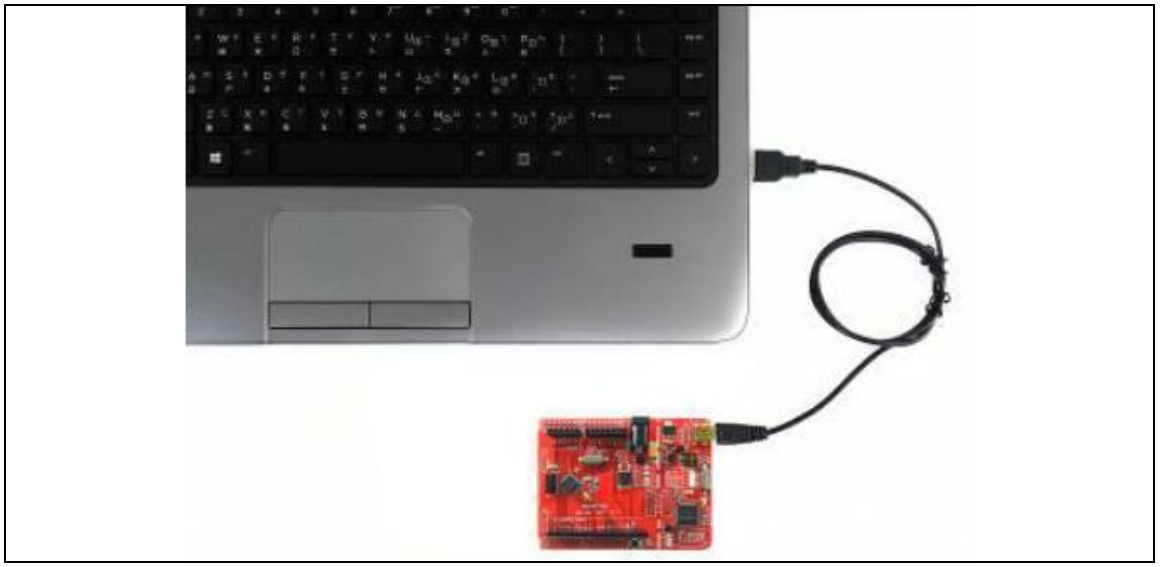


Figure 6-1 NuMaker UNO Connected to the Computer

6.4 Procedure for Downloading and Debugging Example Program

This example demonstrates how to download and debug a program on the NuMaker UNO board. The example file can be found in the directory list as shown in the following figure.

6.4.1 Step 1: Open the Project

Please open the following path.
"C:\Nuvoton\BSPLibrary\NUC131BSP_CMSIS_v3.00.001\SampleCode\StdDriver\ADC_Result Monitor\KEIL"

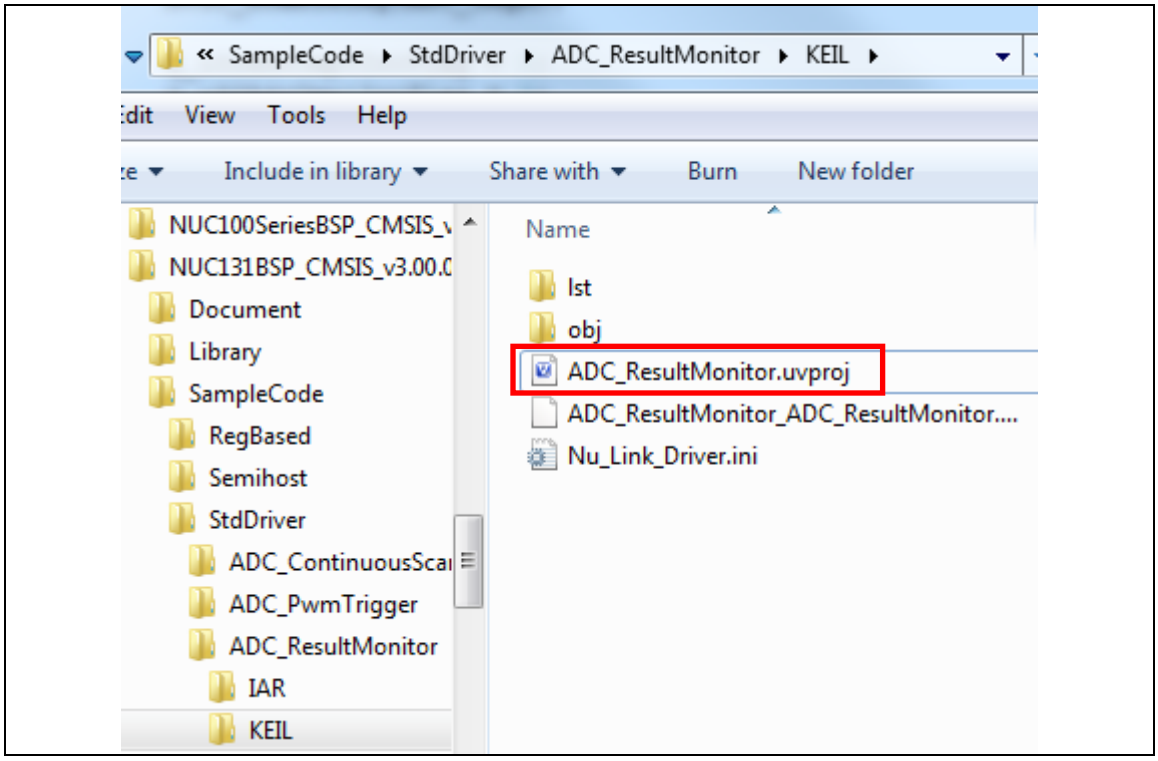


Figure 6-2 ADC Sample Program

6.4.2 Step 2: Check Device Chip and Debug Chip

Please open the “target options” to check the device chip and if the debug chip selection is correct. Figure 6-3 shows the correct device chip selection. Figure 6-4 shows the correct debug chip selection. Figure 6-5 shows the debugging tool to confirm the selection is correct.

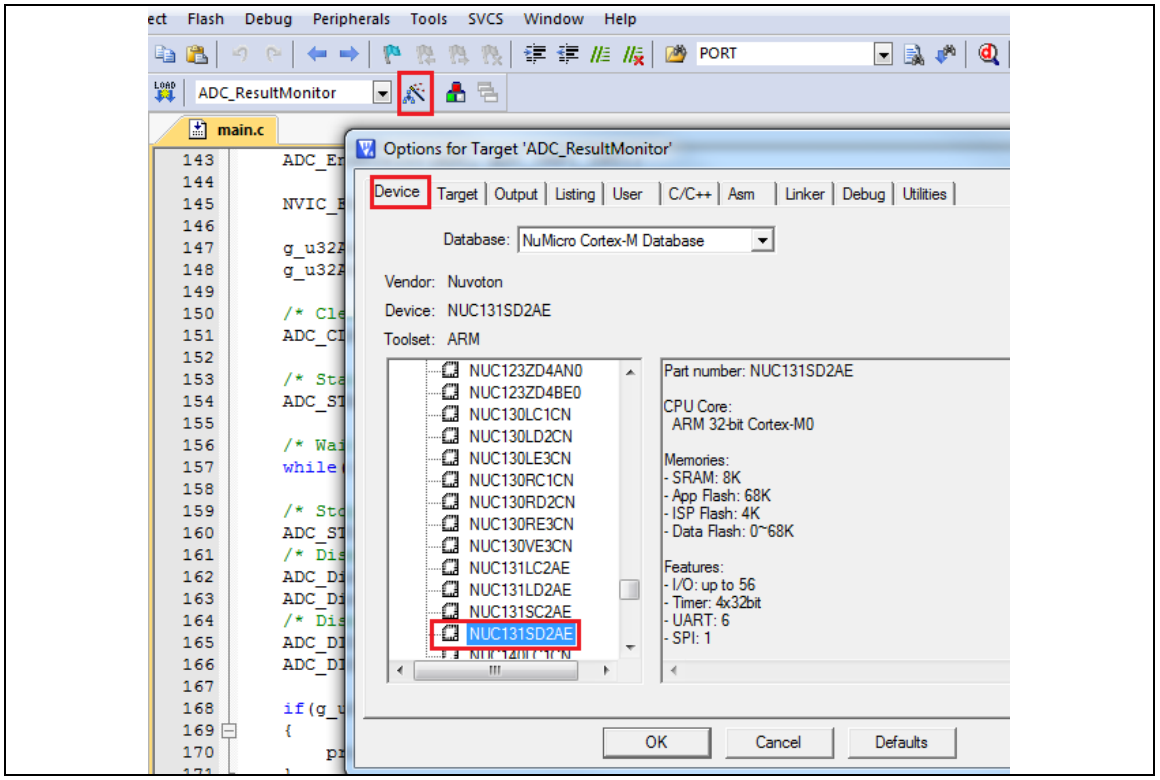


Figure 6-3 Select Device Chip

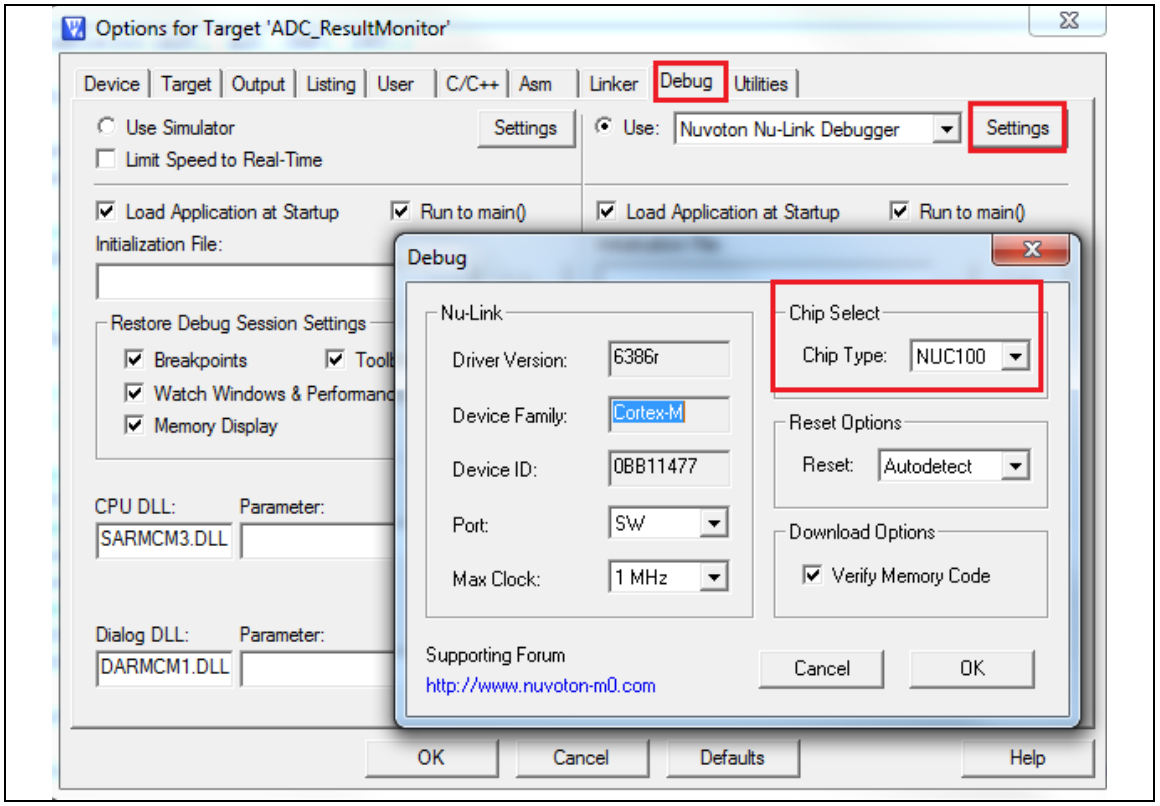


Figure 6-4 Select Debug Chip

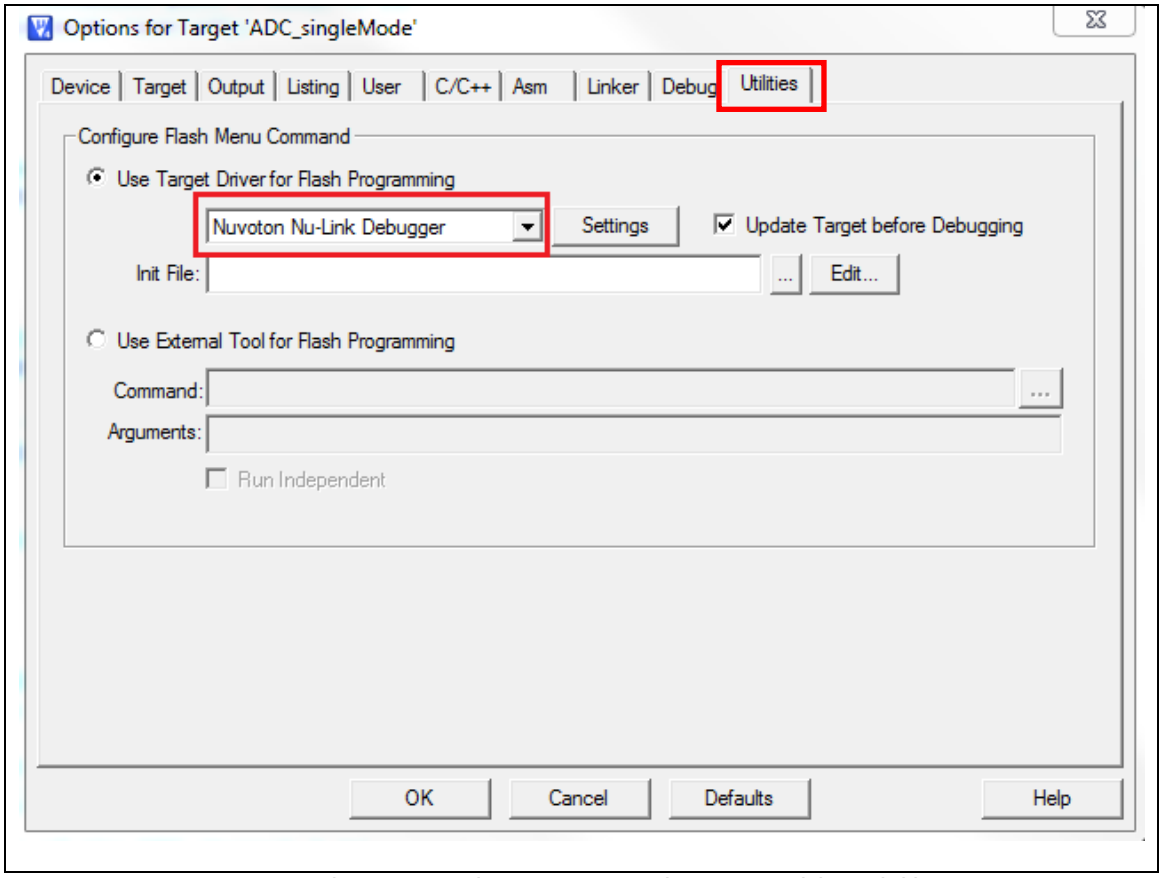


Figure 6-5 Select Nuvoton Debugger Tool (Nu-Link)

6.4.3 Step 3: Build and Download Sample Code

Click the Build button to see the completed compilation for error if any, and finally load the code to the development board.

Build

Download

6.4.4 Step 4: Open the Serial Monitor and Set Baud Rate

User can open the serial monitor to print debug message. The present example uses the "PuTTY tool".

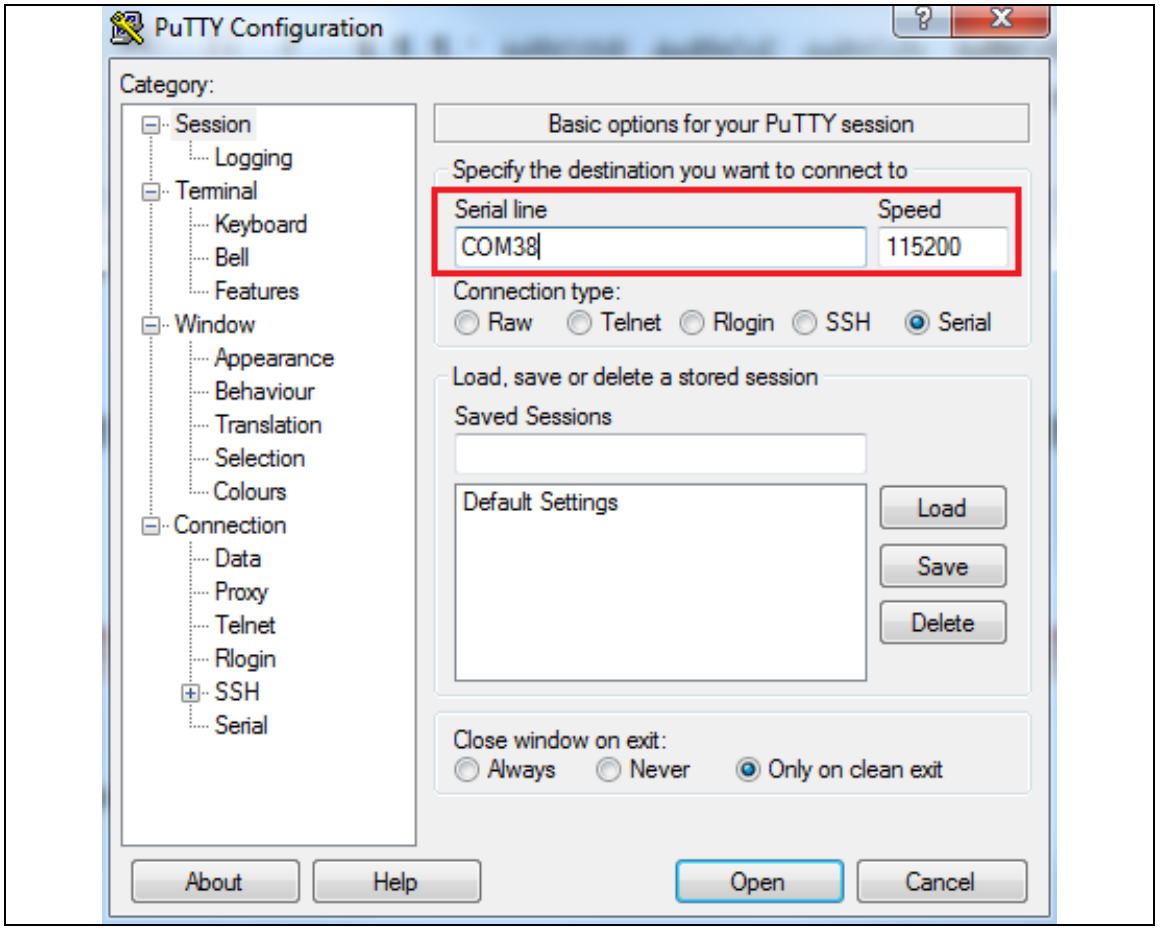


Figure 6-6 Set the Baud Rate on PuTTY Tool

6.4.5 Step 5: Click the Reset Button to Run Your Code.

After clicking the Reset button, chip will re-execute the program. The debug messages are displayed as shown below

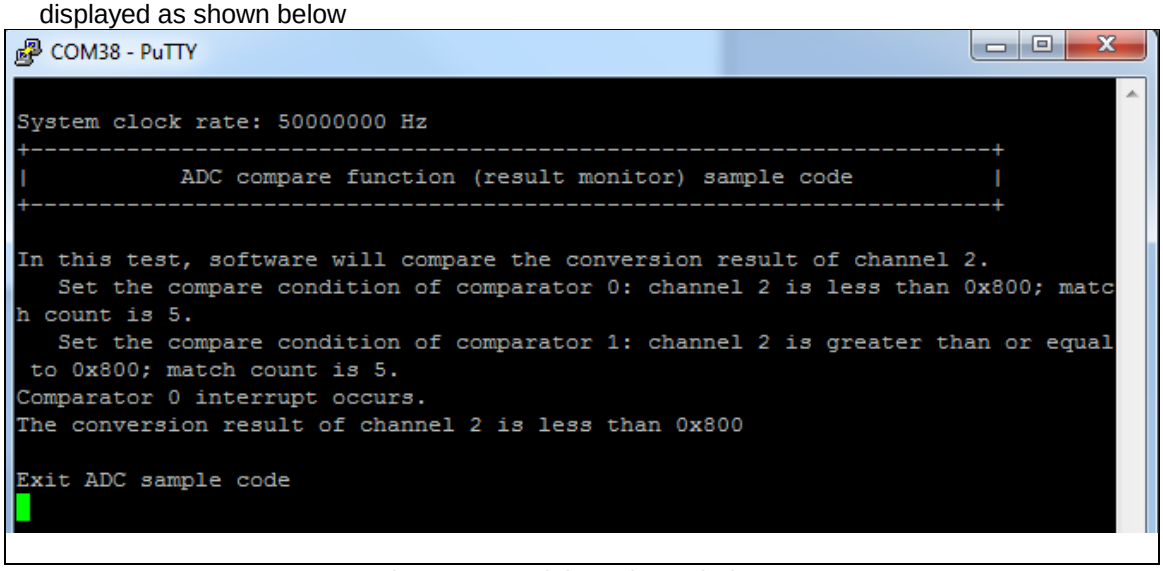


Figure 6-7 Serial Monitor Window

Function Button Description:

1. **Open µVision®** Development Tool
   Project – Open
   - Open the SYS.uvproj project file (Step 1 in Figure 6-8)

2. **Project – Build**
   - Compile and link the SYS application (Step 2 in Figure 6-9)

3. **Flash – Download**
   - Program the application code into on-chip Flash ROM (Step 3 in Figure 6-9)

4. **Open Debug Mode** (Step 4 in Figure 6-9)

   When using the debugger commands, it has the following features:

   a. **Window View** to detect changes in the value of variables and registers (Step 4a in Figure 6-10)

   b. **Single-Step** through code (Step 4b in Figure 6-10)

   c. **Reset** the device (Step 4c in Figure 6-10)

   d. **Run** the application (Step 4d in Figure 6-10)

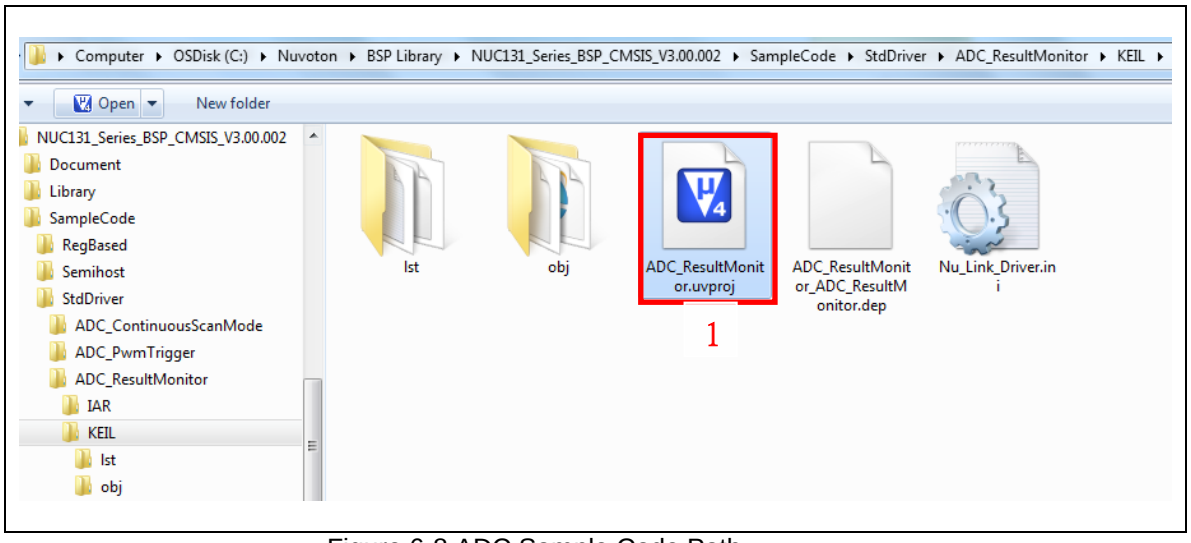


Figure 6-8 ADC Sample Code Path

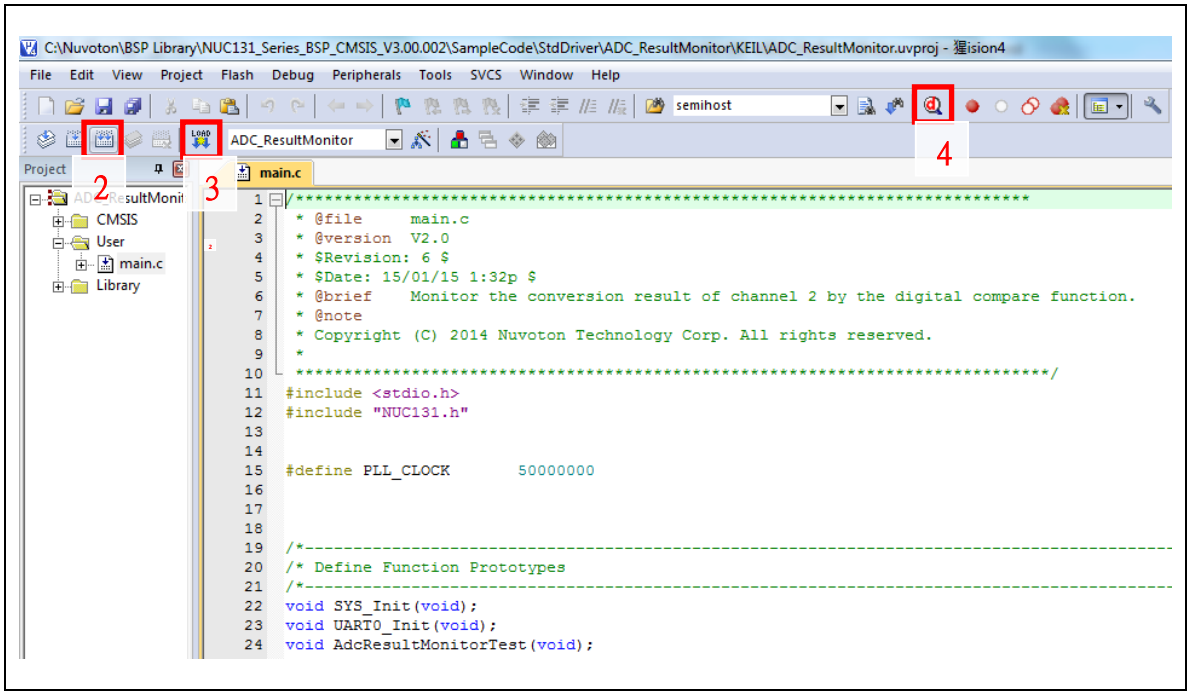


Figure 6-9 ADC Enter Compile Mode Interface

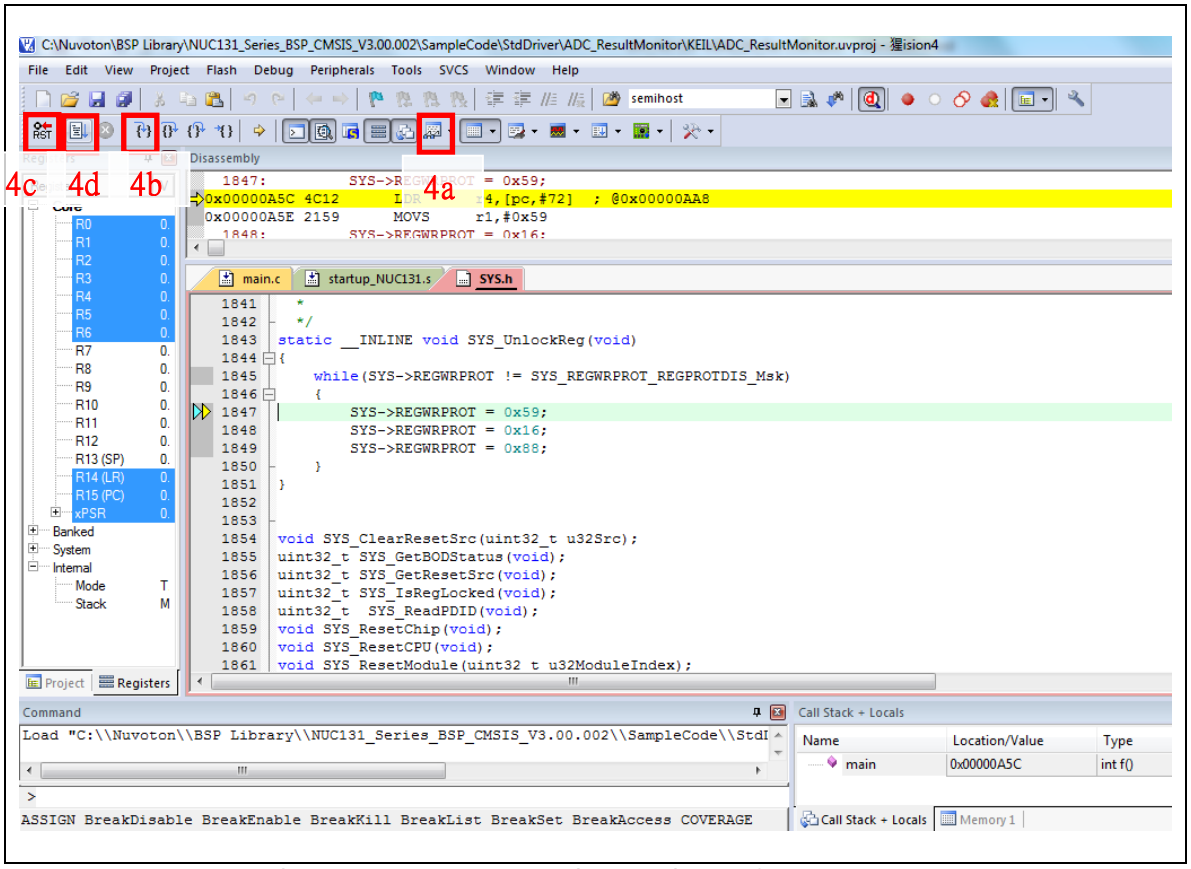


Figure 6-10 ADC Enter Debug Mode Interface

7 STARTING TO USE NUMAKER UNO ON IAR EMBEDDED WORKBENCH

7.1 Downloading and Installing IAR Embedded Workbench Software

Please visit IAR official website (http://www.iar.com) to download the IAR Embedded Workbench and install the EWARM.

7.2 Downloading and Installing Nu-Link IAR Driver

Please visit Nuvoton NuMicro® official website (http://www.nuvoton.com/NuMicro) to download "NuMicro Nu-Link IAR Driver" file. Please refer to section 8.2 for the detailed download flow. After the Nu-Link driver is downloaded, please unzip the file and execute the "Nu-Link_IAR_Driver.exe" to install the driver.

7.3 Hardware Setup

The hardware setup is shown in the figure 7-1. If users want to use the VCOM function, please turn on all the SW2 pins. Please refer to section 3.1.2.

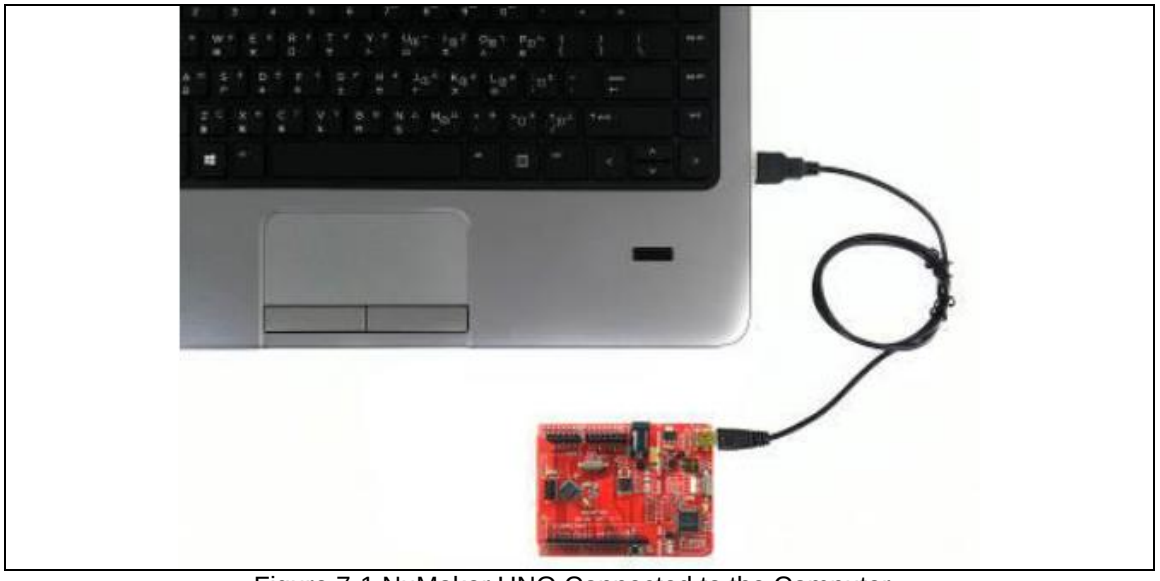


Figure 7-1 NuMaker UNO Connected to the Computer

7.4 Procedure for Downloading and Debugging Example Program

This example demonstrates how to download and debug an application on a NuMaker UNO board. The example file can be found in the directory list as shown in the following figure.

7.4.1 Step 1: Open the Project

Please open the following path example program.
"C:\Nuvoton\BSPLibrary\NUC131BSP_CMSIS_v3.00.001\SampleCode\StdDriver\ADC_Result Monitor\IAR".

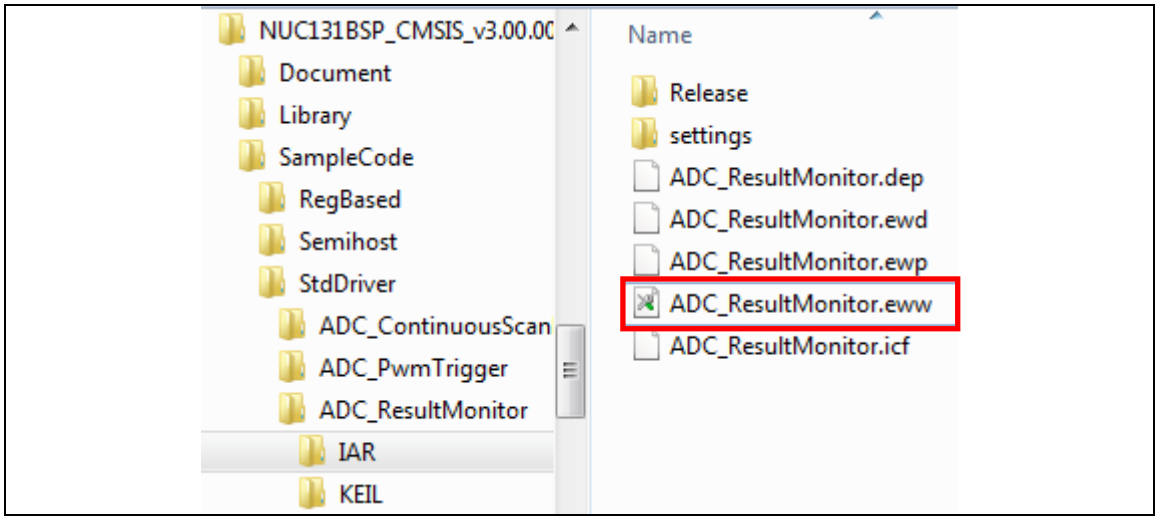


Figure 7-2 ADC Sample Program in IAR

7.4.2 Step 2: Download the Sample Code.

Click the button on the top right corner to download the program.

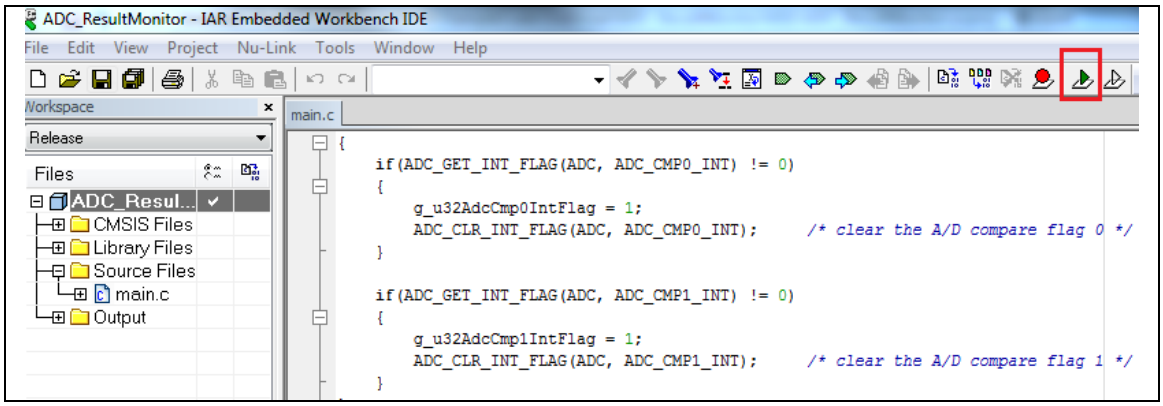


Figure 7-3 Compile and Load Program in IAR

7.4.3 Step 3: Click the Reset Button to Run Your Code

After clicking the Reset button, chip will re-execute the application and debug messages are displayed.

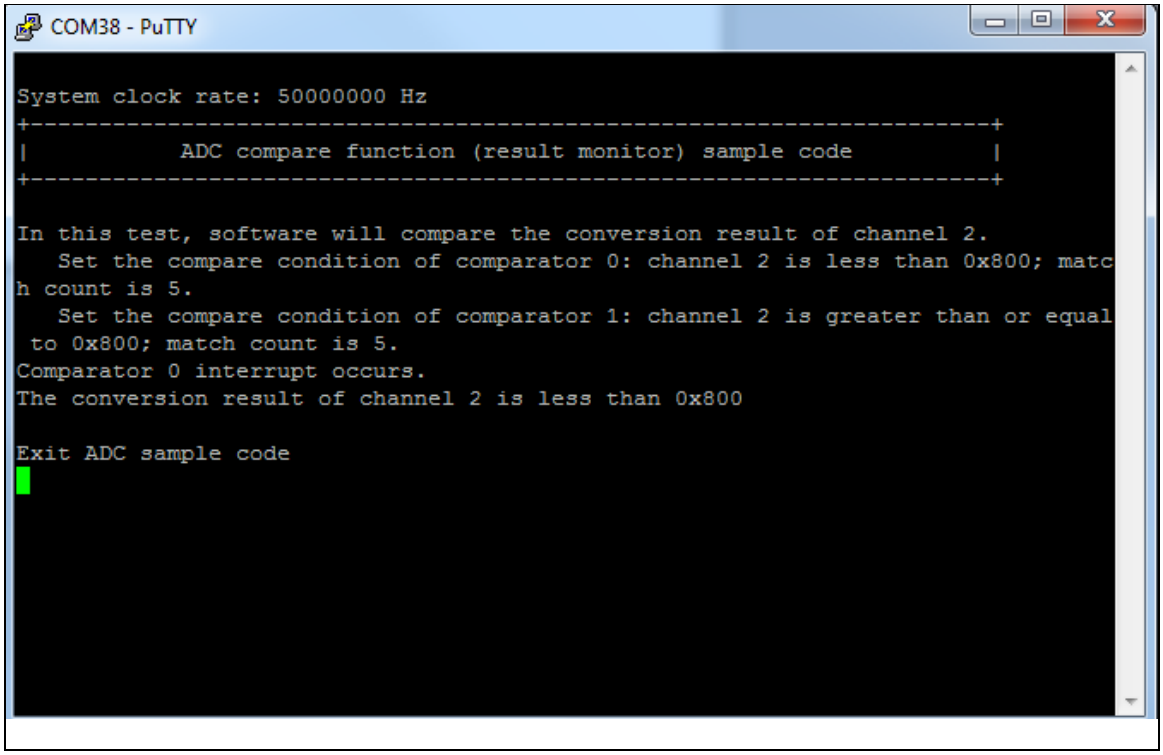


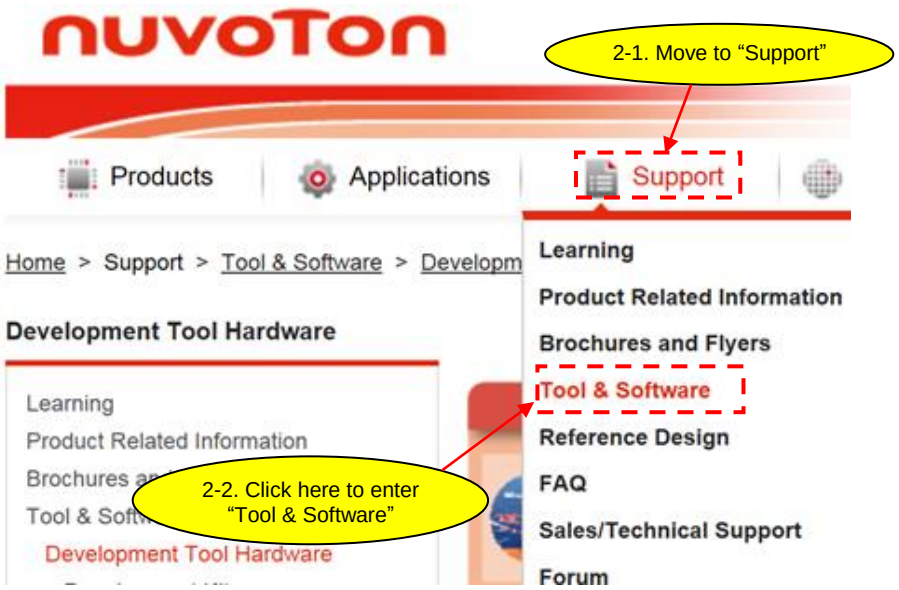
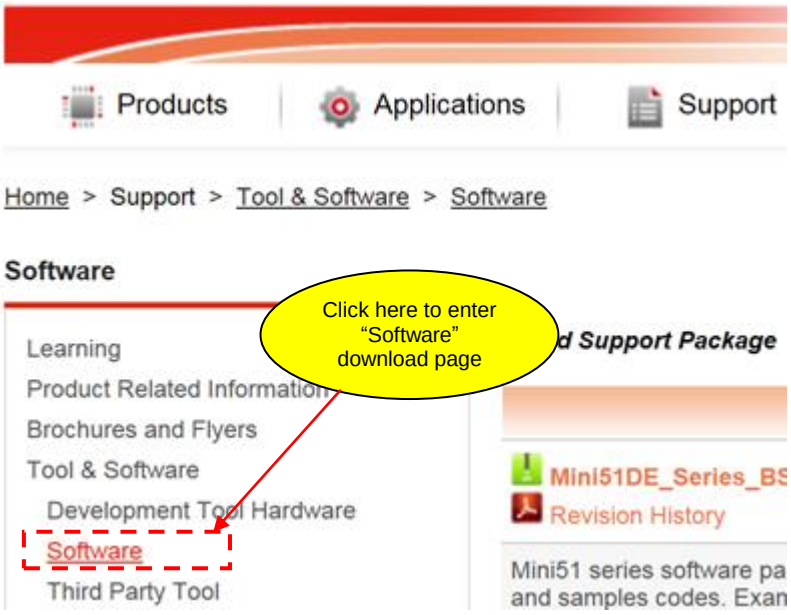
Figure 7-4 Serial Monitor Window

Function Button Description:

- **Open IAR Embedded Workbench**
  File – Open – Workspace
  Open the SYS.eww workspace file
- **Project – Make**
  Compile and link the SYS application
- **Project – Download and Debug**
  Program the application code into on-chip Flash ROM. It has the following features:
  - Single-step through code
  - Reset the device
  - Run the application

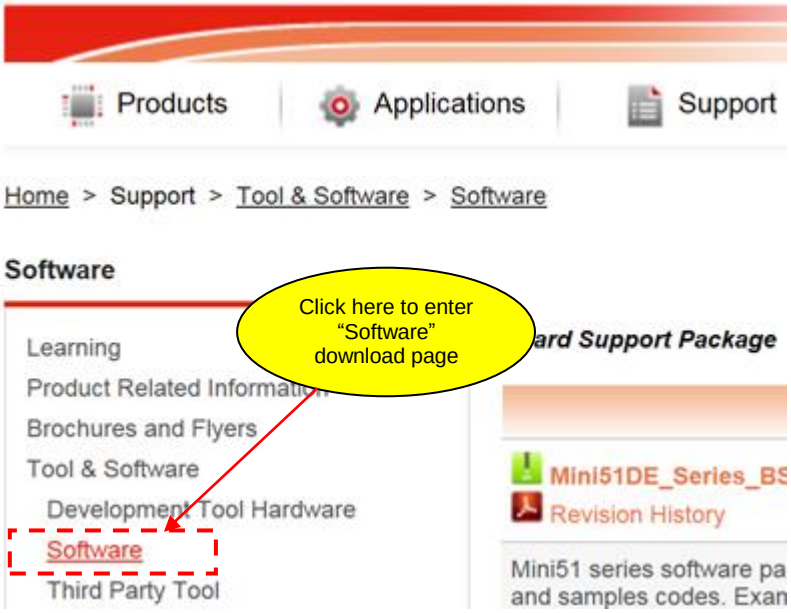
8 DOWNLOADING NU-LINK DRIVER FROM NUVOTON WEBSITE

8.1 Downloading and Installing Nu-Link Keil Driver

Step 1	Visit Nuvoton NuMicro® Website: http://www.nuvoton.com/NuMicro
Step 2	
Step 3	

Step 4	
Step 5	Download "Nu-Link Driver for Keil RVMDK" file. After download is complete, unzip the file and execute "Nu-Link_Keil_Driver.exe". Follow the instructions to install Nu-Link Keil driver.

8.2 Downloading and Installing Nu-Link IAR Driver

Step 1	Visit Nuvoton NuMicro® Website: http://www.nuvoton.com/NuMicro
Step 2	
Step 3	

Step 4

Programmer Software Tools Package

File name	Description	Version	Date
ICP_Programming_Tool_V2.00.6561 Revision History	NuMicro ICP tool & user manual	V2.00.6561	2016-8-16
ISP Programming Tool V2.00.zip Revision History	NuMicro ISP programming tool & user manual	V2.00	2016-9-7
NuGang Programmer V7.02.zip Revision History	NuGang programmer software & user manual	V7.02	2015-11-27

Nu-Link Driver

Click here to download "Nu-Link IAR Driver"

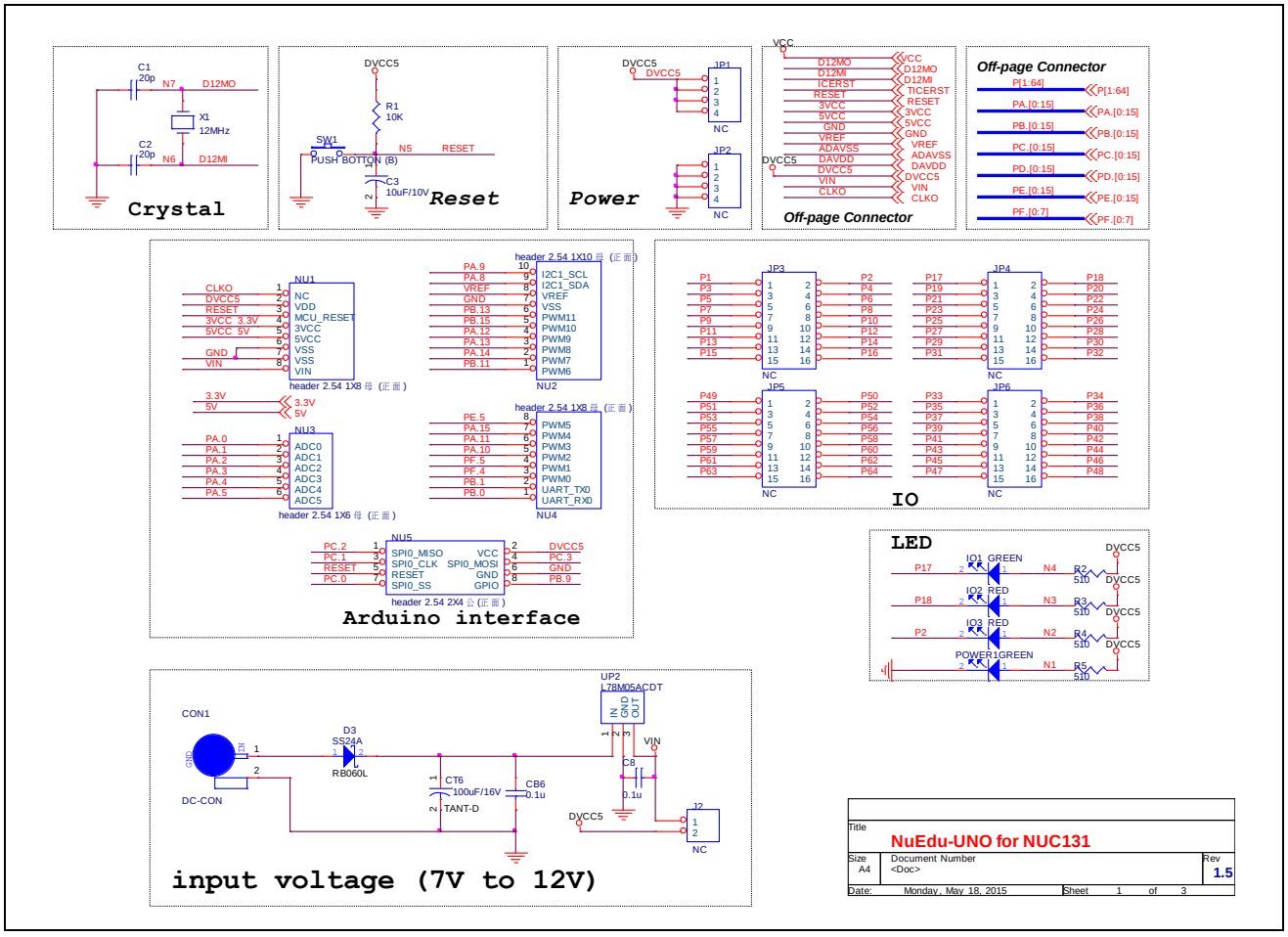
File name	Description	Version	Date
Nu-Link_Keil_Driver_V2.00.6561 Revision History	This driver is to support Nu-Link to work under Keil RVMDK Development Environment for all NuMicro Family Devices.	V2.00.6561	2016-8-16
Nu-Link_IAR_Driver_V2.00.6561 Revision History	This driver is to support Nu-Link to work under IAR EWARM Development Environment for all NuMicro Family Devices.	V2.00.6561	2016-8-16

Step 5

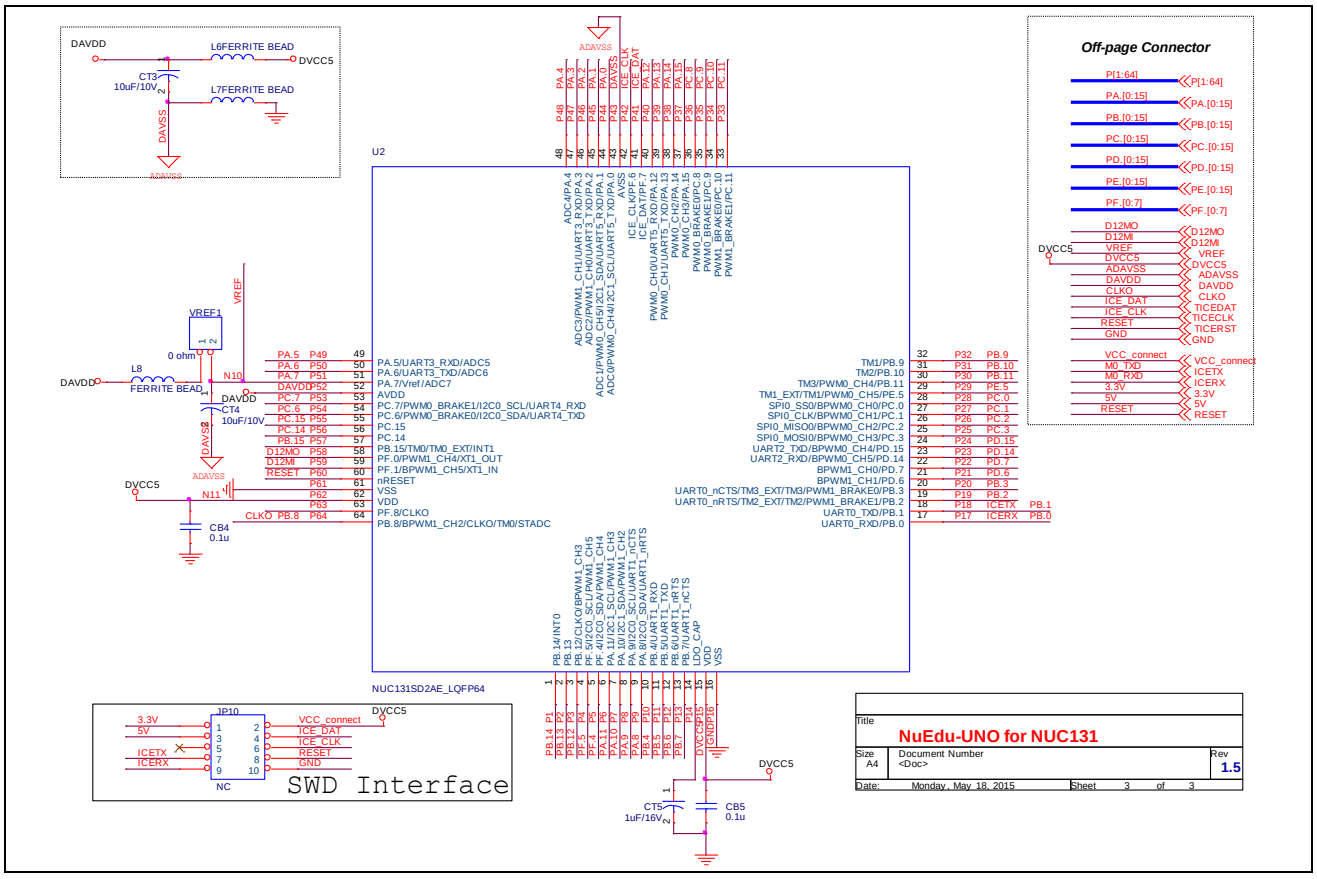
Download "Nu-Link Driver for IAR" file. After download is complete, unzip the file and execute "Nu-Link_IAR_Driver.exe". Follow the instructions to install Nu-Link IAR driver.

9 NUMAKER UNO SCHEMATICS

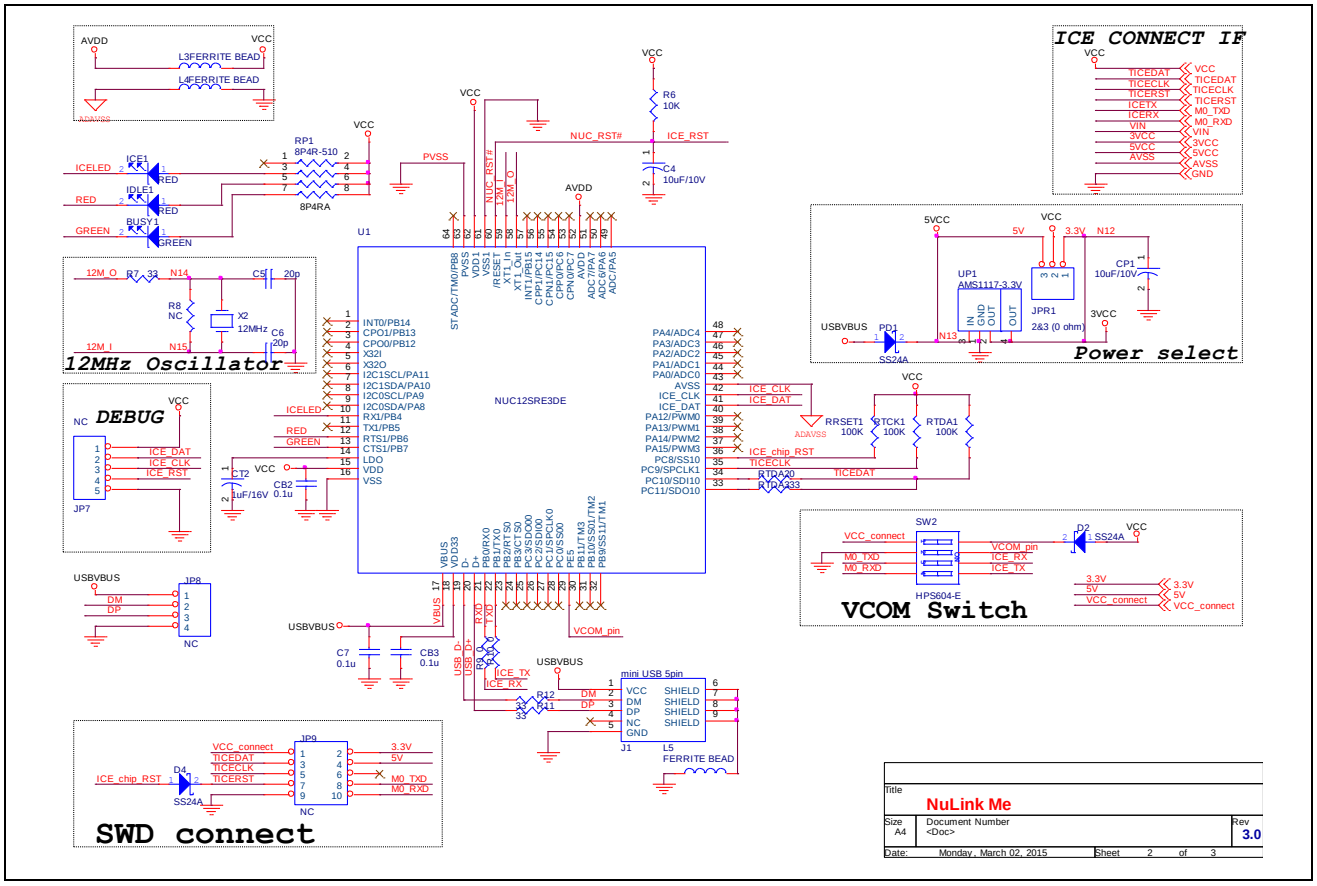
9.1 NuMaker UNO I/O Schematic



9.2 NuMaker UNO NUC131SD2AE Schematic



9.3 Nu-Link-Me Schematic



10 REVISION HISTORY

Date	Revision	Description
2015.07.20	1.00	1. Initially issued.
2016.11.02	1.01	1. Changed the board name from NuEdu-UNO to NuMaker UNO. 2. Updated NuMicro® Patch for Arduino 1.5.8 to Nu-Link_USB_Driver_V1.2 in section 4.2. 3. Added section 3.3 NuMaker UNO NUC131SD2AE Extended Connectors Layout.
2018.08.20	1.02	1. Update installing instruction in section 4.1, 4.2 . 2. Update NuMaker-UNO package for Arduino 1.8.5

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.