Nu-Link2-Pro
Debugger and Programmer
User Manual

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

www.nuvoton.com

Table of Contents

List of Figures

List of Tables

1 INTRODUCTION

The Nu-Link2-Pro is a powerful Debugger and Programmer for Nuvoton NuMicro® Family microcontrollers. The usage of Nu-Link2-Pro can vary from software and hardware development to mass production.

The Nu-Link2-Pro Debugger and Programmer provides SWD and ETM debugging and emulator support for the NuMicro® Family microcontrollers. The system can program target chips based on In-Circuit Programming (ICP) and SWD interface. The NuMicro® Family microcontroller programming are supported by a diverse range of IDEs, such as Keil MDK, IAR EWARM, and NuEclipse GCC. With the Nu-Link2-Pro, users can program and debug directly on IDEs with full access and visibility into the microcontrollers.

The Nu-Link2-Pro can be used as a mass production programmer for NuMicro® Family microcontrollers. The programming system is based on In-Circuit Programming (ICP) and SWD interface. The Nu-Link2-Pro can work with Nuvoton NuMicro® ICP Programming Tool, or serve as a stand-alone ICP programmer. It also provides a control bus interface that can connect to Automated IC programming system. The programming process can be triggered by ICP Programming tool, Physical button, or Automated IC programming system.

The Nu-Link2-Pro is also an In-System Programming (ISP) programmer. It can work with Nuvoton NuMicro® ISP Programming Tool, or serve as a stand-alone (Offline) ISP programmer. It provides multi-interfaces bridge, such as UART, RS-485, USB, I²C, SPI, and CAN, to perform ISP function to NuMicro® Family microcontrollers.

The Nu-Link2-Pro provides a Virtual COM part for a microcontroller to communicate to PC. It also supports multi-interfaces, such as UART, RS-485, I²C, SPI, and CAN analyzer function. For DAPLink and PyOCD, Nuvoton provides dedicated firmware of Nu-Link2-Pro to support them.

For simplicity and clarity, parts of specific terms in this user manual are contracted or abbreviated, as listed in Table 1.1-1.

Short Name	Full Name
NuMicro® Family	Nuvoton NuMicro® Family
ICP Tool	Nuvoton NuMicro® ICP Programming Tool
Keil MDK	Keil ARM Microcontroller Development Kit (MDK-ARM®)
IAR EWARM	IAR Embedded Workbench for ARM
NuEclipse GCC	NuEclipse Integrated Development Environment
SWD	Serial Wire Debug
ICP	In-Circuit Programming
ISP	In-System Programming
ETM	Embedded Trace Macrocell

Table 1.1-1 Nu-Link Debugger/ Programmer Technical Abbreviations

1.1 Nu-Link2-Pro Features

- Supports programming and debugging of all NuMicro® Family microcontrollers
- Supports In-Circuit Programming (ICP)
  - Selectable SWD output voltage (1.8 V / 2.5 V / 3.3 V / 5.0 V)
  - ICP Programming Tool with image file protection
  - Drag & drop Flash programming
  - USB flash drive, SD card and SPI Flash as image file storage
  - Start button
  - Automatic IC programming system connector (Control Bus)
  - Powered by Micro USB or target-powered via SWD interface
- Support In-System Programming (ISP)
  - Via multi-interfaces bridge
  - Supports PC control ISP
  - Supports ISP Programming Tool
- Supports multi-debug interfaces and tool
  - Supports Serial Wire Debug (SWD)
  - Supports high speed up to 96 MHz of Embedded Trace Macrocell (ETM)
  - Unlimited breakpoint and step execution
  - Supports Arm DAPLink
  - Supports PyOCD
- Supports multi-interfaces analyzer
  - SPI, I^2C, CAN and RS-485 signal monitor
- Supports Multiple bridge connect
  - multi-interfaces bridge for ISP function (I^2C, SPI, CAN, UART, RS-485)
  - Virtual COM port by USB

2 GETTING STARTED WITH NU-LINK2-PRO

2.1 Nu-Link2-Pro Kit Contents

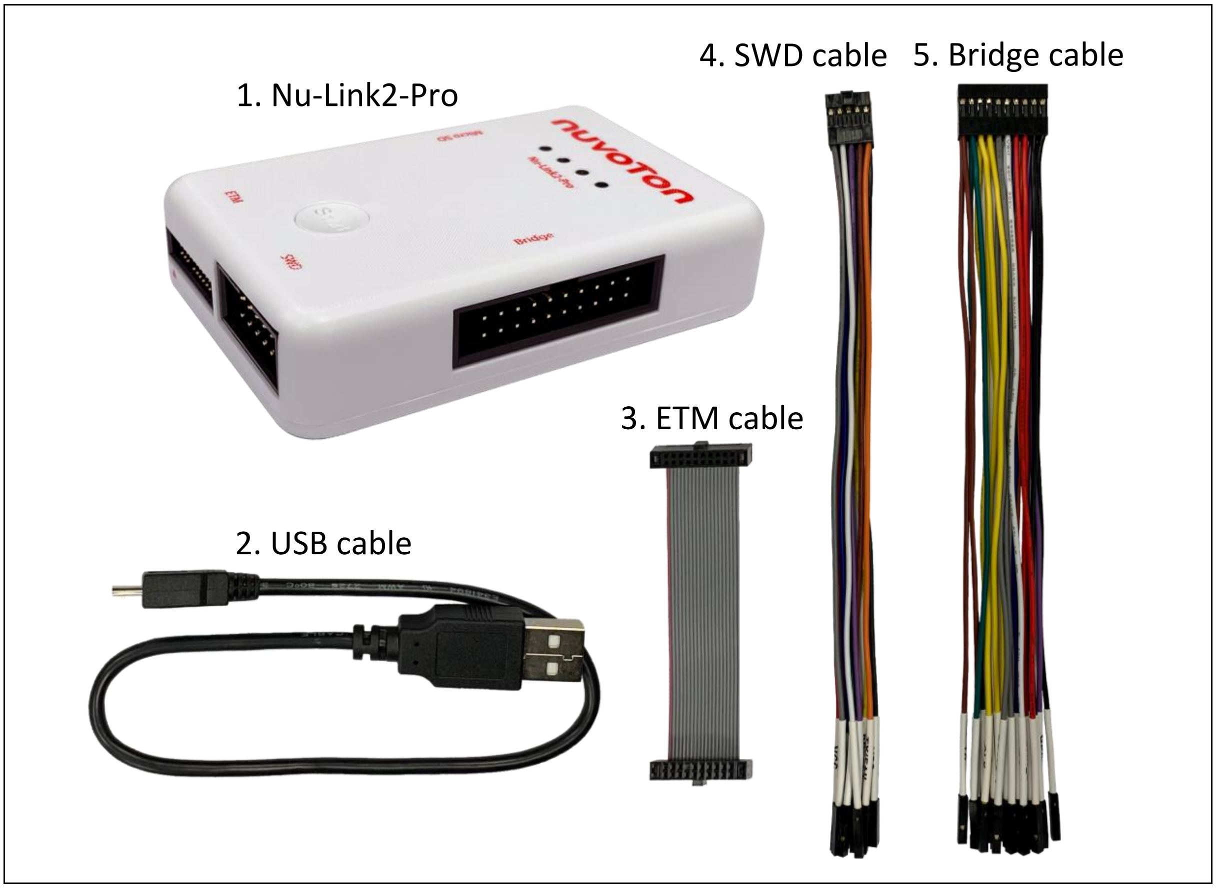


Figure 2.1-1 Nu-Link2-Pro Full Kit Contents

Figure 2.1-1 shows the contents of Nu-Link2-Pro full kit:

- Nu-Link2-Pro main body (2952mil x 1968mil x 688mil)
- USB cable (0.3m, high-speed, Micro-B)
- ETM cable (50-mil 20-pin IDC flat cable with 50-mil 20-pin connectors)
- SWD cable (100-mil 10-pin squid cable with 10 x 100-mil sockets)
- Bridge cable (100-mil 20-pin squid cable with 20 x 100-mil sockets)

2.2 Nu-Link2-Pro PCBA

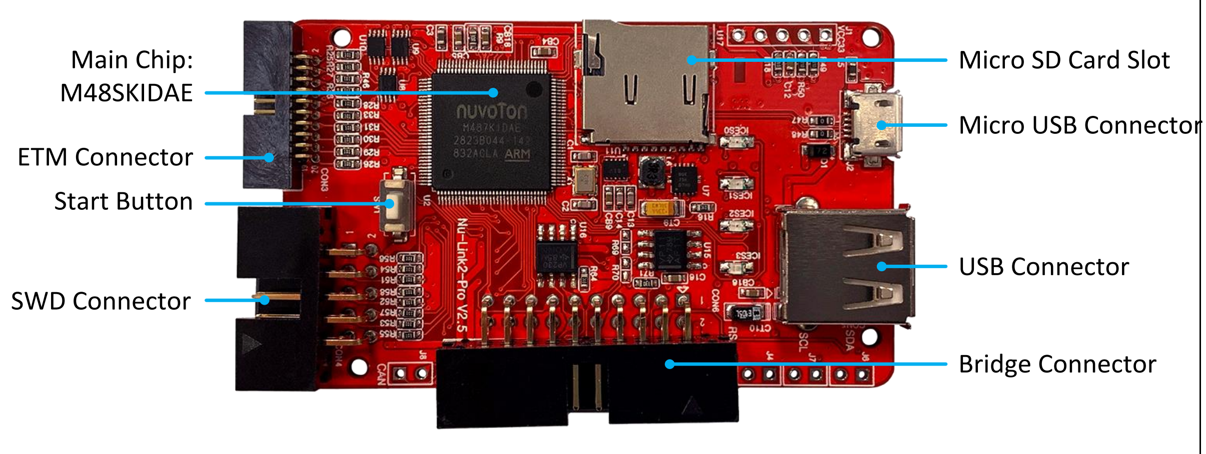


Figure 2.2-1 Front View of Nu-Link2-Pro PCBA

Figure 2.2-1 shows the main components and connectors from the front side of Nu-Link2-Pro PCBA.

The following lists components and connectors from the front view:

- Main Chip: M48SKIDAE
- Micro SD Card Slot
- Micro USB Connector
- USB Connector
- Bridge Connector
- SWD Connector
- Start Button
- ETM Connector

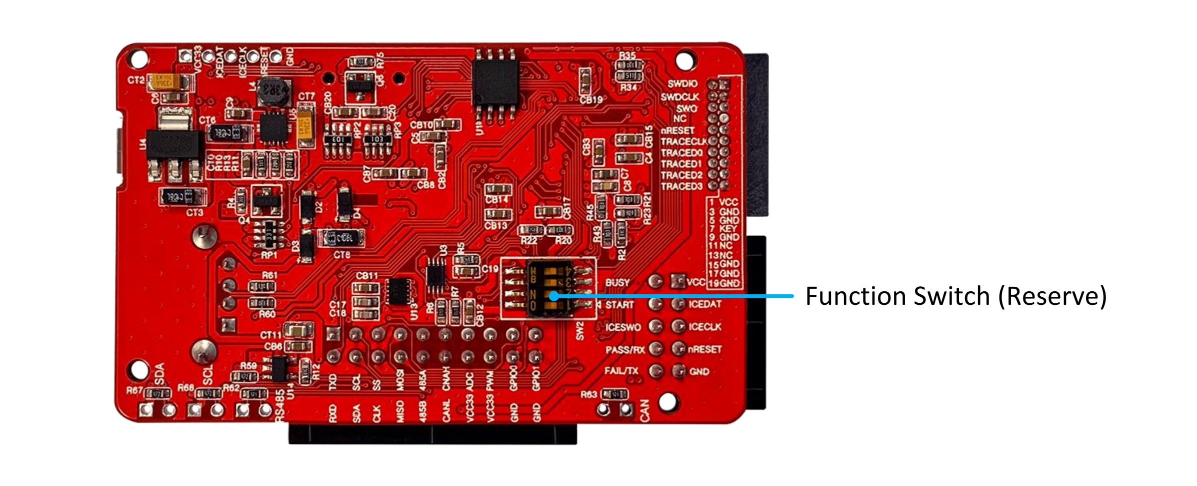


Figure 2.2-2 Rear View of Nu-Link2-Pro PCBA

Figure 2.2-2 shows the main components and connectors from the rear side of Nu-Link2-Pro PCBA.

The following lists components and connectors from the rear view:

- Function Switch (Reserved)

2.3 Nu-Link2-Pro Overview

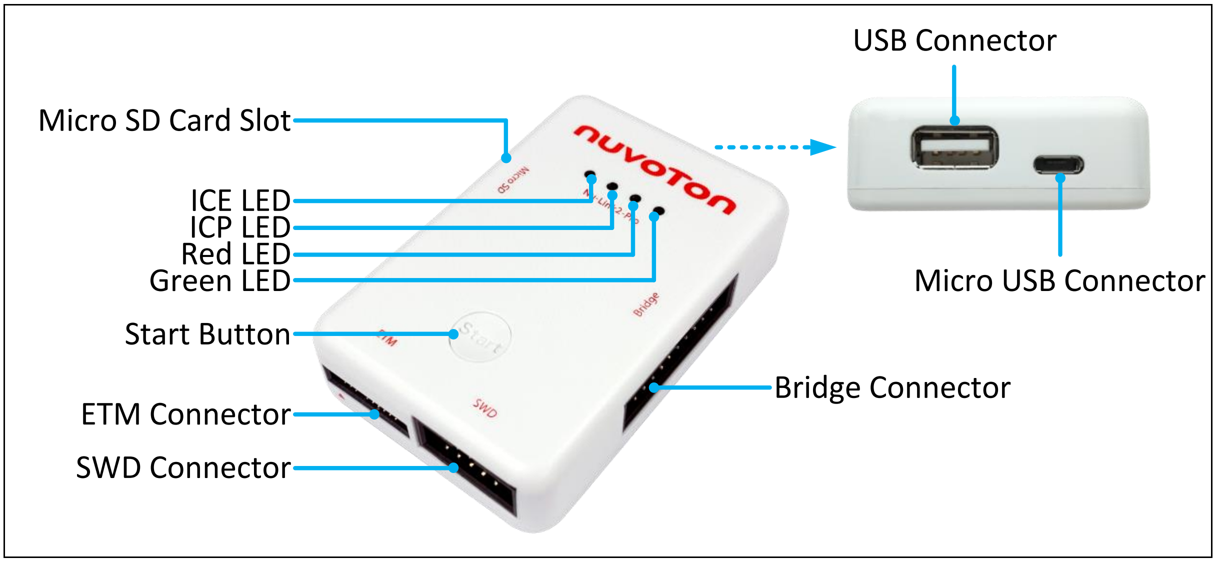


Figure 2.3-1 Nu-Link2-Pro Connector and Function Overview

Figure 2.3-1 shows the Nu-Link2-Pro profile and connector overview, the following lists of function brief description

- USB Connector (CON5)
  - USB Flash Drive for ICP Offline Programming
- Micro USB Connector (J2)
  - Micro USB port of a PC to debug and program target chips through the development software tool
- Bridge Connector (CON6)
  - UART (Only supports multi-interfaces analyzer related information transmission)
  - I^{2}C Transmission Interface
  - SPI Transmission Interface
  - RS-485 Transmission Interface
  - CAN BUS Transmission Interface
  - PWM/Capture
  - ADC
  - GPIO
- SWD Connector (CON4)
  - SWD Host Interface
  - ICP Offline Programming
  - Virtual COM by UART
  - Automatic IC Programming

- ETM Connector (CON3)
  - ETM Interface
  - SWD Host Interface
- Start Button (SW1)
  - Click this button to proceed with offline programming
- Micro SD Card Slot
  - Save bin file for ICP Offline Programming
- Status LED (ICES0, ICES1, ICES2, ICES3)
  - Display the operation status of the Nu-Link2-Pro

Nu-Link2-Pro Operation Status	**Status LED**			
	ICE	**ICP**	**Red**	**Green**
Boot	Flash×3	Flash×3	Flash×3	Flash×3
One Nu-Link2-Pro selected to connect	Flash×3	Flash×3	Flash×3	On
ICE Online (Not connected with a target chip)	On	-	Flash×3	Flash×3
ICE Online (Connected with a target chip)	On	-	-	On
ICE Online (Failed to connect with a target chip)	On	Any	Flash	On
During Offline Programming	-	On	-	Flash
Offline Programming Completed	On	-	-	-
Offline Programming Completed (Auto mode)	On	On	-	-
Offline Programming Failed	On	Flash	-	-

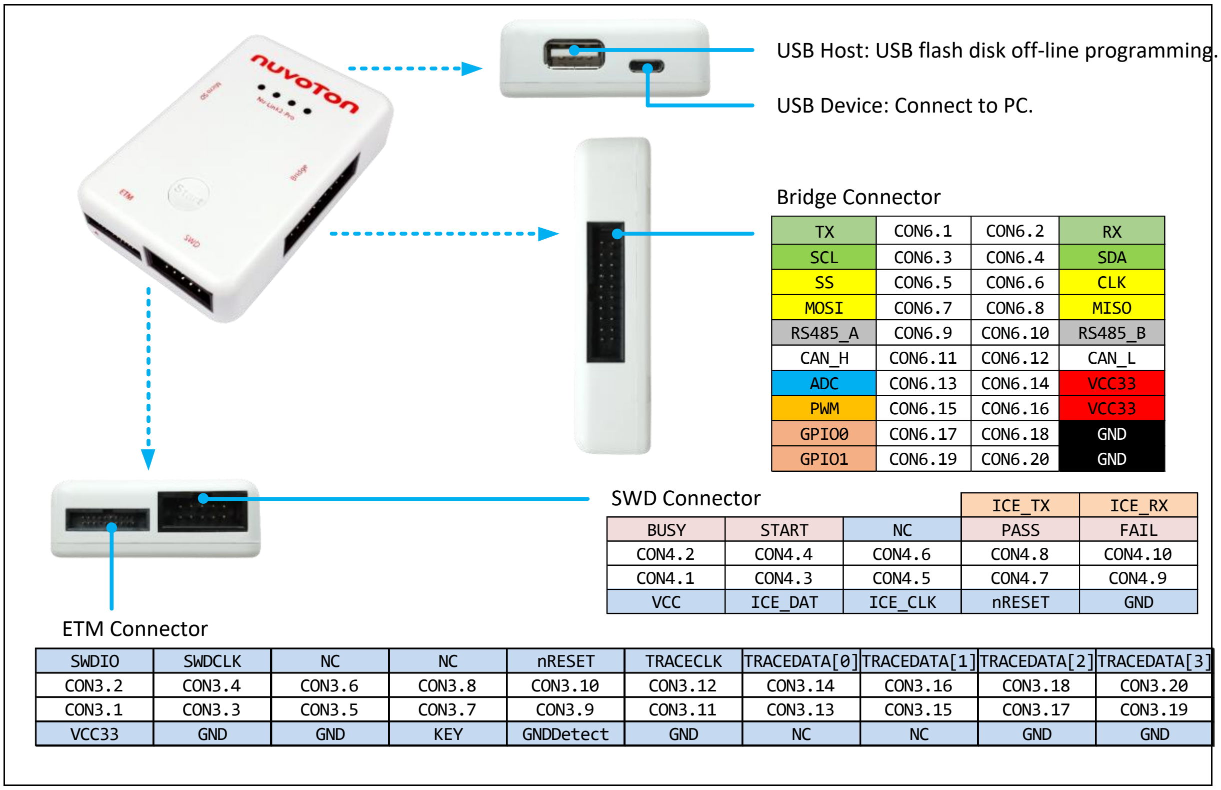
Table 2.3-1 Status LEDs Difference List

3 CONNECTING THE NU-LINK2-PRO

This chapter introduces how to connect the Nu-Link2-Pro to a computer, and how to connect individual connectors to development board or products.

3.1 Nu-Link2-Pro Compatible Extension Connectors

Figure 3.1-1 shows the Nu-Link2-Pro definition pin of each connector, the Nu-Link2-Pro mainly contains USB, Micro USB, Bridge interface, ETM interface and SWD interface. User can freely select a suitable interface for debugger and programmer.



Bridge Connector

TX	CON6.1	CON6.2	RX
SCL	CON6.3	CON6.4	SDA
SS	CON6.5	CON6.6	CLK
MOSI	CON6.7	CON6.8	MISO
RS485_A	CON6.9	CON6.10	RS485_B
CAN_H	CON6.11	CON6.12	CAN_L
ADC	CON6.13	CON6.14	VCC33
PWM	CON6.15	CON6.16	VCC33
GPIO0	CON6.17	CON6.18	GND
GPIO1	CON6.19	CON6.20	GND

SWD Connector

			ICE_TX	ICE_RX
BUSY	START	NC	PASS	FAIL
CON4.2	CON4.4	CON4.6	CON4.8	CON4.10
CON4.1	CON4.3	CON4.5	CON4.7	CON4.9
VCC	ICE_DAT	ICE_CLK	nRESET	GND

ETM Connector

SWDIO	SWDCLK	NC	NC	nRESET	TRACECLK	TRACEDATA[0]	TRACEDATA[1]	TRACEDATA[2]	TRACEDATA[3]
CON3.2	CON3.4	CON3.6	CON3.8	CON3.10	CON3.12	CON3.14	CON3.16	CON3.18	CON3.20
CON3.1	CON3.3	CON3.5	CON3.7	CON3.9	CON3.11	CON3.13	CON3.15	CON3.17	CON3.19
VCC33	GND	GND	KEY	GNDDetect	GND	NC	NC	GND	GND

Figure 3.1-1 Pin Definition of Nu-Link2-Pro Connectors

3.2 SWD Interface Pin Definition and Function Connection

Table 3.2-1 shows SWD interface pin definition and description. The Nu-Link2-Pro provides a SWD interface connector with a 100-mil 10-pin cable. The SWD supports ICE Programming, Virtual COM and automatic IC Programming. The following sections will introduce the definition of the SWD interface pin and the connection of each function.

Pin Name	Pin Number	Pin Description
VCC	CON4.1	Target Board voltage supply. The Nu-Link2-Pro supports the wide voltage programming function, by ICP tool can adjust the SWD port voltage as 1.8V, 3.3V, 2.5V or 5.0V. For detailed adjustment method, please refer to section 4.3.
BUSY	CON4.2	“BUSY” is Control Bus signals for IC Programmer. For details, please refer to section 6.3.
ICE_DAT	CON4.3	Serial Wired Debugger Data pin
START	CON4.4	“START” is Control Bus signals for IC Programmer. For details, please refer to section 6.3.
ICE_CLK	CON4.5	Serial Wired Debugger Clock pin
NC	CON4.6	NC
/RESET	CON4.7	IC reset pin, Nu-Link2-Pro will automatically reset the target IC during the programming process.
PASS/TX	CON4.8	“PASS” is Control Bus signals for IC Programmer. For details, please refer to section 6.3.
GND	CON4.9	Ground
FAIL/RX	CON4.10	“FAIL” is Control Bus signals for IC Programmer. For details, please refer to section 6.3.

Table 3.2-1 SWD Interface Pin Definition and Description

3.2.1 ICE Programming Connection

The Nu-Link2-Pro provides ICE function to Programming and debugging on PC. The ICE connection pins are VCC(CON4.1), ICE_DAT(CON4.3), ICE_CLK(CON4.5), /RESET(CON4.7) and VSS(CON4.9). Figure 3.2-1 presents how to connect the target board to use ICE and Table 3.2-2 shows the pin corresponding to the target board.

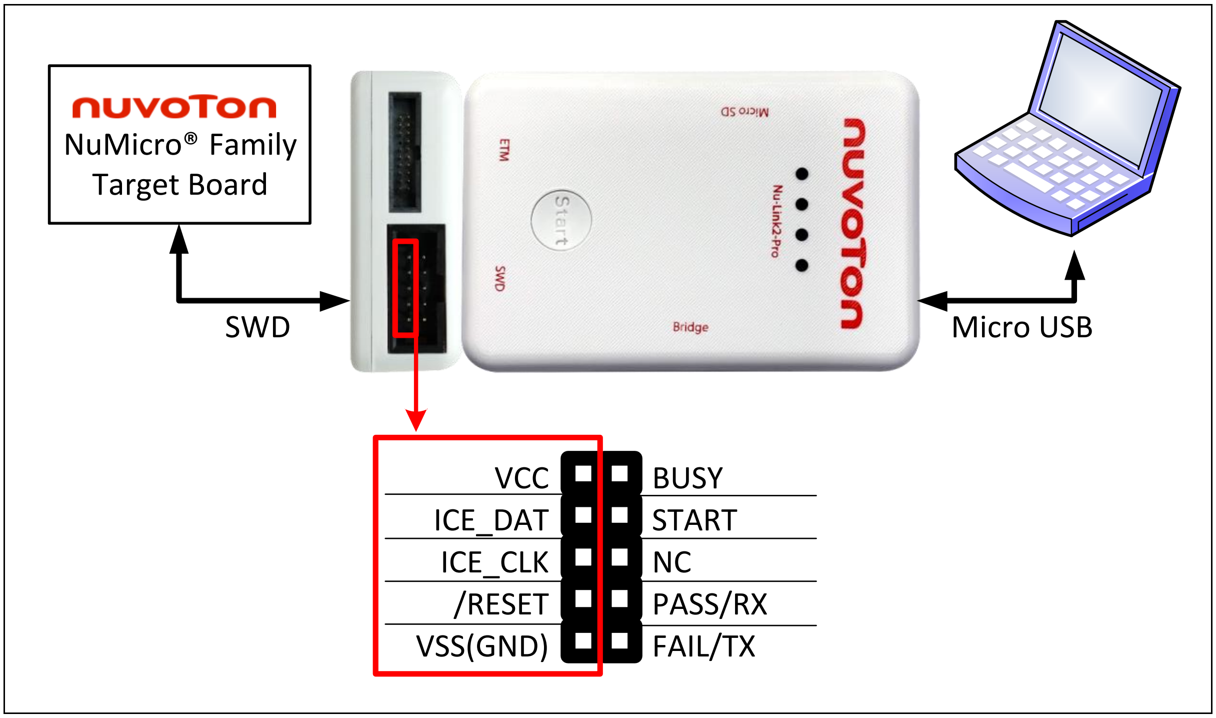


Figure 3.2-1 SWD Interface Connection Diagram for ICE

Pin Name	Pin Number	Pin Corresponding to the Target Board
VCC	CON4.1	VCC
ICE_DAT	CON4.3	ICE_DAT
ICE_CLK	CON4.5	ICE_CLK
/RESET	CON4.7	/RESET
VSS(GND)	CON4.9	VSS(GND)

Table 3.2-2 SWD Interface Corresponding Pin for ICE

3.2.2 **Virtual COM Connection**

The Nu-Link2-Pro provides virtual COM port (VCOM) function to print out messages on PC, and the Virtual COM transmission data by UART0. The connection pins are VCC (CON4.1), VSS (CON4.9), TX (CON4.8) and RX (CON4.10). Figure 3.2-2 presents how to connect the target board to use VCOM and Table 3.2-3 shows the pin corresponding to the target board.

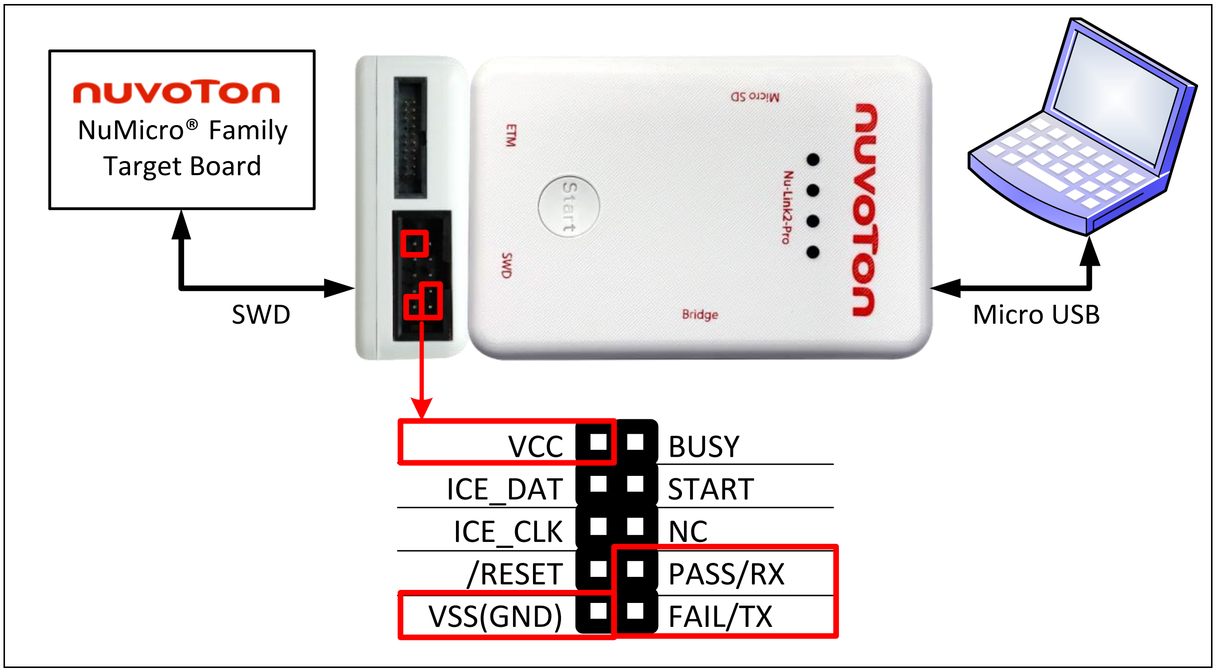


Figure 3.2-2 SWD Interface Connection Diagram for Virtual COM

Pin Name	Pin Number	Pin Corresponding to the Target Board
VCC	CON4.1	VCC
PASS/TX	CON4.8	UART_RX
VSS(GND)	CON4.9	VSS(GND)
FAIL/RX	CON4.10	UART_TX

Table 3.2-3 SWD Interface Corresponding Pin for Virtual COM

3.2.3 Automatic IC Programming Connection

The Nu-Link2-Pro provides Automatic IC Programming function to mass production. The Automatic IC Programming connection pins are VCC (CON4.1), VSS (CON4.9), BUSY (CON4.2), START (CON4.4), PASS (CON4.8) and FAIL (CON4.10). Figure 3.2-3 presents how to connect the target board to use Automatic IC Programming and Table 3.2-4 shows the pin corresponding to the target board.

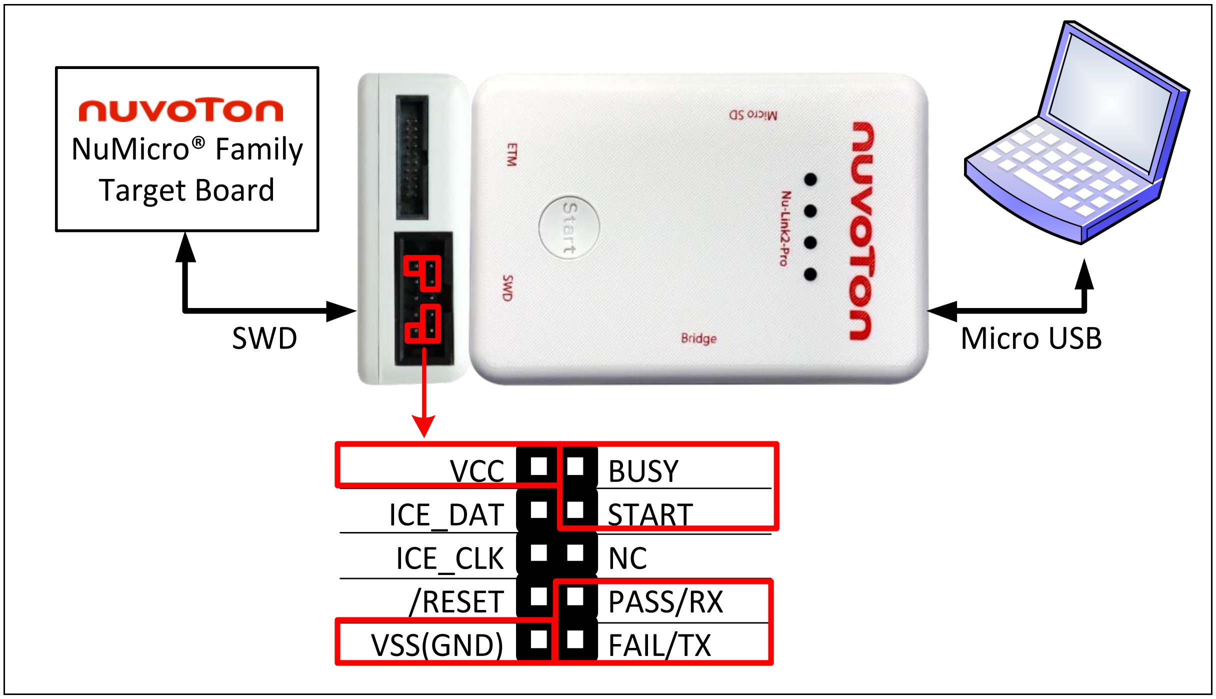


Figure 3.2-3 SWD Interface Connection Diagram for Automatic IC Programming

Pin Name	Pin Number	Pin Corresponding to the Target Board
VCC	CON4.1	VCC[1]
BUSY	CON4.2	BUSY
START	CON4.4	START
PASS	CON4.8	PASS
VSS(GND)	CON4.9	VSS(GND)
FAIL	CON4.10	FAIL
Note: 1. The target board power setting should be the same as Nu-Link2-Pro.		

Table 3.2-4 SWD Interface Corresponding Pin for Automatic IC Programming

3.3 Bridge Interface Pin Definition and Function Connection

Table 3.3-1 shows the bridge interface pin definition and description. The Nu-Link2-Pro provides a bridge interface connector with a 100-mil 20-pin cable. The bridge interface supports one channel UART, I²C, SPI, RS-485, CAN BUS, ADC, PWM and two GPIOs. The following sections will introduce the definition of the bridge interface pin and the connection of each function.

Pin Name	Pin Number	Pin Description
TXD	CON6.1	Data transmitter output pin for UART
RXD	CON6.2	Data receiver input pin for UART
SCL	CON6.3	I²C clock
SDA	CON6.4	I²C data input/output
SS	CON6.5	SPI slave select
CLK	CON6.6	SPI serial clock
MOSI	CON6.7	SPI MOSI (Master Out, Slave In)
MISO	CON6.8	SPI MISO (Master In, Slave Out)
RS-485A	CON6.9	RS-485 Data plus signal
RS-485B	CON6.10	RS-485 Data minus signal
CANH	CON6.11	CAN BUS Data plus signal
CANL	CON6.12	CAN BUS Data minus signal
ADC	CON6.13	ADC analog input signal
VCC33	CON6.14	Target Board voltage supply. The Nu-Link2-Pro Bridge VCC only support 3.3V.
PWM	CON6.15	PWM output/Capture input
VCC33	CON6.16	Target Board voltage supply. The Nu-Link2-Pro Bridge VCC only support 3.3V.
GPIO0	CON6.17	General Purpose I/O 0
GND	CON6.18	Ground
GPIO1	CON6.19	General Purpose I/O 1
GND	CON6.20	Ground

Table 3.3-1 Bridge Interface Pin Definition and Description

3.3.1 UART Connection

The Nu-Link2-Pro provides one channel UART function for monitor mode print out information. This information is received from I²C, SPI, RS-485 or CAN BUS. The UART connection pins are VCC33 (CON6.14 and CON6.16), VSS(CON6.18 and CON6.20), TXD(CON6.1) and RXD(CON6.2). Figure 3.3-1 presents how to connect the target board to use UART function and Table 3.3-2 shows the pin corresponding to the target board.

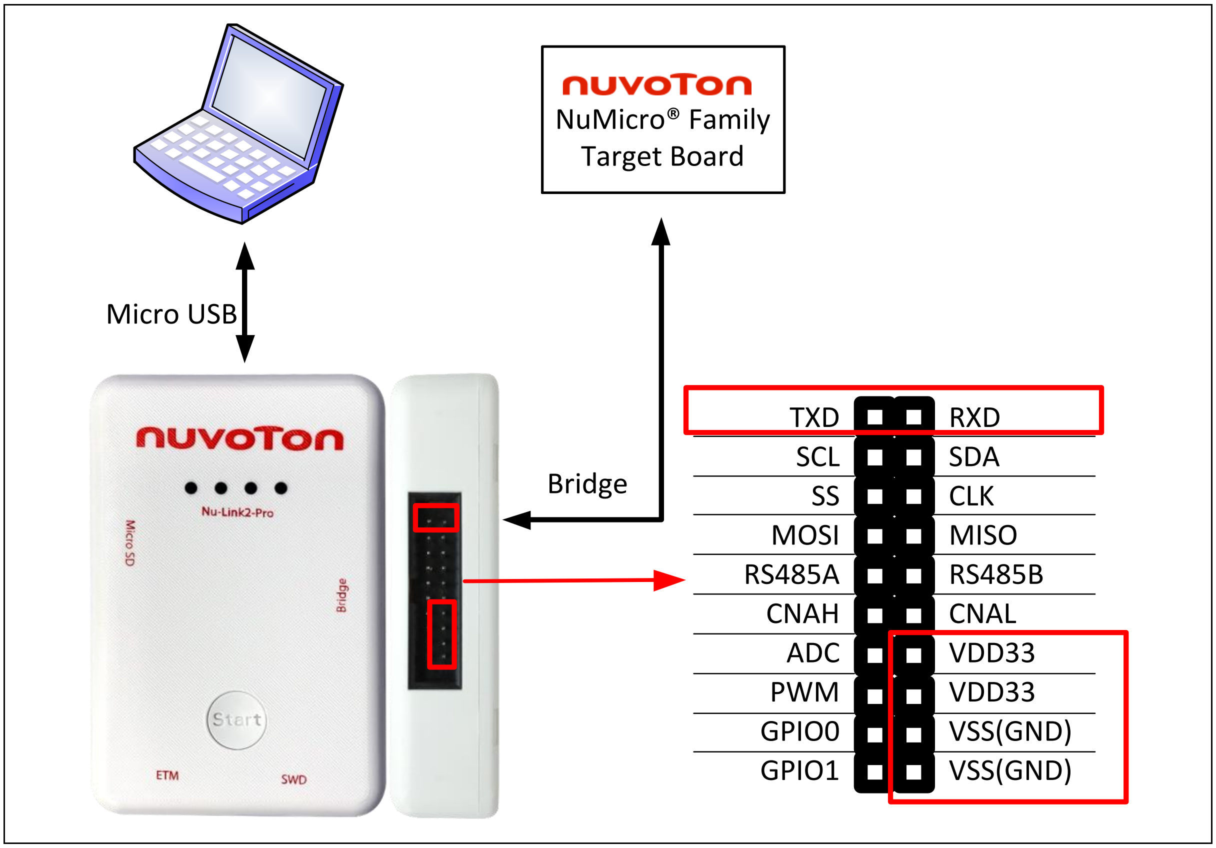


Figure 3.3-1 Bridge Interface Connection Diagram for UART

Pin Name	Pin Number	Pin Corresponding to the Target Board
TXD	CON6.1	RXD[1]
RXD	CON6.2	TXD[1]
VCC33	CON6.14	VCC[1]
VCC33	CON6.16	VCC[1]
VSS(GND)	CON6.18	VSS(GND)
VSS(GND)	CON6.20	VSS(GND)

Note:

1. The target board power and signal only supports 3.3V at Nu-Link2-Pro Bridge interface.

Table 3.3-2 Bridge Interface Corresponding Pin for UART

3.3.2 **I²C Connection**

The Nu-Link2-Pro provides one channel I²C function for monitor mode receive information, and print out information by UART. The I²C connection pins are VCC33(CON6.14 and CON6.16), VSS (CON6.18 and CON6.20), SCL(CON6.3) and SDA(CON6.4). Figure 3.3-2 presents how to connect the target board to use I²C function and Table 3.3-3 shows the pin corresponding to the target board.

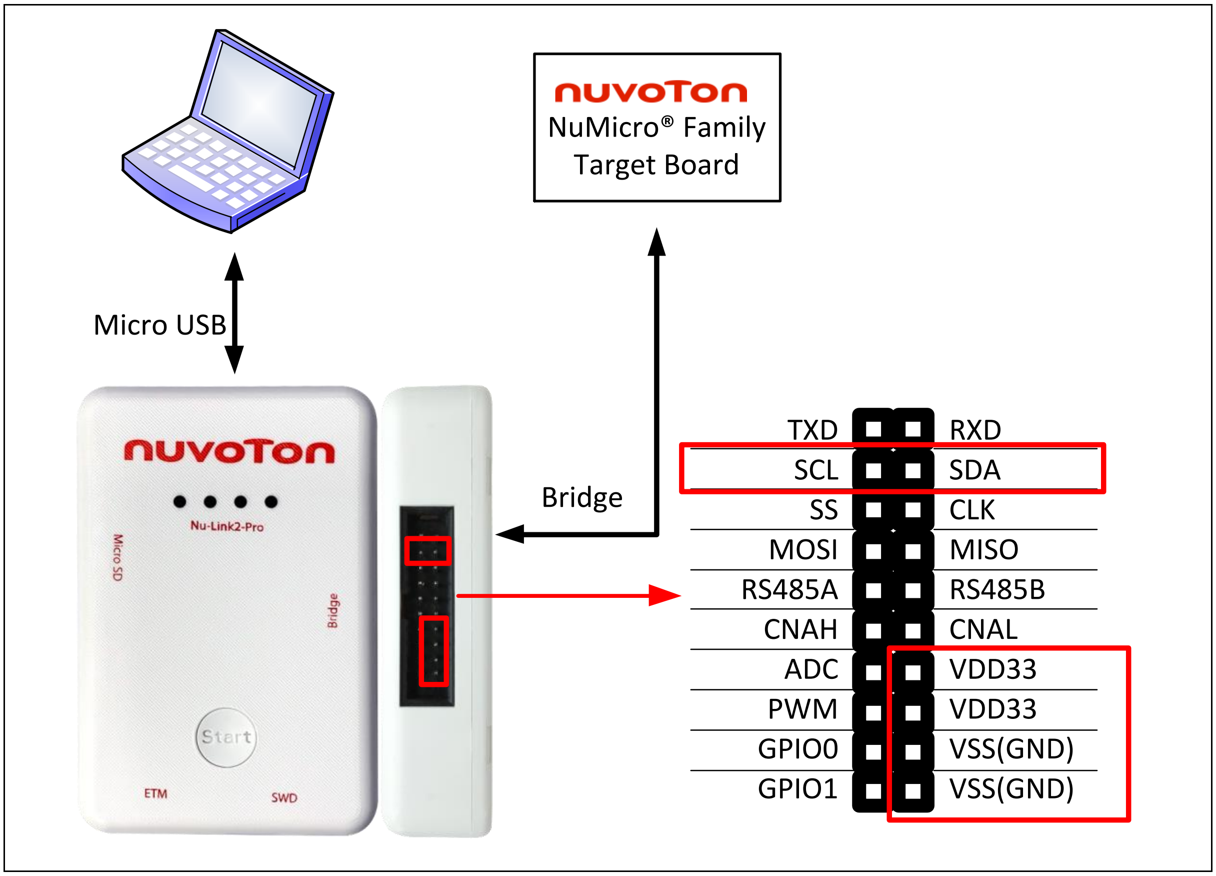


Figure 3.3-2 Bridge Interface Connection Diagram for I²C

Pin Name	Pin Number	Pin Corresponding to the Target Board
SCL[1]	CON6.3	SCL[2]
SDA[1]	CON6.4	SDA[2]
VCC33	CON6.14	VCC[2]
VCC33	CON6.16	VCC[2]
VSS(GND)	CON6.18	VSS(GND)
VSS(GND)	CON6.20	VSS(GND)

Note:

1. Internal 4.7 kΩ pull-up resistors R67 and R68 on Nu-Link2-Pro; the user can adjust them according to needs.
2. The target board power and signal only support 3.3 V at Nu-Link2-Pro Bridge interface.

Table 3.3-3 Bridge Interface Pin for I²C

3.3.3 SPI Connection

The Nu-Link2-Pro provides one channel SPI function for monitor mode receive information, and print out information by UART. The SPI connection pins are VCC33(CON6.14 and CON6.16), VSS (CON6.18 and CON6.20), SS(CON6.5), CLK(CON6.6), MOSI(CON6.7) and MISO(CON6.8). Figure 3.3-3 presents how to connect the target board to use SPI function and Table 3.3-4 shows the pin corresponding to the target board.

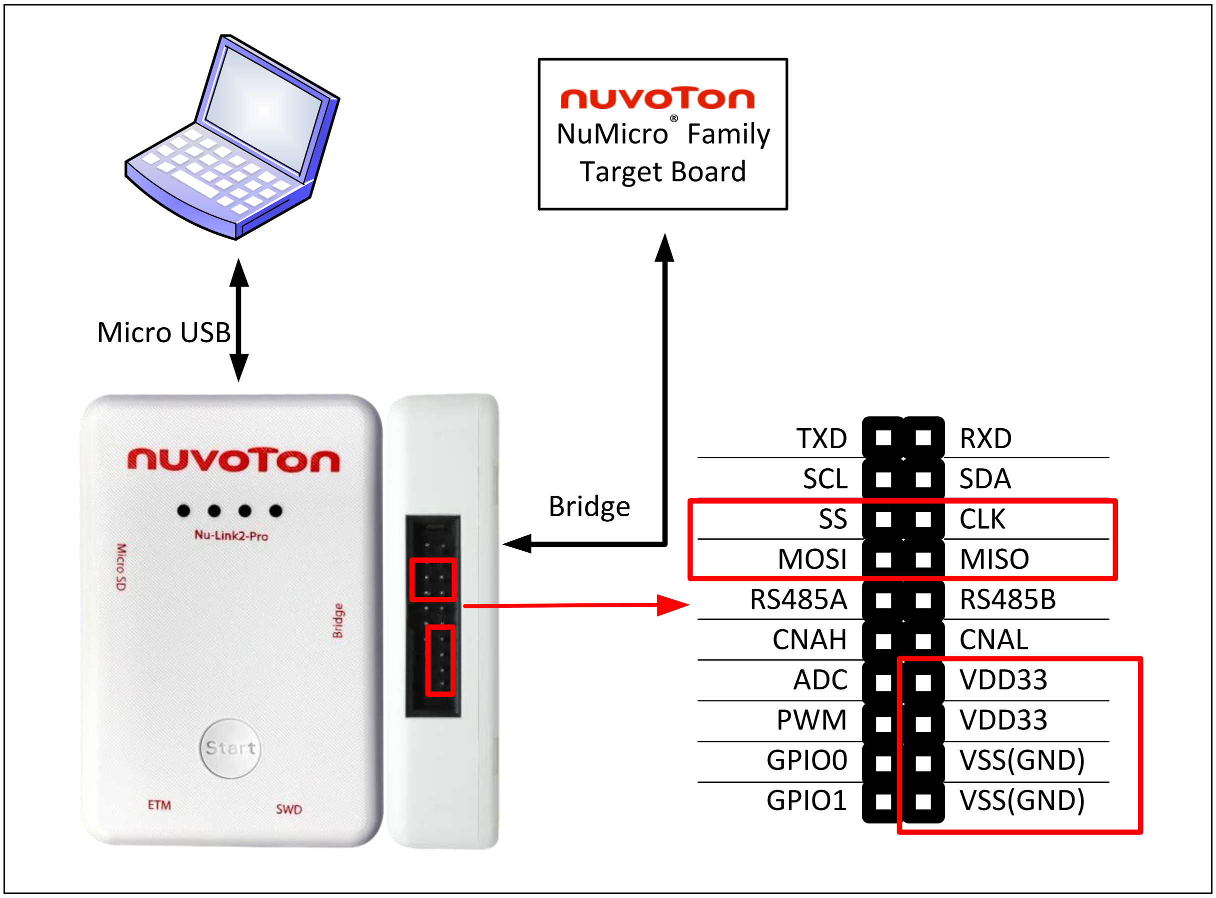


Figure 3.3-3 Bridge Interface Connection Diagram for SPI

Pin Name	Pin Number	Pin Corresponding to the Target Board
SS	CON6.5	SS[1]
CLK	CON6.6	CLK[1]
MOSI	CON6.7	MOSI[1]
MISO	CON6.8	MISO[1]
VCC33	CON6.14	VCC[1]
VCC33	CON6.16	VCC[1]
VSS (GND)	CON6.18	VSS (GND)

VSS (GND)	CON6.20	VSS (GND)
Note: The target board power and signal only support 3.3 V at Nu-Link2-Pro Bridge interface.		

Table 3.3-4 Bridge Interface Corresponding Pin for SPI

3.3.4 RS-485 Connection

The Nu-Link2-Pro provides one channel RS-485 function for monitor mode receive information, and print out information by UART. The RS-485 connection pins are VCC33(CON6.14 and CON6.16), VSS(CON6.18 and CON6.20), RS485A(CON6.9) and RS485B(CON6.10). Figure 3.3-4 presents how to connect the target board to use RS-485 function and Table 3.3-5 shows the pin corresponding to the target board.

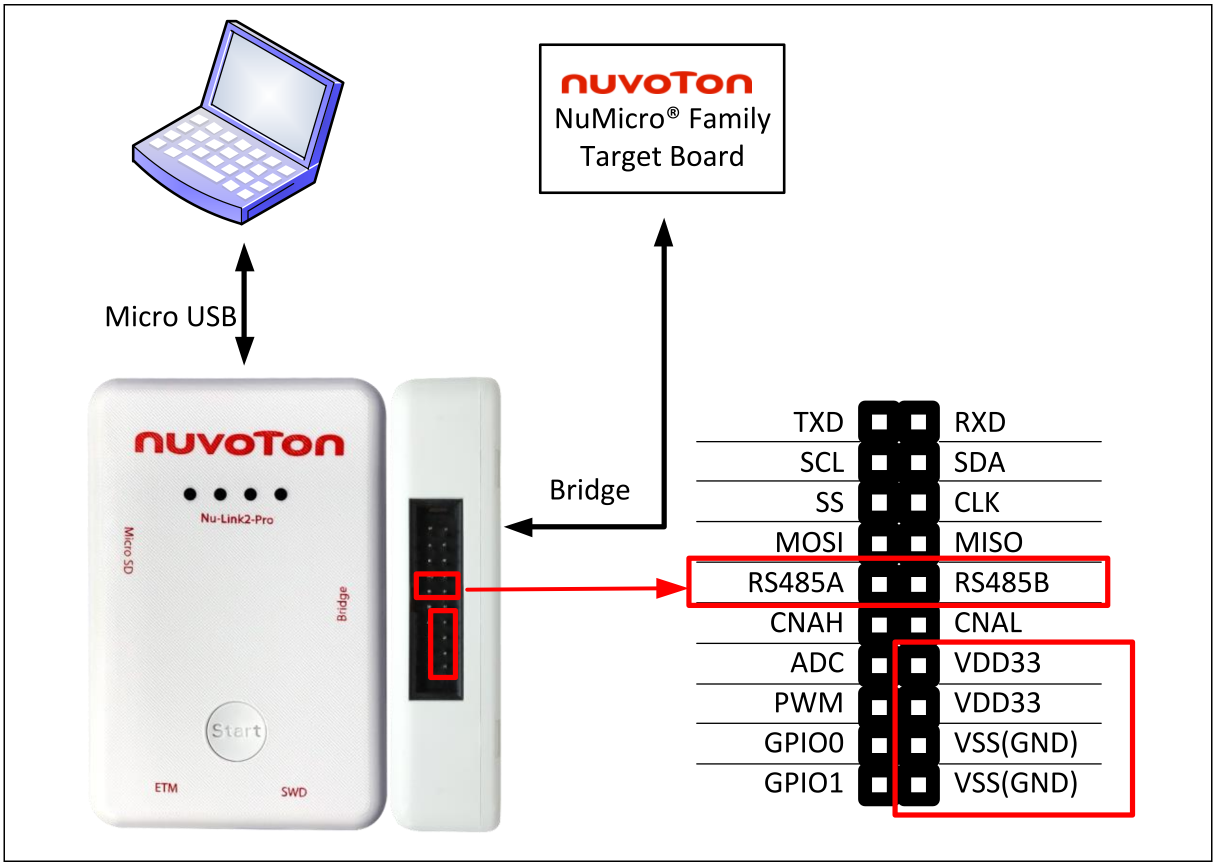


Figure 3.3-4 Bridge Interface Connection Diagram for RS-485

Pin Name	Pin Number	Pin Corresponding to the Target Board
RS485A[1]	CON6.9	RS485A[2]
RS485B[1]	CON6.10	RS485B[2]
VCC33	CON6.14	VCC[2]
VCC33	CON6.16	VCC[2]

VSS(GND)	CON6.18	VSS(GND)
VSS(GND)	CON6.20	VSS(GND)
Note: 1. Internal 120 Ω terminal resistors R62 on Nu-Link2-Pro; the user can adjust them according to needs. 2. The target board power and signal only support 3.3 V at Nu-Link2-Pro Bridge interface.		

Table 3.3-5 Bridge Interface Corresponding Pin for RS-485

3.3.5 CAN BUS Connection

The Nu-Link2-Pro provides one channel CAN BUS function for monitor mode receive information, and print out information by UART. The CAN BUS connection pins are VCC33(CON6.14 and CON6.16), VSS(CON6.18 and CON6.20), CANH(CON6.11) and CANL(CON6.12). Figure 3.3-5 presents how to connect the target board to use CAN BUS function and Table 3.3-6 shows the pin corresponding to the target board.

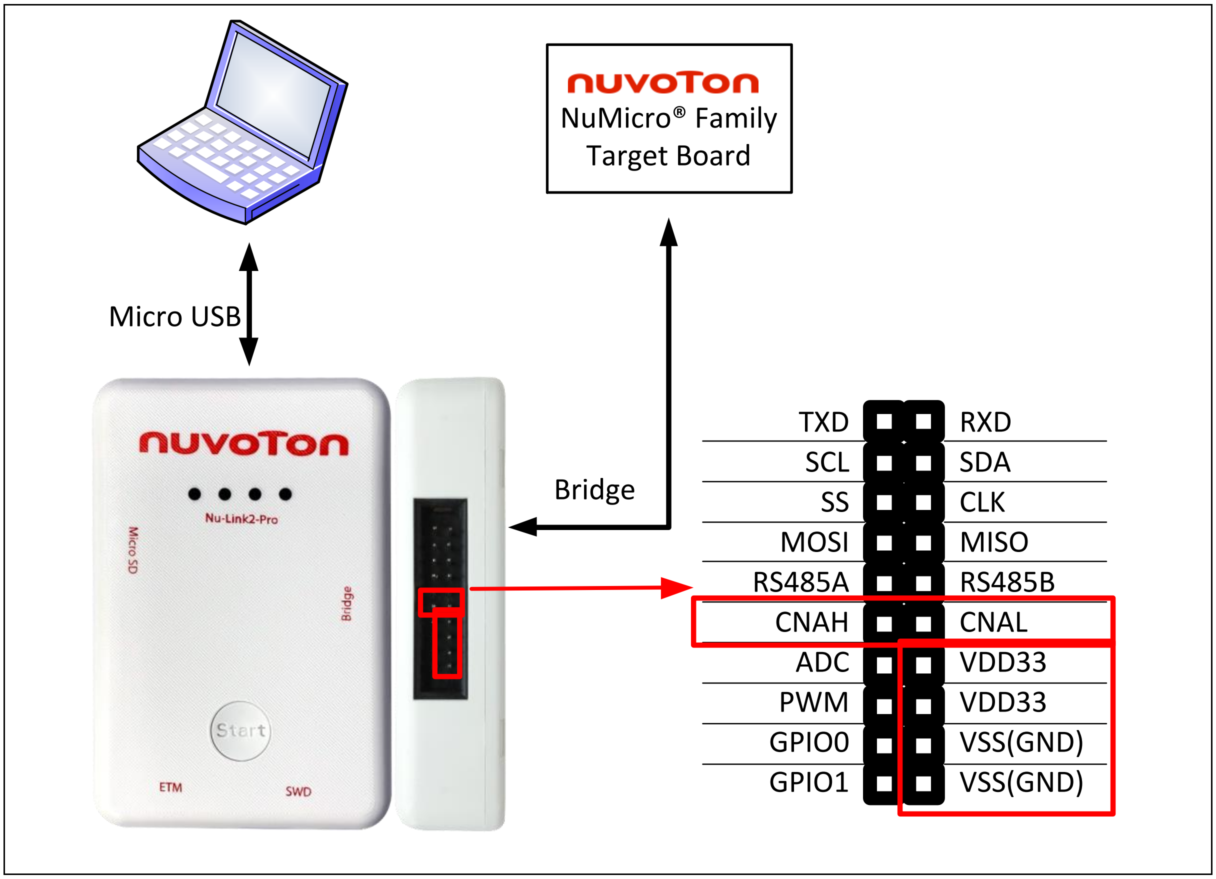


Figure 3.3-5 Bridge Interface Connection Diagram for CAN BUS

Pin Name	Pin Number	Pin Corresponding to the Target Board
CANH[1]	CON6.11	CANH

CANL[1]	CON6.12	CANL
VCC33	CON6.14	VCC[2]
VCC33	CON6.16	VCC[2]
VSS(GND)	CON6.18	VSS(GND)
VSS(GND)	CON6.20	VSS(GND)

Note:

1. Internal 120 Ω terminal resistors R63 on Nu-Link2-Pro; the user can self-adjust them according to needs.
2. The target board power and signal only support 3.3 V at Nu-Link2-Pro Bridge interface.

Table 3.3-6 Bridge Interface Corresponding Pin for CAN BUS

3.3.6 PWM and Capture

The Nu-Link2-Pro provides one channel PWM function for user flexible planning. The PWM connection pins are VCC33(CON6.14 and CON6.16), VSS(CON6.18 and CON6.20) and PWM(CON6.15). Figure 3.3-6 presents how to connect the target board to use PWM function and Table 3.3-7 shows the pin corresponding to the target board.

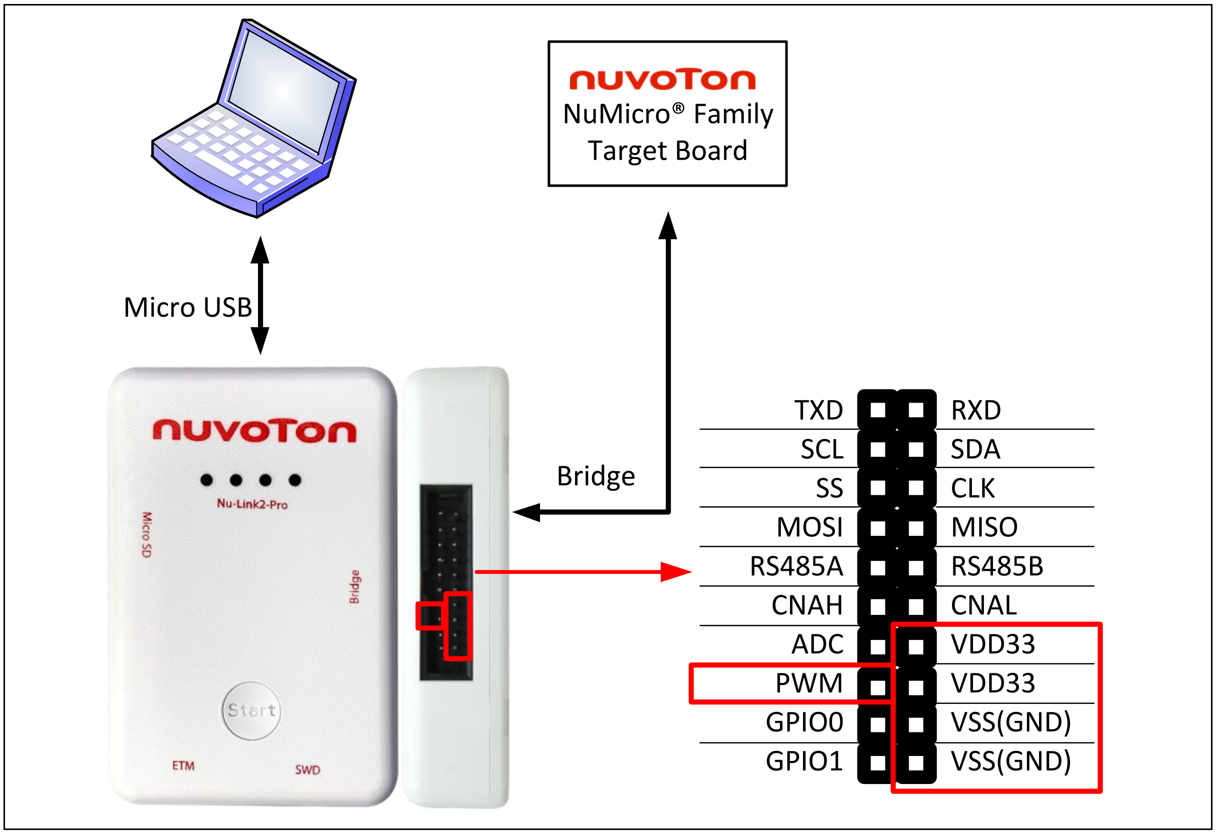


Figure 3.3-6 Bridge Interface Connection Diagram for PWM

Pin Name	Pin Number	Pin Corresponding to the Target Board
PWM	CON6.15	GPIO or Application side [1]
VCC33	CON6.14	VCC[1]
VCC33	CON6.16	VCC[1]
VSS(GND)	CON6.18	VSS(GND)
VSS(GND)	CON6.20	VSS(GND)

Note:

1. The target board power and signal only support 3.3 V at Nu-Link2-Pro Bridge interface.

Table 3.3-7 Bridge Interface Corresponding Pin for PWM

3.3.7 ADC Connection

The Nu-Link2-Pro provides one channel ADC function for user flexible planning. The ADC connection pins are VCC33(CON6.14 and CON6.16), VSS(CON6.18 and CON6.20) and ADC(CON6.13). Figure 3.3-7 presents how to connect the target board to use ADC function and Table 3.3-8 shows the pin corresponding to the target board.

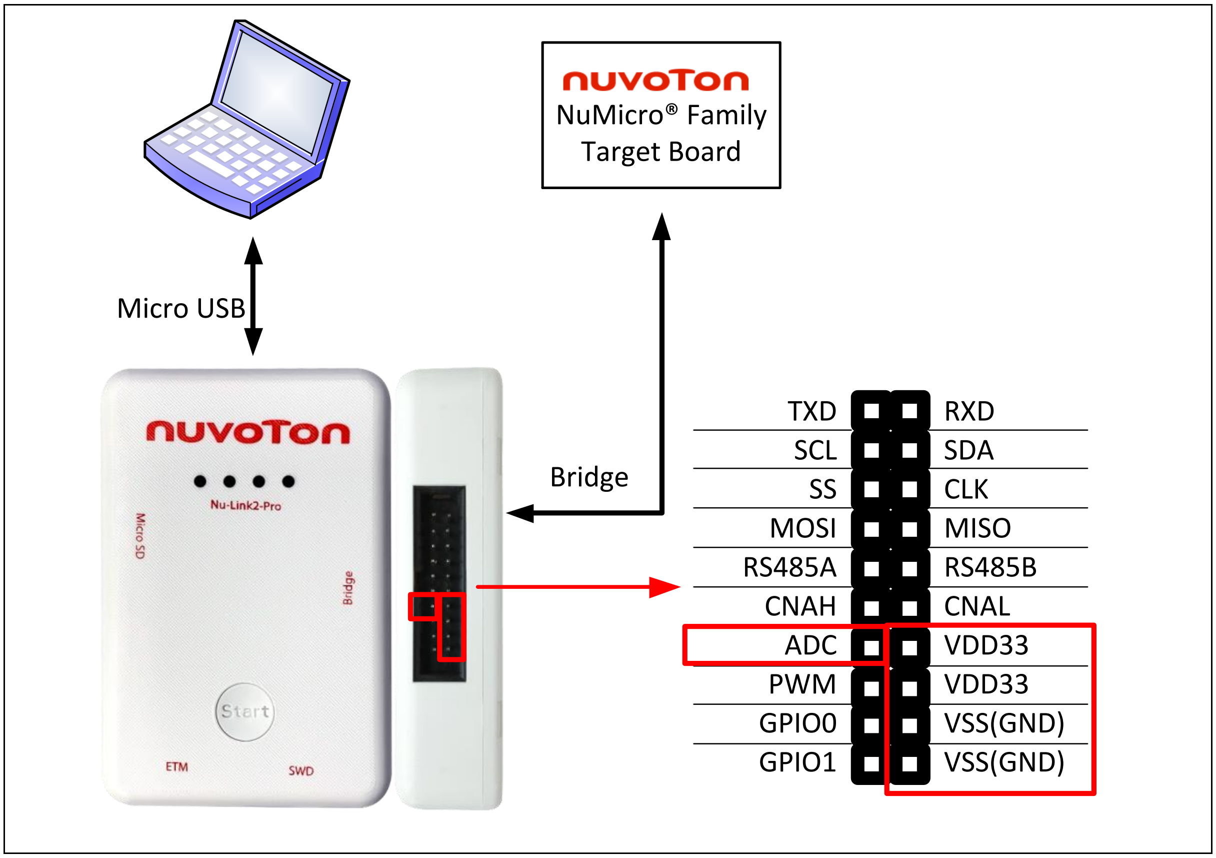


Figure 3.3-7 Bridge Interface Connection Diagram for ADC

Pin Name	Pin Number	Pin Corresponding to the Target Board
ADC[1]	CON6.13	Application side [1]
VCC33	CON6.14	VCC[1]
VCC33	CON6.16	VCC[1]
VSS (GND)	CON6.18	VSS (GND)
VSS (GND)	CON6.20	VSS (GND)

Note:

1. The target board power and signal only support 3.3V at Nu-Link2-Pro Bridge interface.

Table 3.3-8 Bridge Interface Corresponding Pin for ADC

3.3.8 GPIO Connection

The Nu-Link2-Pro provides two channel GPIO function for user flexible planning. The GPIO connection pins are VCC33(CON6.14 and CON6.16), VSS(CON6.18 and CON6.20), GPIO0(CON6.17) and GPIO1(CON6.19). Figure 3.3-8 presents how to connect the target board to use GPIO function and Table 3.3-9 shows the pin corresponding to the target board.

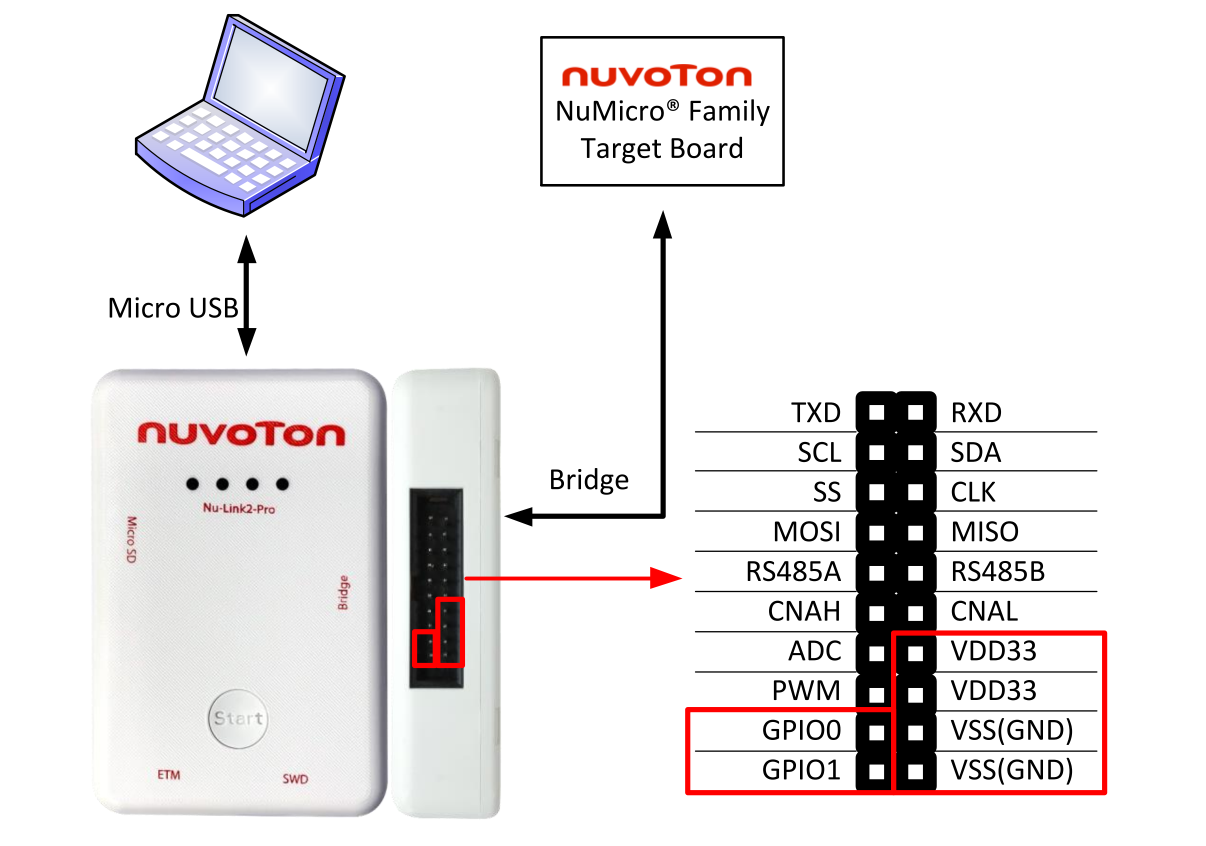


Figure 3.3-8 Bridge Interface Connection Diagram for GPIO

Pin Name	Pin Number	Pin Corresponding to the Target Board
GPIO0	CON6.17	GPIO or Application side [1]
GPIO1	CON6.19	GPIO or Application side [1]
VCC33	CON6.14	VCC [1]
VCC33	CON6.16	VCC [1]
VSS(GND)	CON6.18	VSS(GND)
VSS(GND)	CON6.20	VSS(GND)

Note:

1. The target board power and signal only support 3.3V at Nu-Link2-Pro Bridge interface.

Table 3.3-9 Bridge Interface Corresponding Pin for GPIO

3.4 ETM Interface Pin Definition and Function Connection

Table 3.4-1 shows ETM interface pin definition and description. The Nu-Link2-Pro provide a ETM interface connector with a 50-mil 20-pin cable. The ETM interface supports ETM and SWD function. The following sections will introduce the definition of the ETM interface pin and the connected of each function.

Pin Name	Pin Number	Pin Description
VCC33	CON3.1	Target Board voltage supply. The Nu-Link2-Pro ETM VCC only supports 3.3V.
SWDIO	CON3.2	Serial Wired Debugger Data pin
GND	CON3.3	Ground
SWDCLK	CON3.4	Serial Wired Debugger Clock pin
GND	CON3.5	Ground
NC	CON3.6	NC
KEY	CON3.7	A key pin to properly orient the connector.
NC	CON3.8	NC
GND	CON3.9	Ground
/RESET	CON3.10	IC reset pin, Nu-Link2-Pro will automatically reset the target IC during the programming process.
NC	CON3.11	NC
TRACECLK	CON3.12	ETM trace clock pin.
NC	CON3.13	Ground
TRACEDATA[0]	CON3.14	ETM trace data output pin.
GND	CON3.15	Ground
TRACEDATA[1]	CON3.16	ETM trace data output pin.
GND	CON3.17	Ground
TRACEDATA[2]	CON3.18	ETM trace data output pin.
GND	CON3.19	Ground
TRACEDATA[3]	CON3.20	ETM trace data output pin.

Table 3.4-1 ETM Interface Pin Definition and Description

3.4.1 SWD Connection

The ETM interface provides SWD function for IC programming and Debugging. Figure 3.4-1 presents how to connect the target board to use SWD function. In addition, please pay attention to the behavior and do not program or debug at the same time with the SWD interface; otherwise, an error will occur.

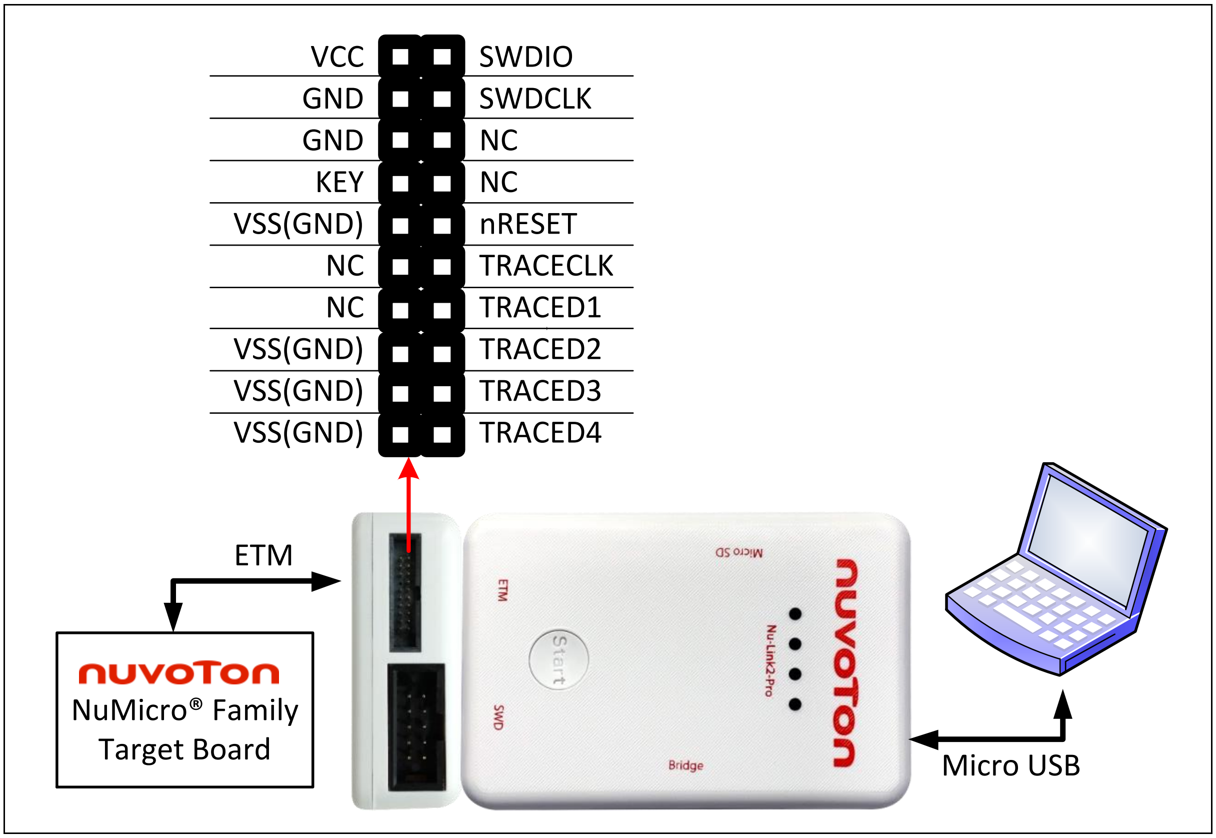


Figure 3.4-1 ETM Interface Connection Diagram for SWD and ETM Function

3.4.2 ETM Connection

The ETM interface provides ETM function for capturing execution steps of microprocessor on the target board, and ETM will display them a readability format. Figure 3.4-1 presents how to connect the target board to use ETM function.

3.5 ICP Offline Programming Function Connection

The Nu-Link2-Pro provides three kinds storage interface for Nu-Link2-Pro ICP offline programming. The user can save the bin file to USB Flash drive, Micro SD card or SPI Flash for offline programming. The priority of reading from these three storage is USB Flash drive > Micro SD card > SPI Flash. Figure 3.5-1 presents how to connect the target board to use GPIO function.

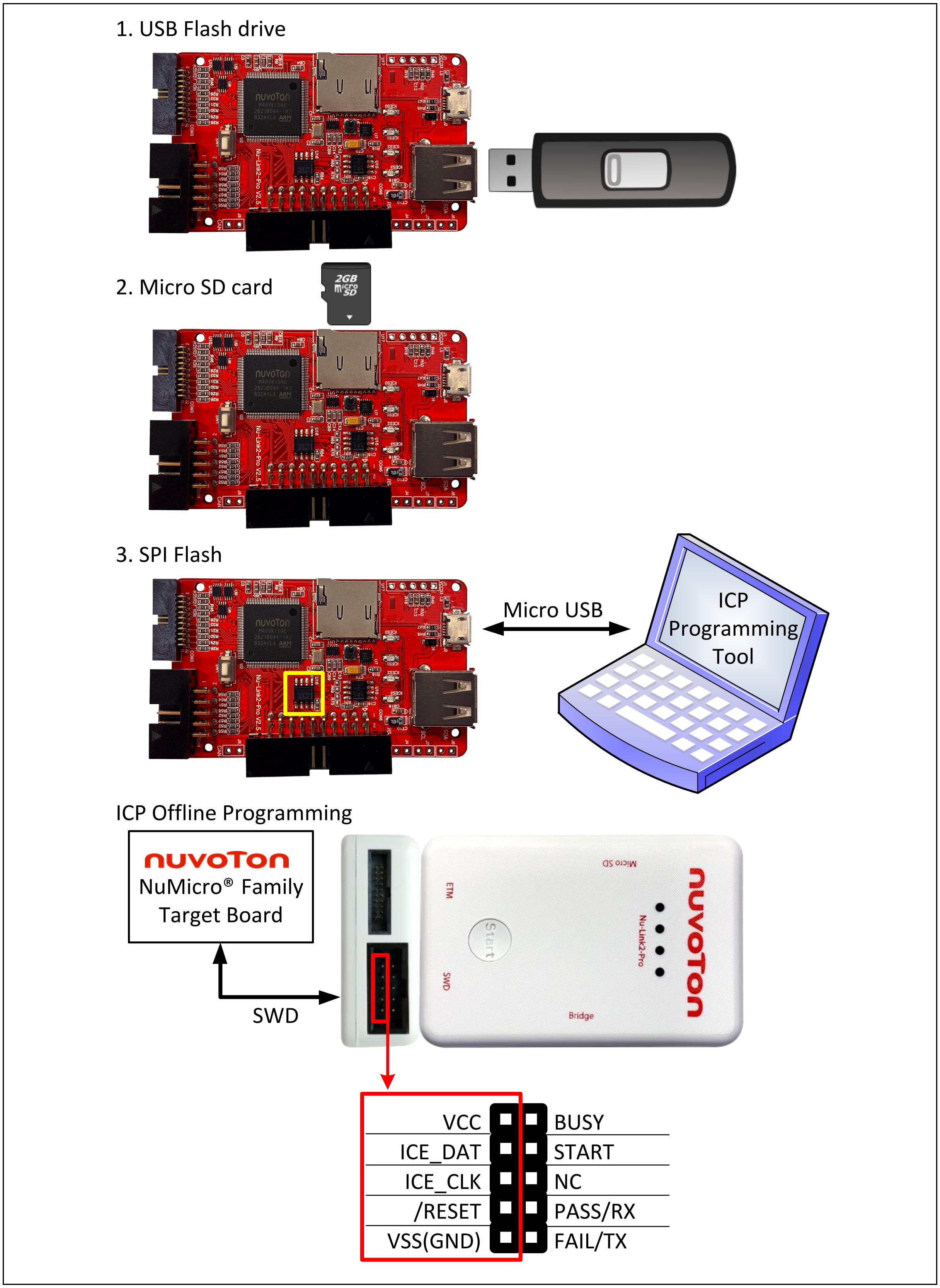


Figure 3.5-1 ICP Offline Programming Illustration of SWD Interface

4 DEBUGGING AND PROGRAMMING

4.1 Debugging

This section briefly describes the debugging function supported by the Nu-Link2-Pro. For more details, please refer to the related user manuals.

4.1.1 Debug Mode

The Nu-Link2-Pro supports debugging for the NuMicro® Family chips based on the SWD signal interface. The third-party tools that support using the Nu-Link2-Pro for chip debugging include Keil MDK, IAR EWARM, and NuEclipse GCC. Some more functions supported in Debug mode are described as follows.

4.1.2 Breakpoints

In Debug mode, the user can add breakpoints in the code for debugging. During the real-time simulation of the Nu-Link2-Pro, the chip simulation will be stopped at a specific breakpoint. Figure 4.1-1 shows the breakpoint settings in Keil MDK Debug mode. The red labels on lines 052 and 059 indicate the breakpoints inserted; the yellow arrow refers to the code to be executed next and shows the register value of Program Counter (PC) (i.e. "**R15(PC)=0x00000D04**" in the **Registers** pane in Figure 4.1-1).

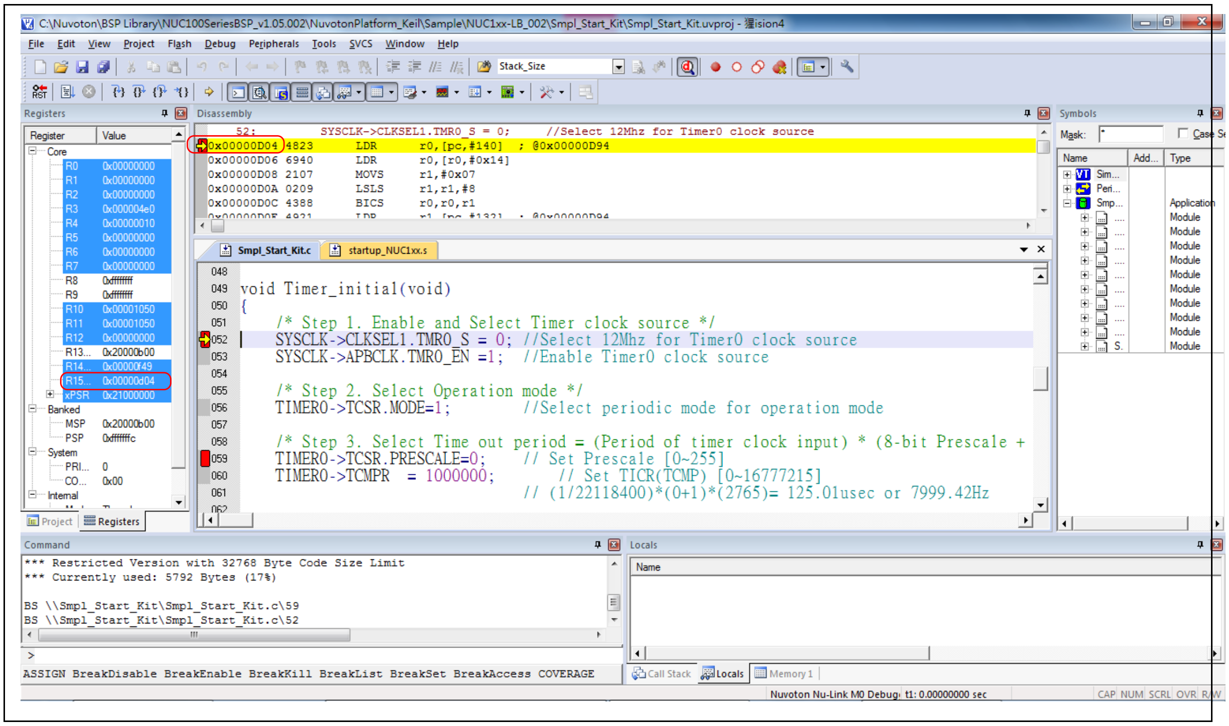


Figure 4.1-1 Setting Breakpoints in Keil MDK Debug Mode

4.1.3 System Viewer

The System Viewer can be used to display the register content in a target chip and manipulate the registers. Take Keil MDK Debug mode for example, invoke the **View → System Viewer** command and select a register from the "function register list" (e.g. ADC, CAN, CLK, etc.) to open the System Viewer of the selected register, as shown in Figure 4.1-2.

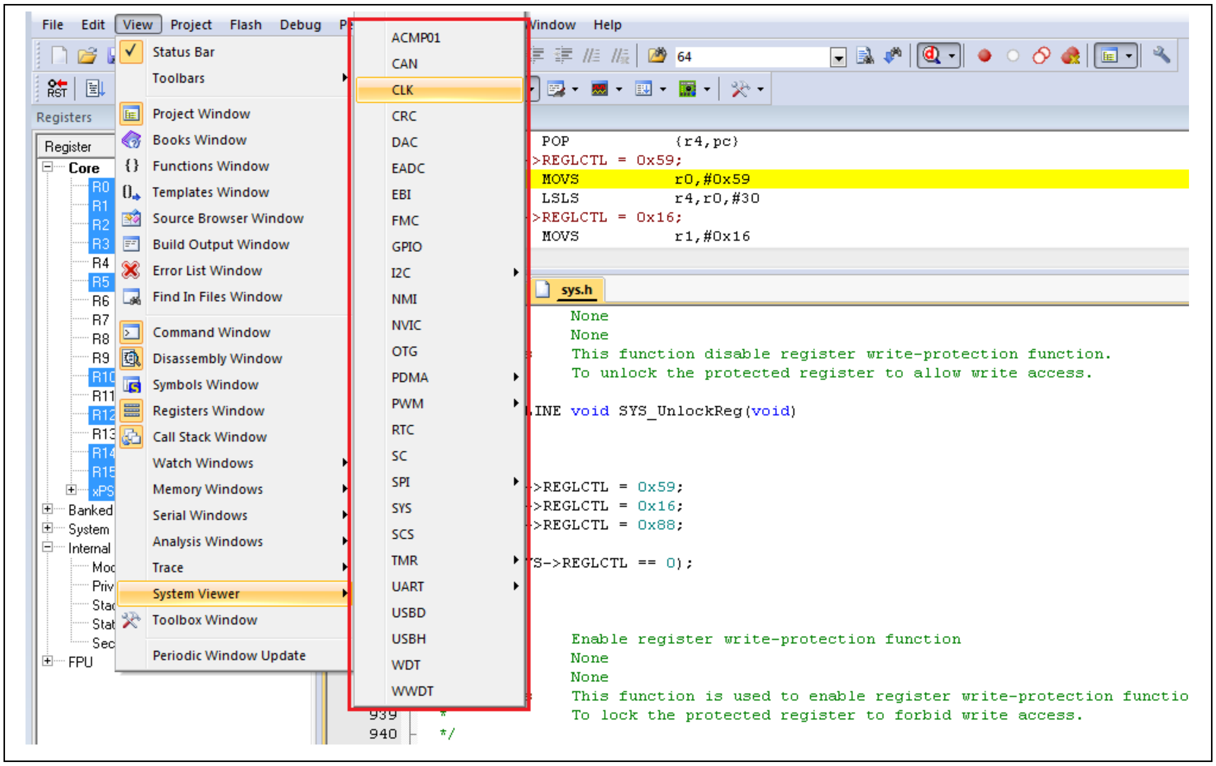


Figure 4.1-2 System Viewer Control Related Options in Keil MDK Debug Mode

The System Viewer for CLK is shown in Figure 4.1-3, where the lower side shows the register address and description, and the upper side shows the register value.

Detailed Operation:

Double-clicking a “register name” will open the register control details, as shown in Figure 4.1-3.

The “register value” can be modified directly. The Nu-Link2-Pro will modify the content of the target chip.

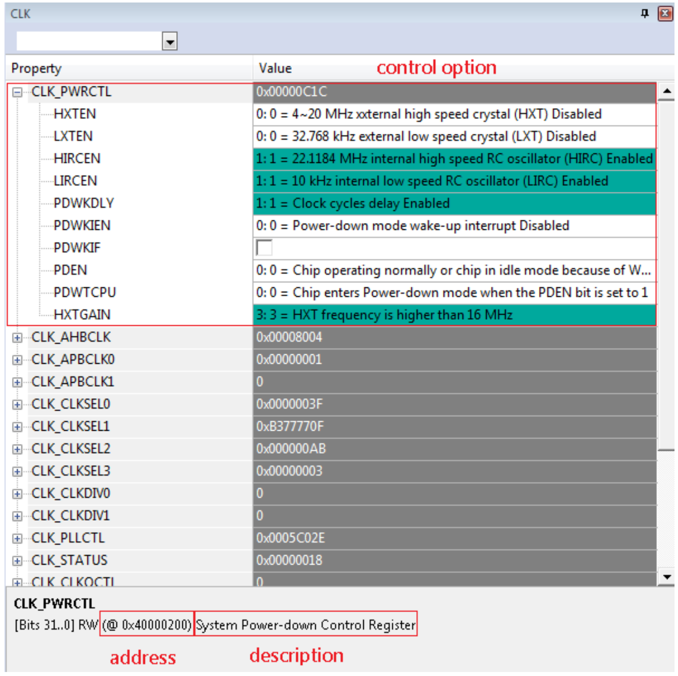


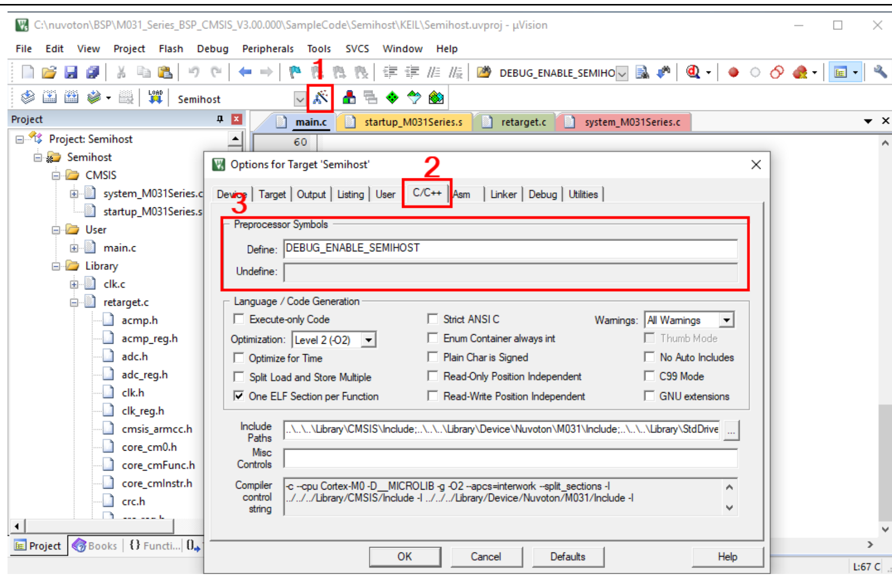
Figure 4.1-3 System Viewer in Keil MDK Debug Mode

4.1.4 Semihost

When using the Semihost function, the message of the NuMicro® Family microcontroller can be output through UART to the debug window by the Nu-Link2-Pro. That is, the message is output without the GPIO. Figure 4.1-4 shows the debug messages in the "UART #1" form, which are the messages output by the Nu-Link2-Pro.

Follow the steps below to use the Semihost (taking the Keil MDK and M031 series as example).

Step 1: Invoking **Project → Options for Target 'Semohost'… →** C/C++, and paste "DEBUG_ENABLE_SEMIHOST" in the Define field to enable semihost.



Step 2: Invoke **Rebuild** to rebuild a project and enter Debug mode.

Step 3: In Debug mode, invoke **View → Serial Windows → UART #1,** as shown in Figure 4.1-4.

Step 4: Press **F5** to program the target chip, and the debug messages are output to the *UART #1* form3

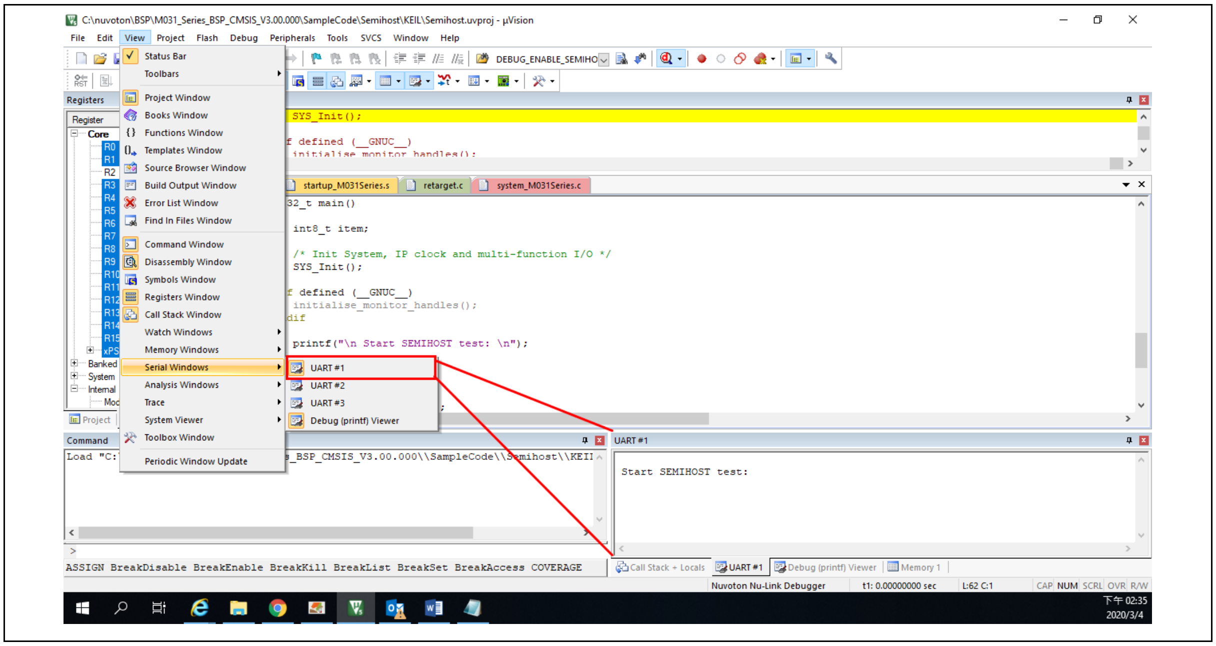


Figure 4.1-4 Semihost Options in Keil MDK Debug Mode

4.1.5 **Embedded Trace Macrocell (ETM)**

Nu-Link2-Pro supports the Embedded Trace Macrocell function that can show every single executed instruction in the current application to PC, For detailed settings and usage, please refer to section 5.2.2.3.

4.2 Programming

This section will briefly describe the programming function supported by the Nu-Link2-Pro. For more details, please refer to the related user manuals.

4.2.1 ICP Online Programming

Online Programming means that the Nu-Link2-Pro can download the firmware of the NuMicro® Family single chip to the target chip through software programs, as shown in Figure 4.2-1.

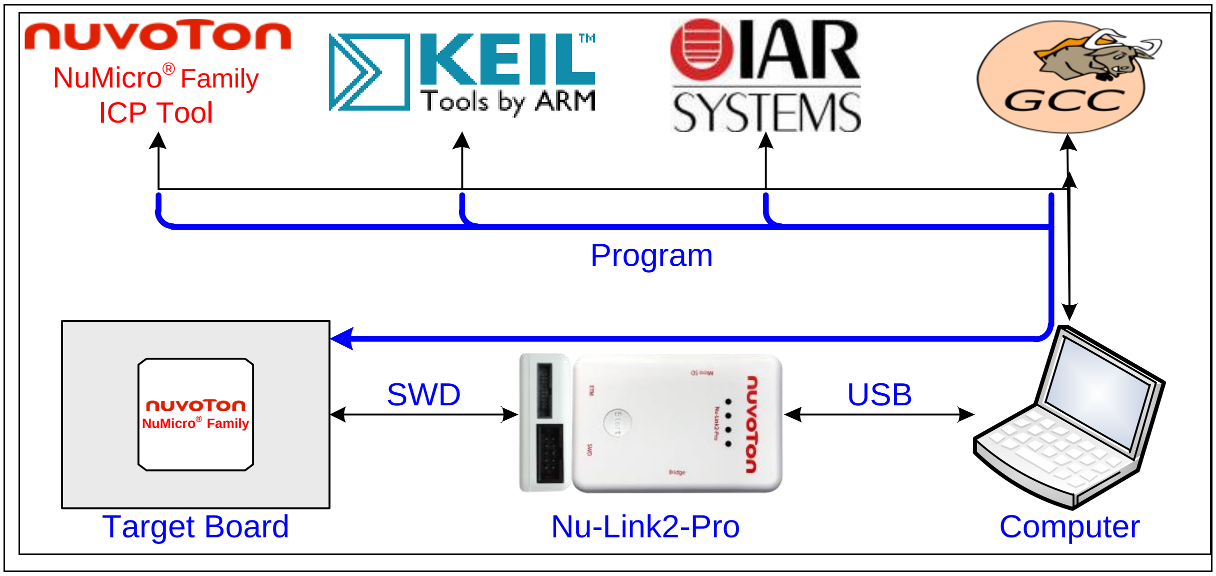


Figure 4.2-1 Online Programming Flow Diagram

4.2.2 ICP Offline Programming

Offline Programming means that the Nu-Link2-Pro can update the firmware of the NuMicro® Family single chip directly without accessing software programs (as shown in Figure 4.2-2). Offline programming is useful for mass production since the original code or firmware file does not need to be delivered and only the Nu-Link2-Pro is needed for mass production. In addition, the Nu-Link2-Pro supports "Limited Offline Programming," which can effectively control the authorized number of the firmware. For details, please refer to the ICP Tool User Manual.

Nu-Link2-Pro has three interfaces to download the offline data for offline download.

1. USB flash drive.
2. SD card.
3. Embedded SPI flash of Nu-Link2-Pro (offline download usage is same as Nu-Link).

Please follow the steps below to use USB flash drive or SD card interface for offline download:

1. Use **Tool -> Create Offline USB/SD File** on the menu bar of ICP Tool to save **NuLink2.us** file and drag and drop the file into USB flash drive or SD card.
2. Plug USB flash drive or SD card into Nu-Link2-Pro.

Pressing the button on the Nu-Link2-Pro will switch the Nu-Link2-Pro to offline download mode and start to download the offline data to target chip immediately.

The Nu-Link2-Pro also supports Control Bus so that can use the automatic IC programming function during mass production. For details, please refer to the section 6.3.

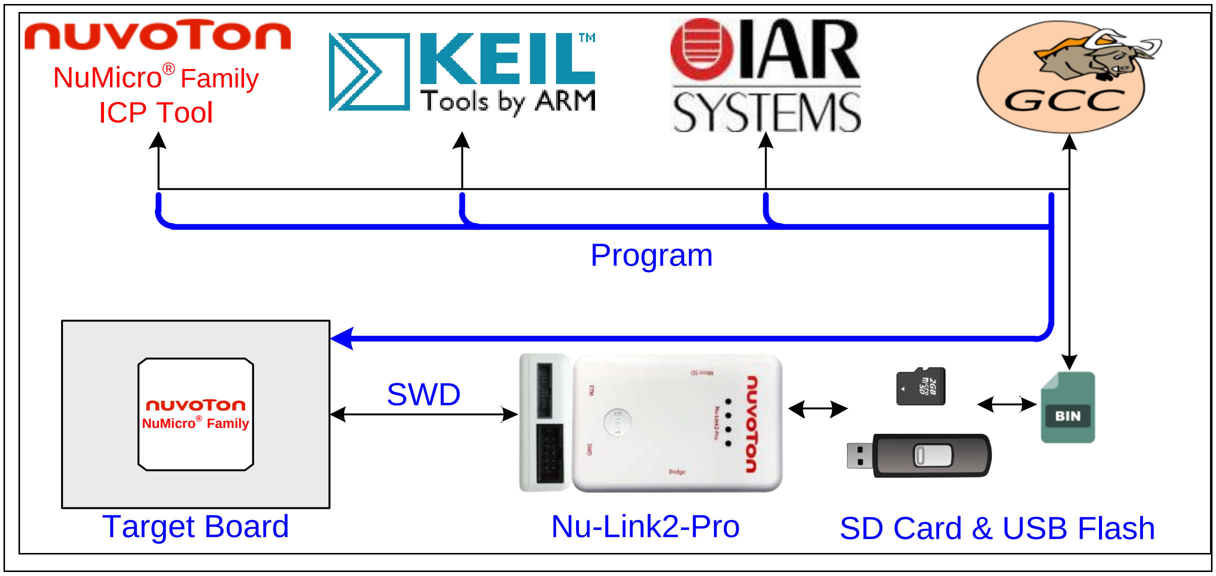


Figure 4.2-2 Offline Programming Flow Diagram

4.2.3 ISP Online Programming

The ISP tool currently only supports online programming and the programming interface supports UART, I^2C, SPI, RS-485 and CAN BUS, Table 5.2-2 show the connection method of each interface and please refer to the section 5.2.5 for ISP online programming details.

4.2.4 Software Serial Number (SN)

The Software Serial Number (SN) function provided by the ICP Tool enables users to specify the value in the “**Increase SN from**” and “**Write address in flash**” fields for the target chip during online/offline programming. Take the M031 series chip for example, the user can specify a set of “Increased Serial Number (SN)” and “Write Address” to any of APROM, LDROM, and Data Flash, and the written Serial Number (SN) will be automatically incremented (as shown in Figure 4.2-3).

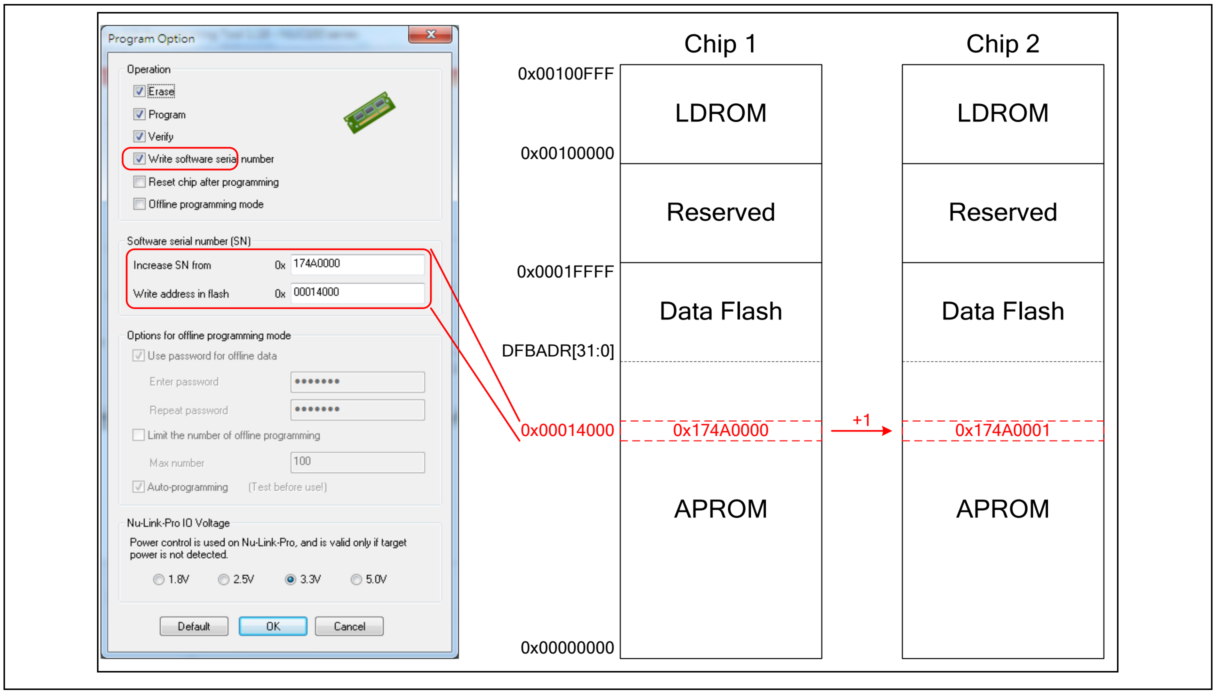


Figure 4.2-3 Software Serial Number (SN) Settings

4.3 Wide Voltage Programming

The Nu-Link2-Pro supports the wide voltage programming function, by which the development software tool can adjust the SWD port voltage as 1.8 V, 2.5 V, 3.3 V, or 5.0 V. As shown in Figure 3.2-1, the pins that can be controlled include VCC, ICE_DAT, ICE_CLK, and /RESET.

4.4 Installing the Nu-Link2-Pro Driver

The Nu-Link2-Pro supports a variety of functions and third-party software tools (e.g. Keil MDK and IAR EWARM). After the software programs are installed, the drivers are also required. You can use the following links: Nu-Link_Keil_Driver for Keil MDK and Nu-Link_IAR_Driver for IAR EWARM to install the latest version. For details about software setup, please refer to section 5.2.

5 INSTALLATION AND SETUP

This chapter introduces how to connect the Nu-Link2-Pro to a computer, and how to set the third-party tool to use the Nu-Link2-Pro as a debugger and a programmer.

5.1 Connecting to the Nu-Link2-Pro

As shown in **Figure 5.1-1**, the Nu-Link2-Pro is a bridge between an USB and the SWD interface, by which software tools can debug and program the target chip through an USB. The user can plug the Nu-Link2-Pro into an USB port of a PC directly or connect using the USB connector. About connection method please refer to section 3.2 for details.

Through a SWD port, the Nu-Link2-Pro can supply power (1.8 V, 2.5 V, 3.3 V, or 5.0 V) to a target circuit board. The maximum is 5 V/500 mA. Refer to Table 6.1-1 for detailed specifications.

Figure 5.1-1 Nu-Link2-Pro Connection Diagram

SWD Connector:

The SWD connector, which can be applied to all of the NuMicro® development tools and evaluation boards, is a 100 mil (2×5) female header, as shown in the left of Figure 3.2-1.

5.2 Software Setup

This section briefly describes required software settings for connecting to the Nu-Link2-Pro. For detailed software operation, refer to the related user manuals.

5.2.1 ICP Tool

(1) Download and install Nuvoton NuMicro® ICP Programming Tool.

(2) Open the ICP Tool, specify the **UI language** and **target chip**, and then click **Continue**, as shown in Figure 5.2-1.

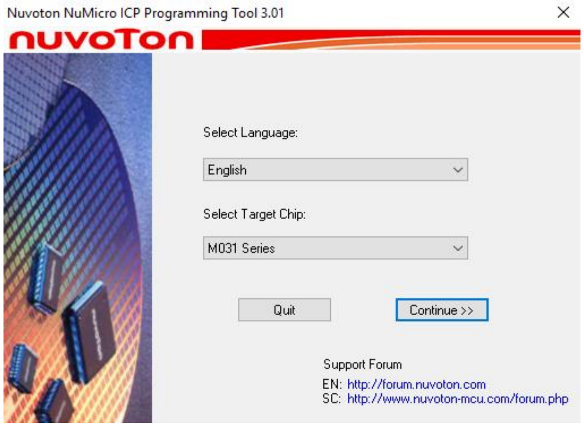


Figure 5.2-1 Startup Screen of ICP Tool

(3) In the ICP Tool window, the connection status is shown as "Disconnected" since the ICP tool has not been connected with the Nu-Link2-Pro, as shown in Figure 5.2-2.

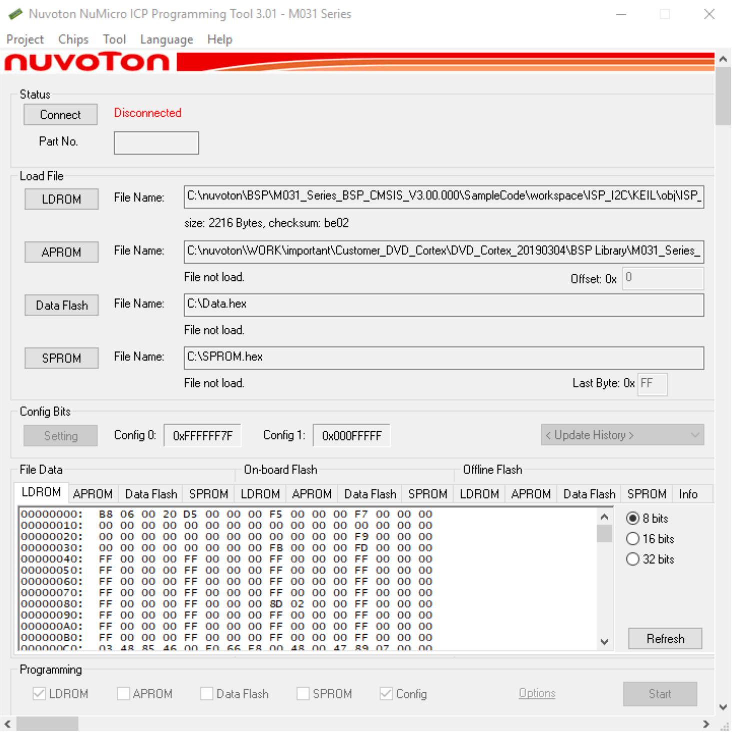


Figure 5.2-2 ICP Tool Main Window

(4) And then click the Connect button. Go to (5) if more than one Nu-Link2-Pro are connected with the host. Go to (6) if only one Nu-Link2-Pro is connected with the host.

(5) If two Nu-Link Debugger and Programmers have been connected with the computer, a message appears and asks to select one from the two adapters. Clicking **OK** will connect the selected adapter with the host, as shown in Figure 5.2-3. When a Nu-Link2-Pro is selected for connection, the Status LED starts blinking.

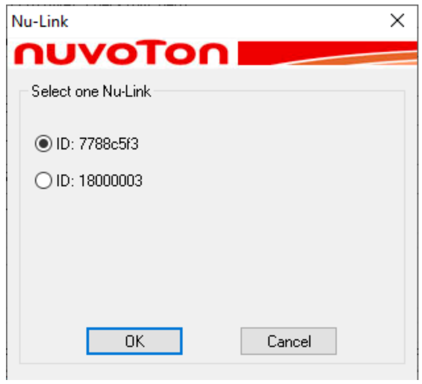


Figure 5.2-3 Select One Nu-Link2-Pro

(6) Click **Option** in the **Programming** section of the ICP Tool Window to open the **Program Option** form, as shown in Figure 5.2-4**.**

(7) In the **Nu-Link Pro IO Voltage** section, specify the power voltage of the SWD port for the target chip, and then click **OK**. To use the offline programming function, the Offline Programming mode option needs to be selected, as shown in Figure 5.2-4.

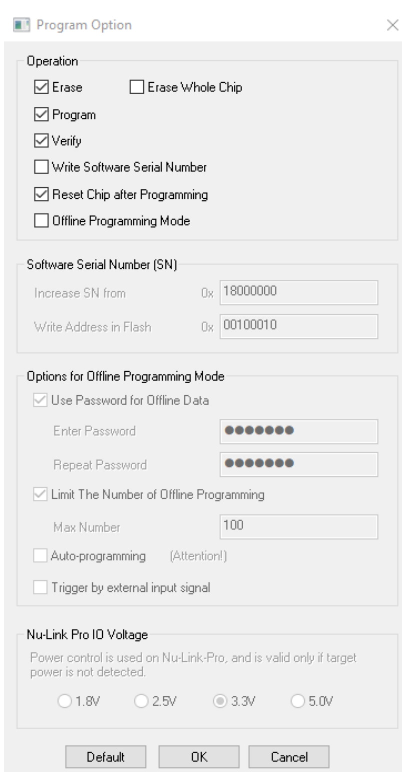


Figure 5.2-4 ICP Tool Programming Options

(8) After the **Connect** button is clicked, the ICP Tool will be connected with the Nu-Link2-Pro, and a

SWD port will be detected. Figure 5.2-5 shows that the ICP Tool has been connected with the Nu-Link2-Pro and a target chip is detected. At this time, the user can start programming the target chip.

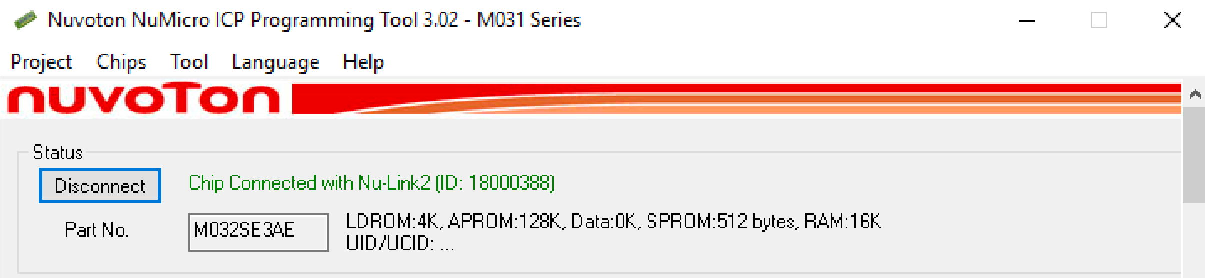


Figure 5.2-5 Nu-Link2-Pro Connected with a Target Chip Detected

(9) Figure 5.2-6 shows that the ICP Tool has been connected with the Nu-Link2-Pro with no target chip detected. The ICP tool will continue detecting the target chip until the **Stop Check** button is clicked. At this time, the user cannot program any chip, but can use the offline programming to save the offline programming information in the Nu-Link2-Pro.

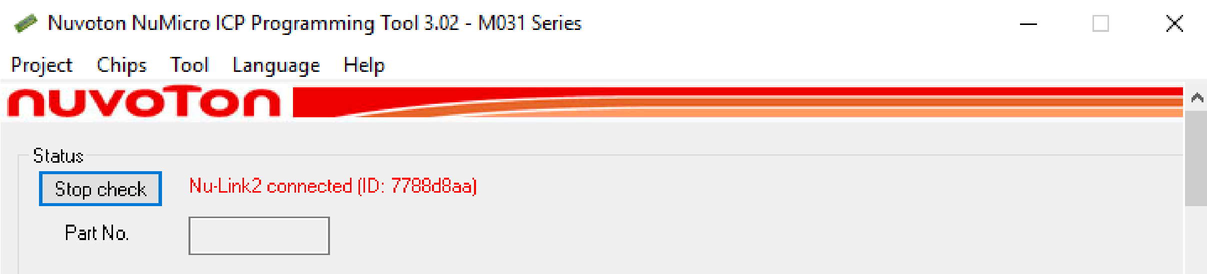


Figure 5.2-6 Nu-Link2-Pro Connected with No Target Chip Detected

(10) Click the **Disconnect** button if programming is not needed (as shown in Figure 5.2-5). Or click the **Stop Check** button to disconnect the ICP Tool with the Nu-Link2-Pro and leave the Nu-Link2-Pro unused (as shown in Figure 5.2-6). As such, the Nu-Link2-Pro can be connected with another tool.

(11) As shown in Figure 5.2-7, select the ".bin" file you want to programming APROM and LDROM, and tick the memory location.

(12) Click the Start button to start programming.

(13) As shown in Figure 5.2-8, click the OK button in the programming completion window to complete the operation.

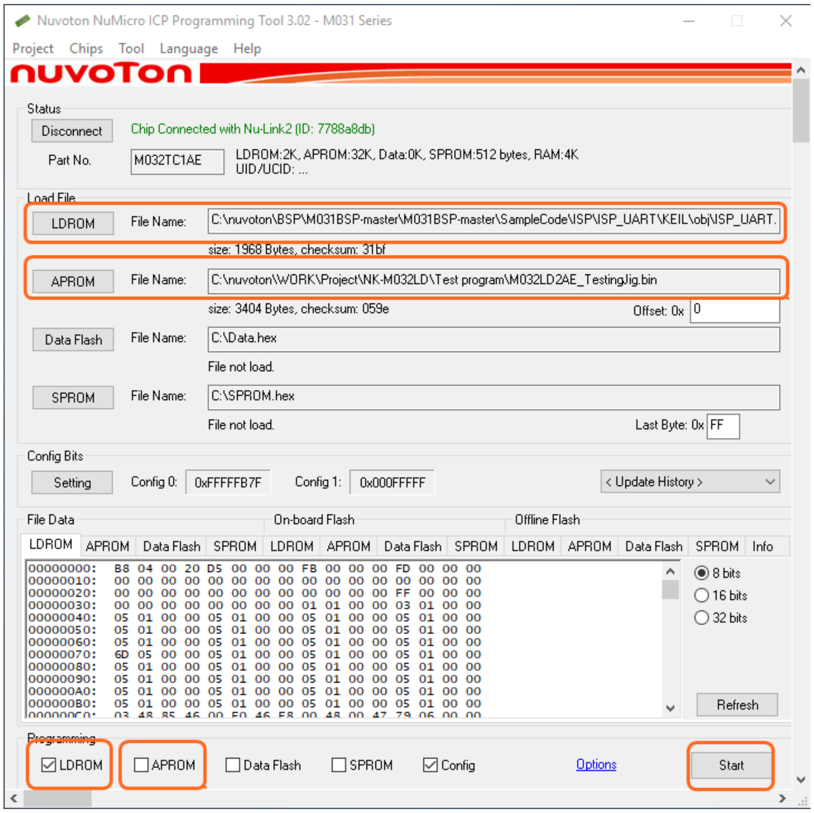


Figure 5.2-7 Programming File Selection Window

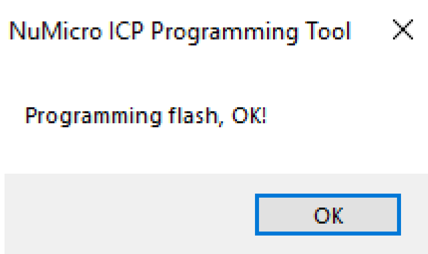


Figure 5.2-8 Programming Completion Window

5.2.2 Keil MDK

Install Keil MDK. Before setting the Nu-Link2-Pro, make sure the Nu-Link_Keil_Driver for Keil MDK has been downloaded and installed such that the Keil MDK can recognize the Nu-Link2-Pro.

(1) Double click the **Template.uvproj** to open the project.

Note: If Figure 5.2-9 warning message jumps out, please migrate to version 5 formats as shown in Figure 5.2-10. The **.uvproj** filename extension will change to **.uvprojx.t**.

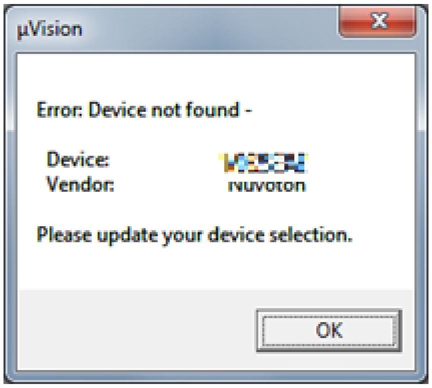


Figure 5.2-9 Warning Message of “Device not found”

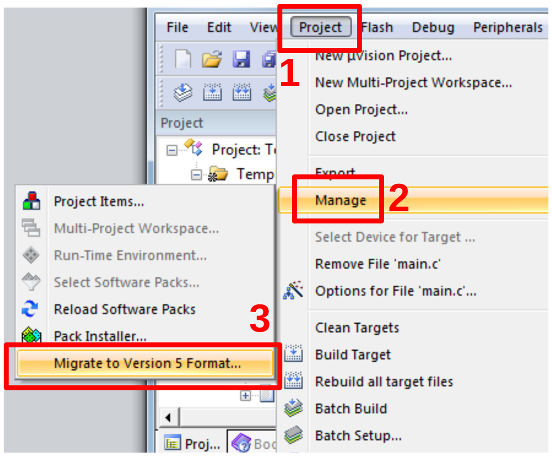


Figure 5.2-10 Project File Migrate to Version 5 Format

5.2.2.1 Debugger Settings

(2) Invoke **Project** → **Options for Target** → **Output**, and enable the **Debug Information** option, as shown in Figure 5.2-11.

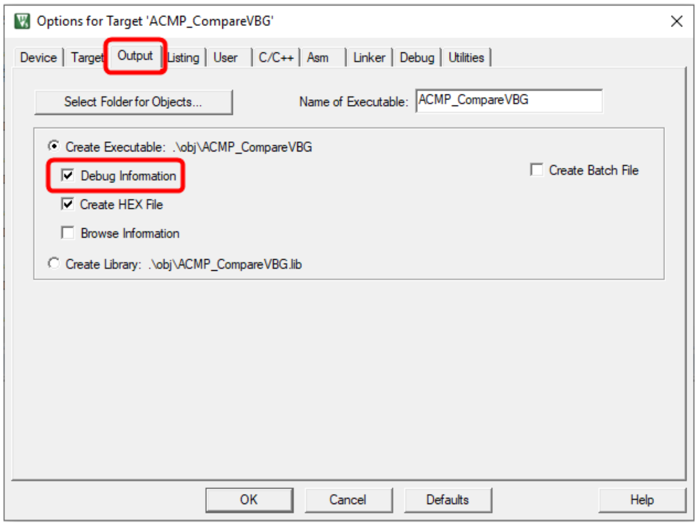


Figure 5.2-11 Enable Debug Information for Keil MDK

(3) Invoke **Project → Options for Target → Debug**, and make sure the **Use:「Nuvoton Nu-Link Debugger** option is checked, as shown in Figure 5.2-12 and Figure 5.2-13.

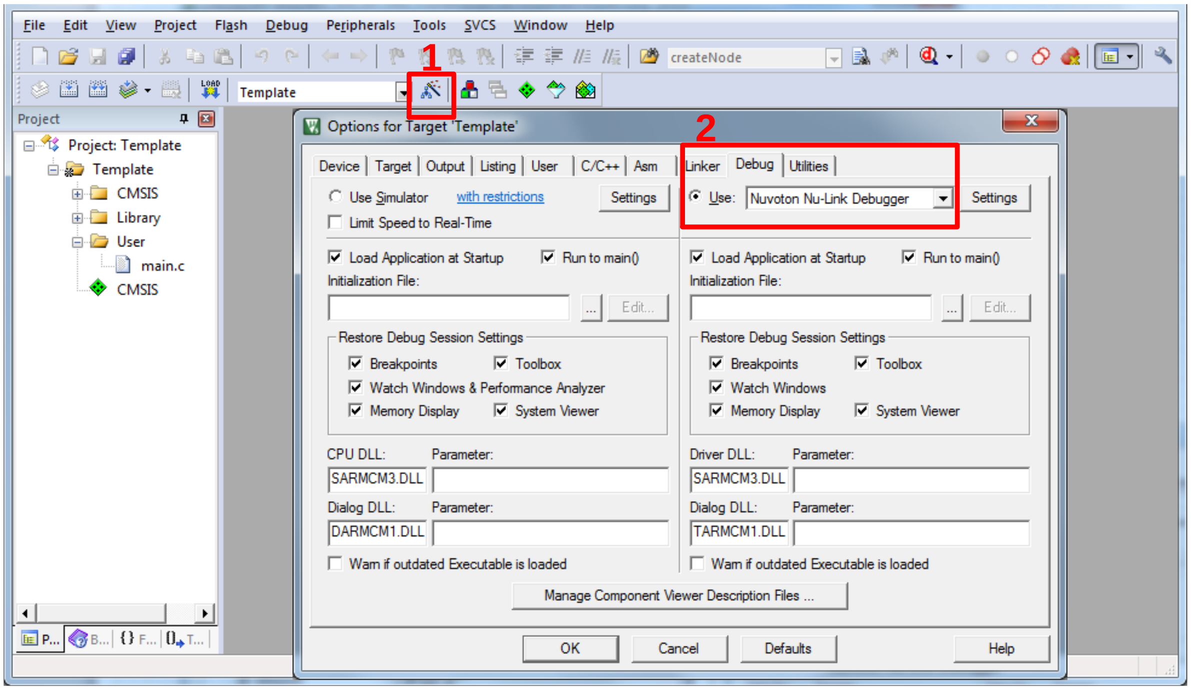


Figure 5.2-12 Debugger Setting in Options Window

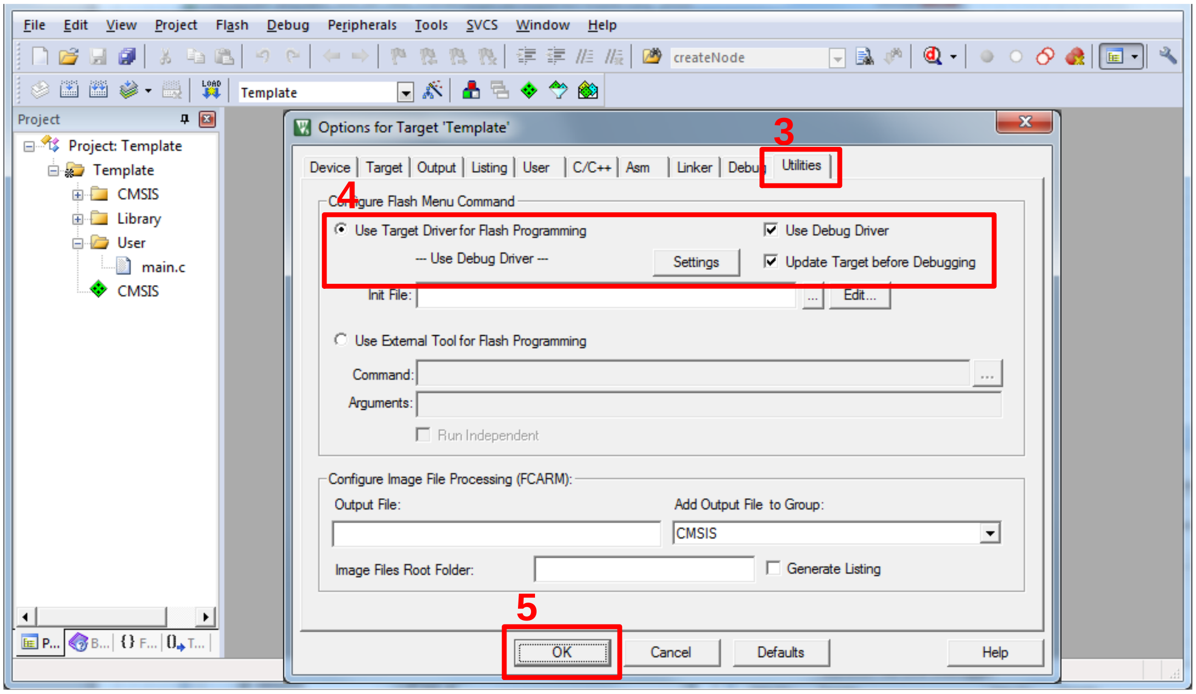


Figure 5.2-13 Programming Setting in Options Window

(4) Click the **Settings** button to open the *Debug* form, as shown in Figure 5.2-14. Refer to Table 5.2-1 for each setting description. The setting options shown in the Debug form may vary depending on the type of the Nu-Link2-Pro used.

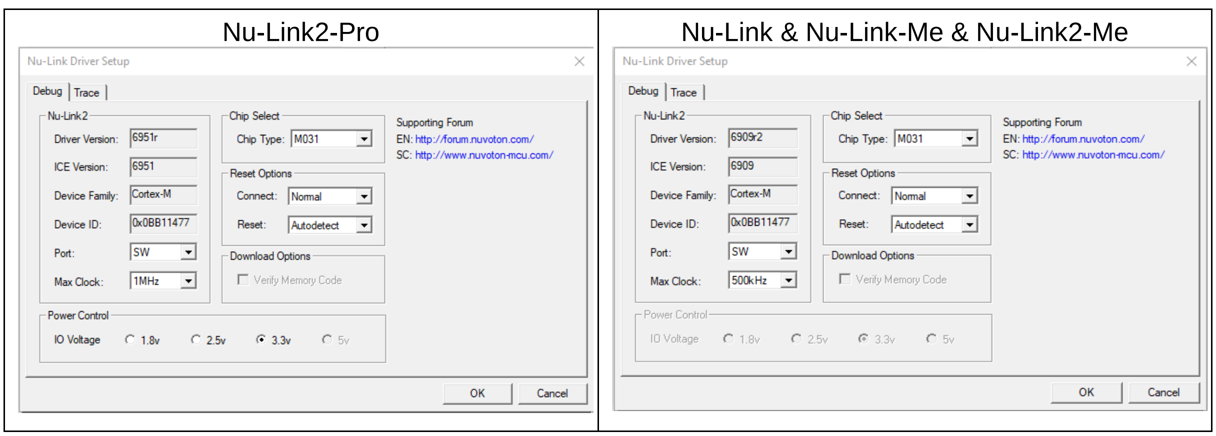


Figure 5.2-14 Nu-Link2-Pro Parameter Settings

Debug Function	Description
Driver Version	Display the Nu-Link2-Pro driver version in the host
Chip Type	Specify the Target chip type
Reset	Select **Auto detect** to reset the target chip

IO Voltage	Specify the SWD port I/O voltage for the target chip; options include 1.8 V, 2.5 V, 3.3 V, and 5 V[1]

Table 5.2-1 Debugger Function Settings Description

[1] Nu-Link2-Pro will automatically determine the target chip support voltage. If the voltage only supports 3.3 V, it will automatically set to 3.3 V power supply.

5.2.2.2 Programmer Settings

(5) Invoke **Project** → **Options for Target** → **Utilities**, select “**Nuvoton Nu-Link Debugger**” when the **Use Target Driver for Flash Programming** option is enabled, and then select the **Update Target before Debugging** option, as shown in Figure 5.2-15.

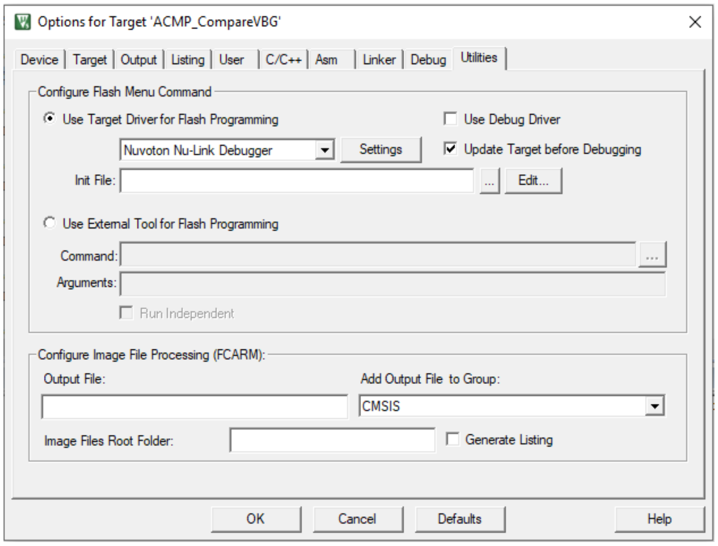


Figure 5.2-15 Keil MDK Programmer Selection

(6) Click the **Settings** button to open the *Flash Download* form, as shown in Figure 5.2-16 where the user can specify the options before or after programming with the Nu-Link2-Pro.

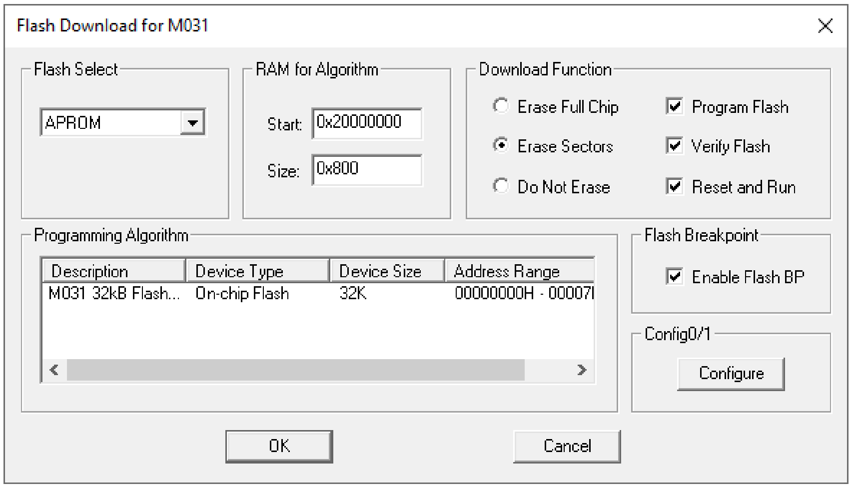


Figure 5.2-16 Nu-Link2-Pro Programming Settings

(7) Rebuild all target files. After successfully compile the project, download code to the flash memory.

Click “Start/Stop Debug Section” button can enter debug mode.

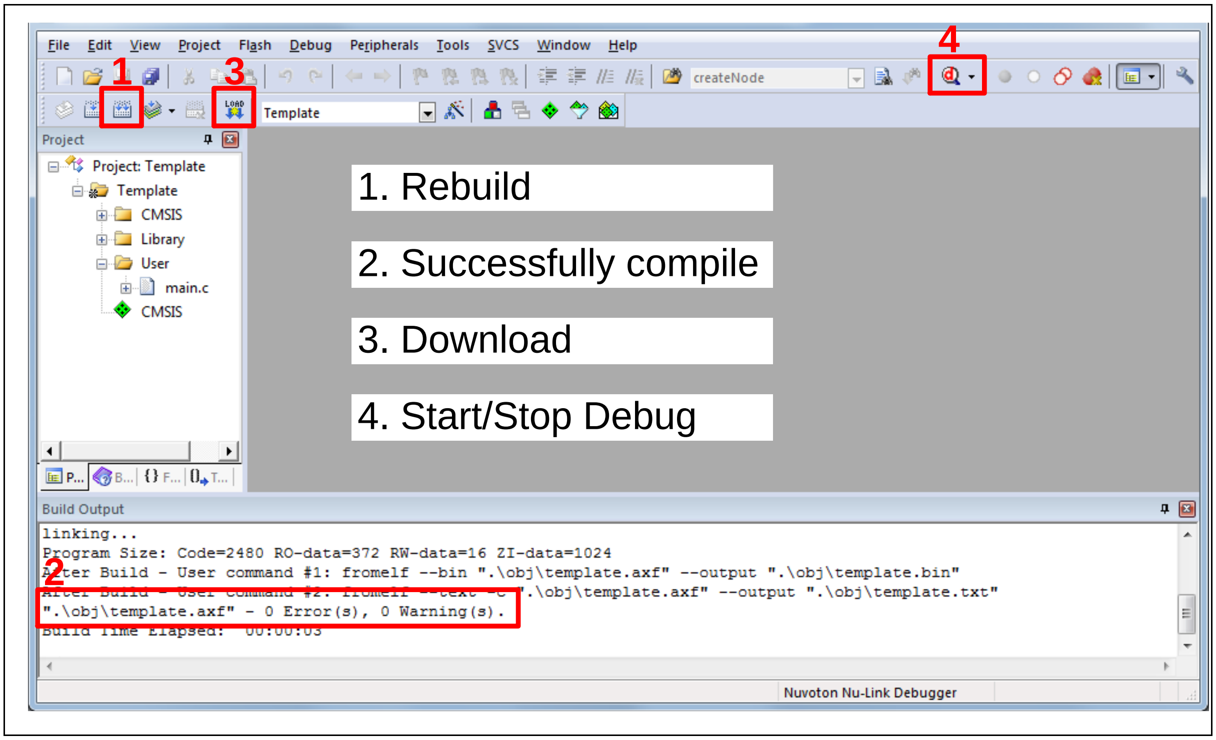


Figure 5.2-17 Compile and Download the Project

(8) Figure 5.2-18 shows the debug mode under Keil MDK. Click “Run” and the debug message will be printed out as shown in Figure 5.2-19. User can debug the project under debug mode by checking source code, assembly language, peripherals’ registers, and setting breakpoint, step run, value monitor, etc.

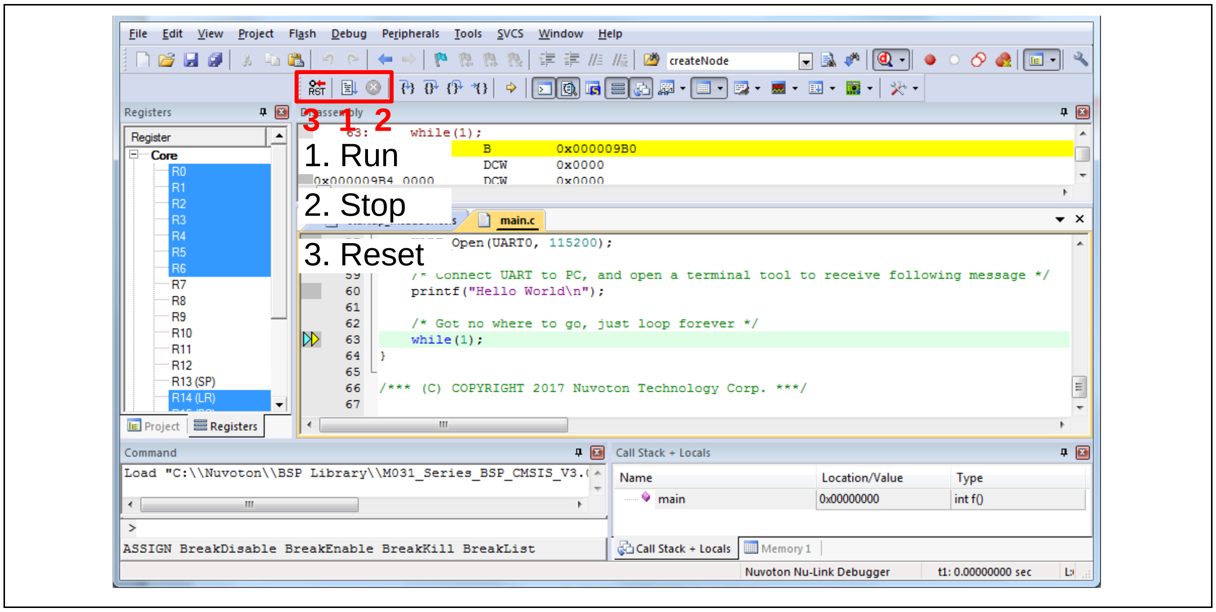


Figure 5.2-18 Keil MDK Debug Mode

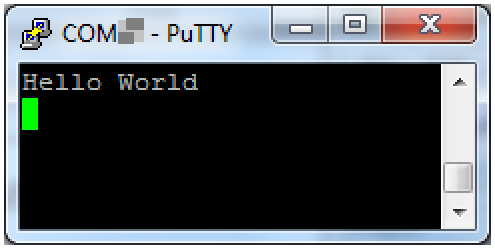


Figure 5.2-19 Debug Message on Serial Port Terminal Windows

5.2.2.3 ETM Settings

To start Embedded Trace Macrocell (ETM) tracing on Nuvoton Cortex®-M4/M23 devices, please connect to the device using the Nu-Link2-Pro with 20-pin connector and follow the steps below.

(9) Open the **Template.uvproj** in M480 BSP, as shown in Figure 5.2-20.

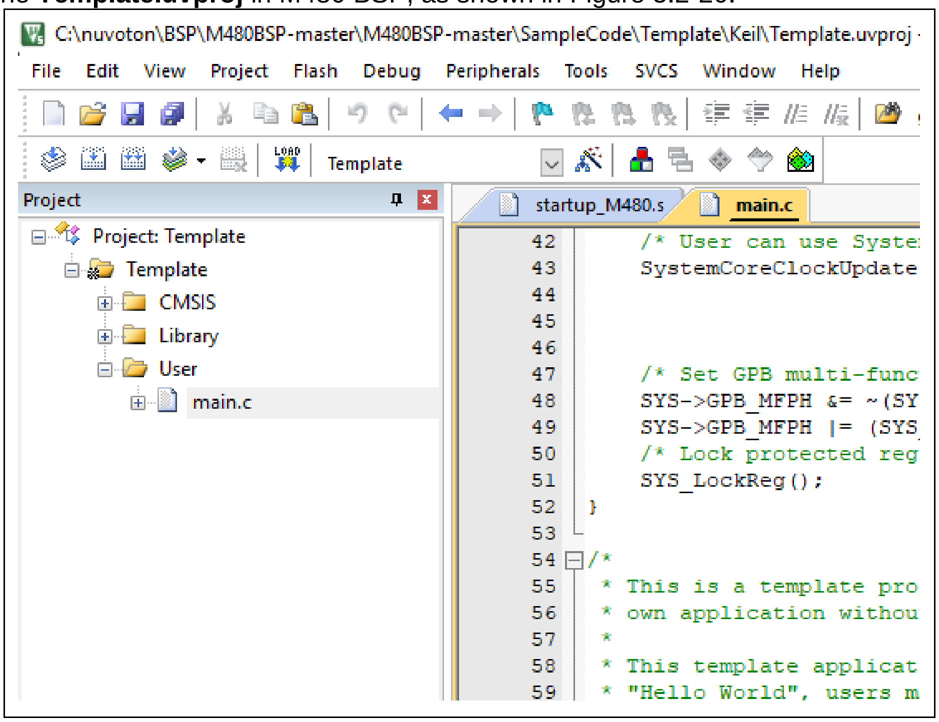


Figure 5.2-20 Open M480 project

(10) Configure the NuTrace.

- As shown in Figure 5.2-21, In debug setting dialog, select the "**Trace**" tab.
- In "**Trace Port**" select **Sync Trace Port with 4 bit data**. It is possible to use other bit sizes but best to use the largest to increase the bandwidth.
- In **Capture Mode**, specify whether trace data is collected before or after a trigger.
  - **Trace After:** Capture the trace information after the trigger point and stop capturing when trace buffer is full.
  - **Trace Before:** Capture the most recent trace information before CPU is stopped.

- **Select Trace Enable** and **ETM Trace Enable.**
- **Click OK** to save the changes.

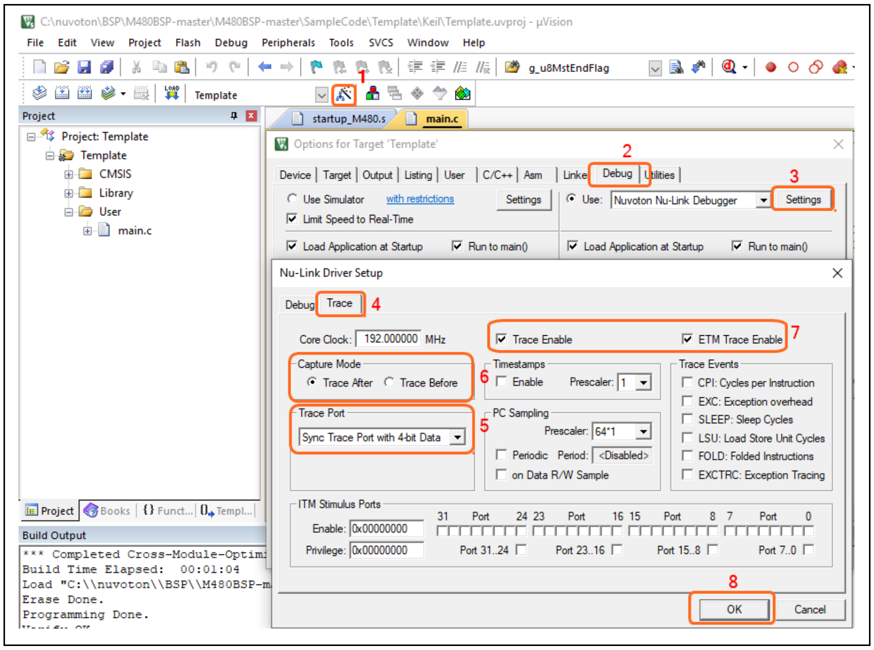


Figure 5.2-21 Trace Setup with ETM

(11) In **Initialization File**, please insert the script file to initialize the device's trace pins when starting the debugger. The following is an example script file.

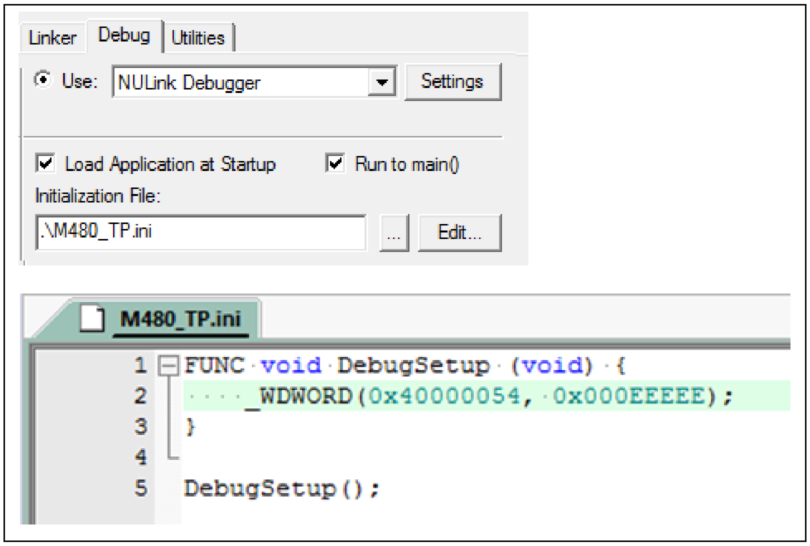


Figure 5.2-22 Initialize File for Trace Pins

Note: The Nu-Link Keil driver with the version v2.07 or later will automatically setup the trace pins when starting the debugger. The user does not need to do the above configuration.

(12) As show in Figure 5.2-23, Build and Download code to target chip.

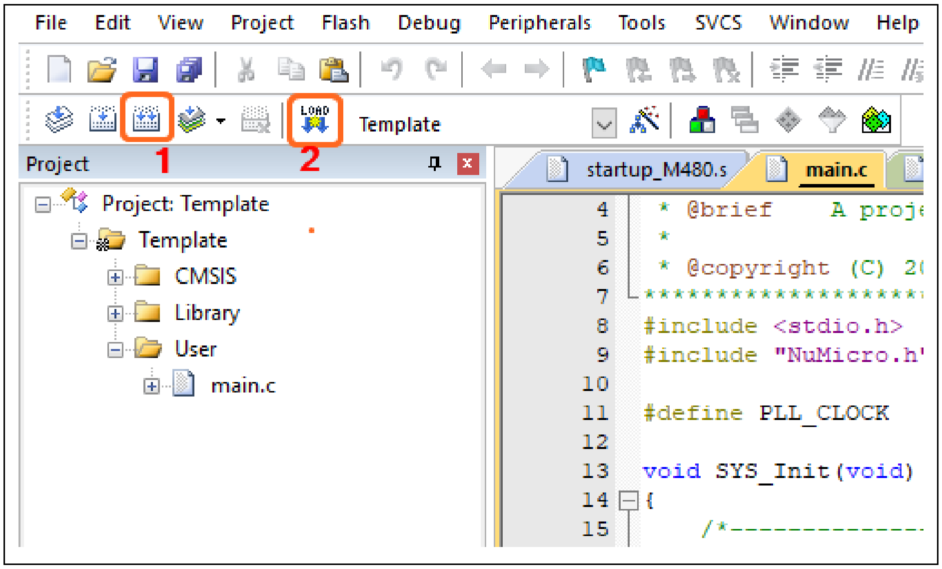


Figure 5.2-23 Build and Download Windows for ETM

(13) After doing above settings, user must start the debugger. In Debug mode, please select **Debug → NuTrace** to invoke the tracing information dialog, and it will show every single executed instruction in the current application as shown Figure 5.2-24.

Note: As show in Figure 5.2-25, It is recommended to set a break point or use single-step execution, it will be easier to track the status of the program.

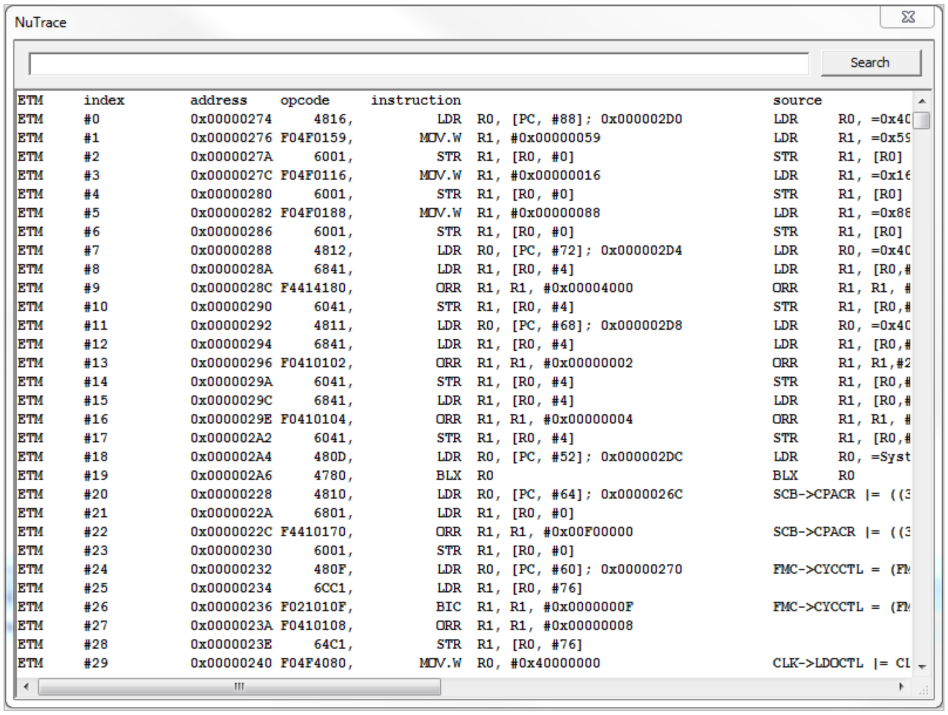


Figure 5.2-24 Tracing Information Dialog

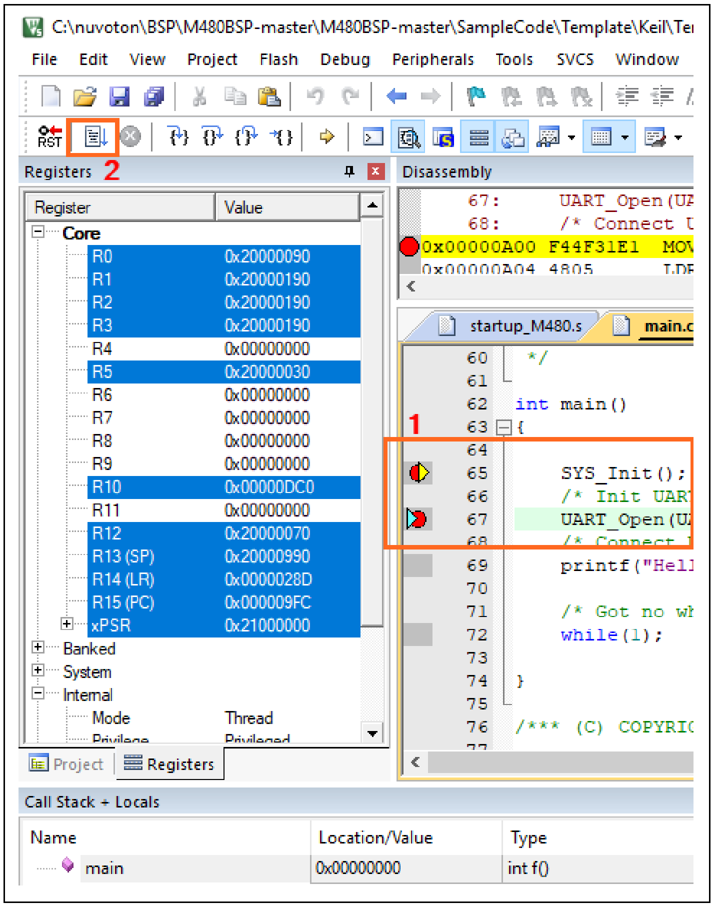


Figure 5.2-25 Breakpoint setting and Execution

5.2.3 IAR EWARM

(1) Install IAR EWARM. Make sure that Nu-Link_IAR_Driver for IAR EWARM has been downloaded and installed before setting the Nu-Link2-Pro such that the IAR EWARM can recognize the Nu-Link2-Pro.

(2) Open IAR EWARM, and open the project to be set.

(3) In the **Target** tab of the **General Options** page (through invoking **Project** → **Options**), click the button in the right of the **Device** option (make sure the **Device** option is enabled), and select "**Nuvoton** → **Nuvoton M031AE series**" as the target chip (M031AE series is this case), as shown in Figure 5.2-26 and Figure 5.2-27.

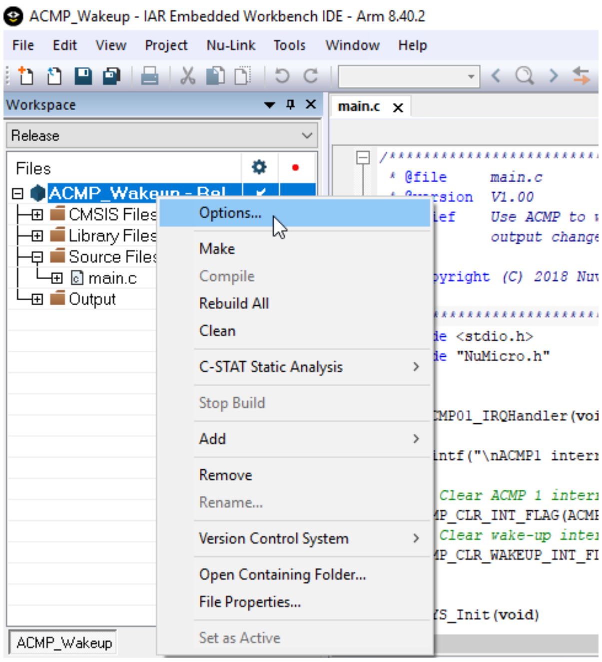


Figure 5.2-26 Options Selection

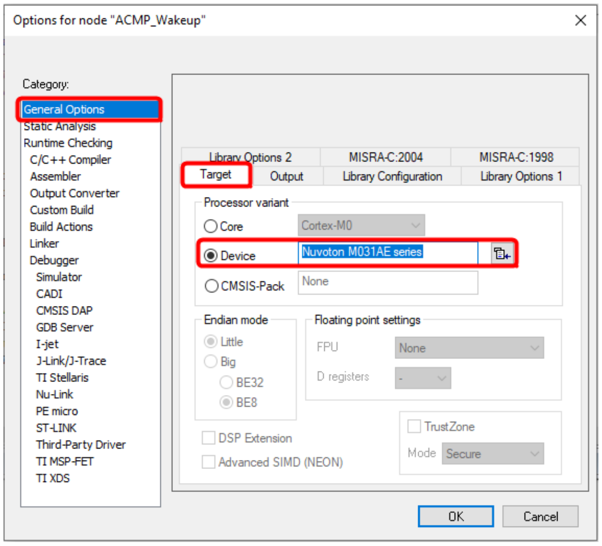


Figure 5.2-27 IAR EWARM Target Chip Selection

5.2.3.1 Debugger and Programmer Settings:

(4) In the **Setup** tab of the **Debugger** page, select **Third-Party Driver** as the driver, as shown in Figure 5.2-28.

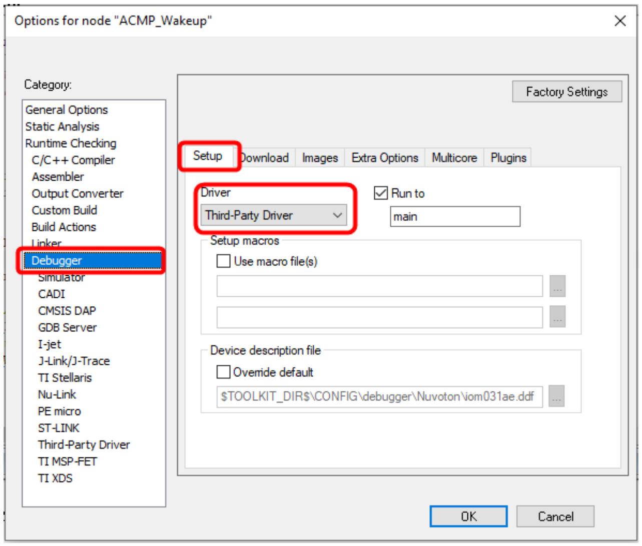


Figure 5.2-28 Set IAR EWARM as Third-Party Driver for Debugger & Programmer

(5) In the **Download** tab of the **Debugger** page, make sure that the **Use flash loader(s)** option is selected, as shown in Figure 5.2-29.

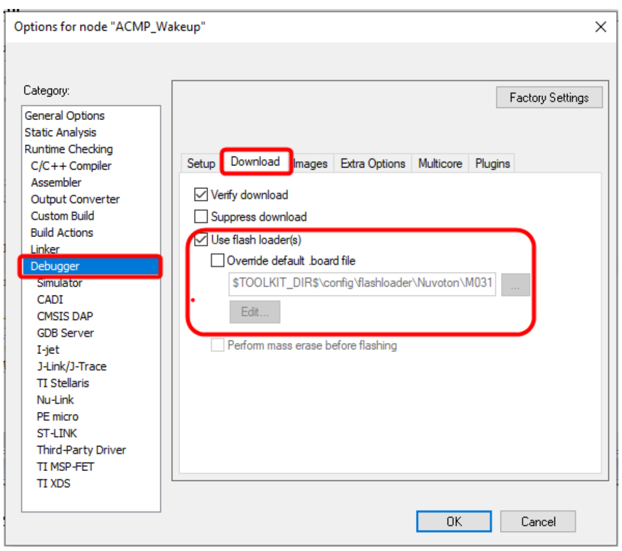


Figure 5.2-29 IAR EWARM Programming Settings

(6) In the **Download** tab of the **Debugger** page, select the **Override default .board file** option if you want the firmware to be downloaded to APROM or LDROM, and then specify the *M031_APROM.board* or *M031_LDROM.board file* (M031 series is used in this case). If no file is founded, specify the following path "*$TOOLKIT_DIR$\config\flashloader\Nuvoton*", as shown in Figure 5.2-30.

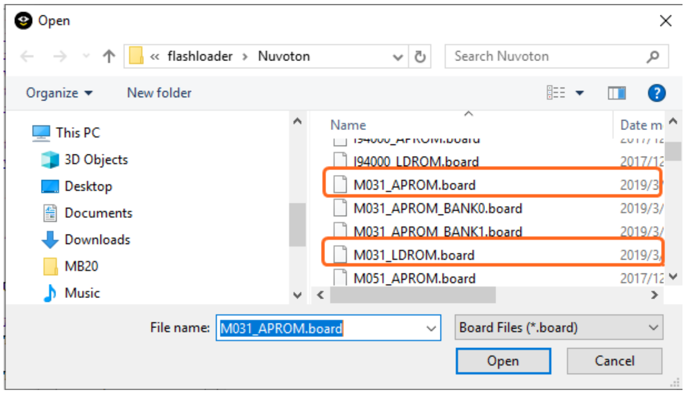


Figure 5.2-30 Select.board File for IAR EWARM

5.2.3.2 *Driver Plugin File Settings:*

(7) In the **Third-Party Driver** page, specify the path of the IAR debugger driver plugin "*C:\Program Files\Nuvoton Tools\Nu-Link_IAR\Nu-Link_IAR.dll*", as shown in Figure 5.2-31.

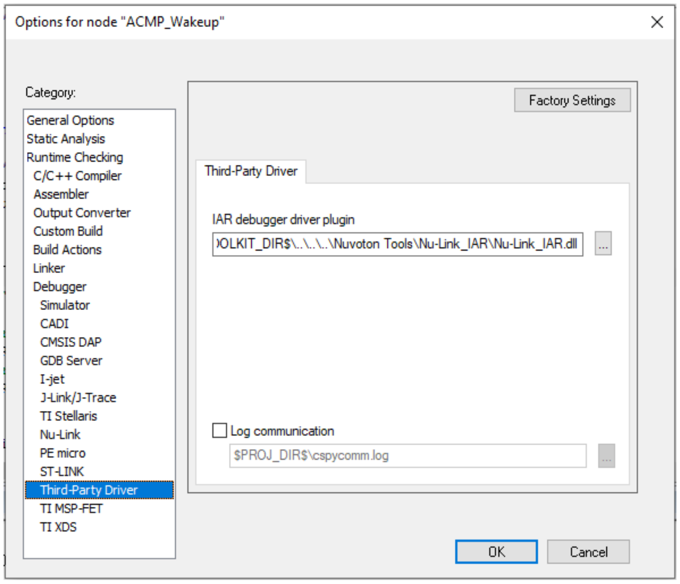


Figure 5.2-31 Set the Path of the IAR EWARM Debugger Driver Plugin

(8) Click **OK** to save the settings and return to the IAR EWARM main window.

(9) Invoke **Nu-Link2-Pro** to open the *Nu-Link* form, select **SWD** as the Port, and specify the **Nu-Link2-Pro I/O Voltage** in the **Target power control** section (3.3V in this case), as shown in Figure 5.2-32.

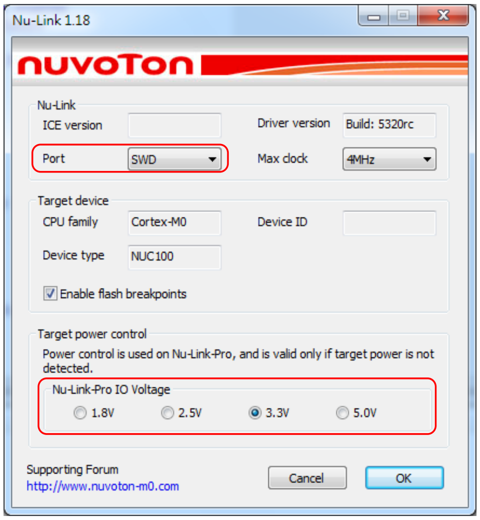


Figure 5.2-32 Specify the Port and Target I/O Voltage

5.2.3.3 *Start Programmer*

(10) Make target file as presented in Figure 5.2-33. After successfully compile the project, download code to the flash memory and enter debug mode.

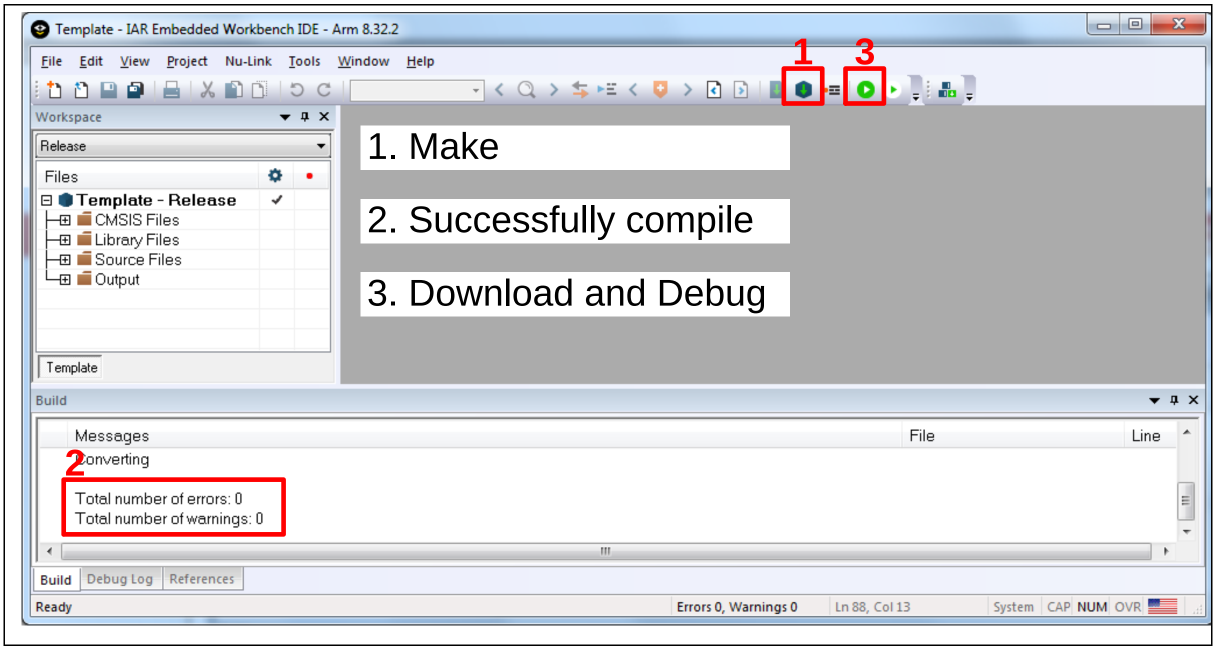


Figure 5.2-33 Compile and Download the Project

(11) Figure 5.2-34 shows the debug mode under IAR EWARN. Click “Go” and the debug message will be printed out as shown in Figure 5.2-35. User can debug the project under debug mode by checking source code, assembly language, peripherals’ registers, and setting breakpoint, step run, value monitor, etc.

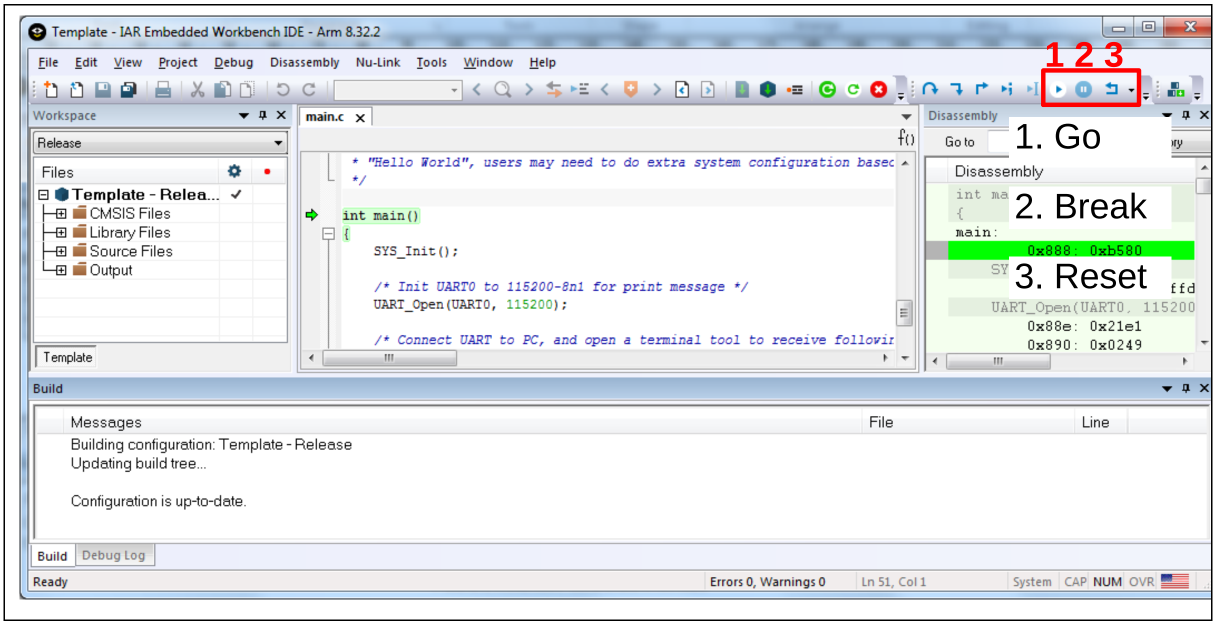


Figure 5.2-34 IAR EWARM Debug Mode

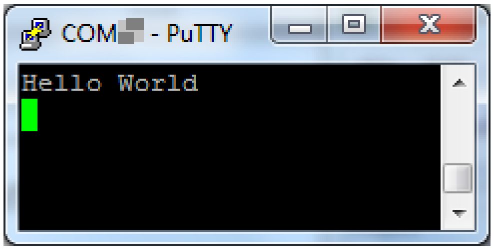


Figure 5.2-35 Debug Message on Serial Port Terminal Windows

5.2.4 NuEclipse GCC

(1) Install NuEclipse GCC, which does not require any driver installation.

(2) Double-click NuEclipse.exe to open the toolchain.

(3) Import the “Template” project by following the steps presented in Figure 5.2-36 and Figure 5.2-37.

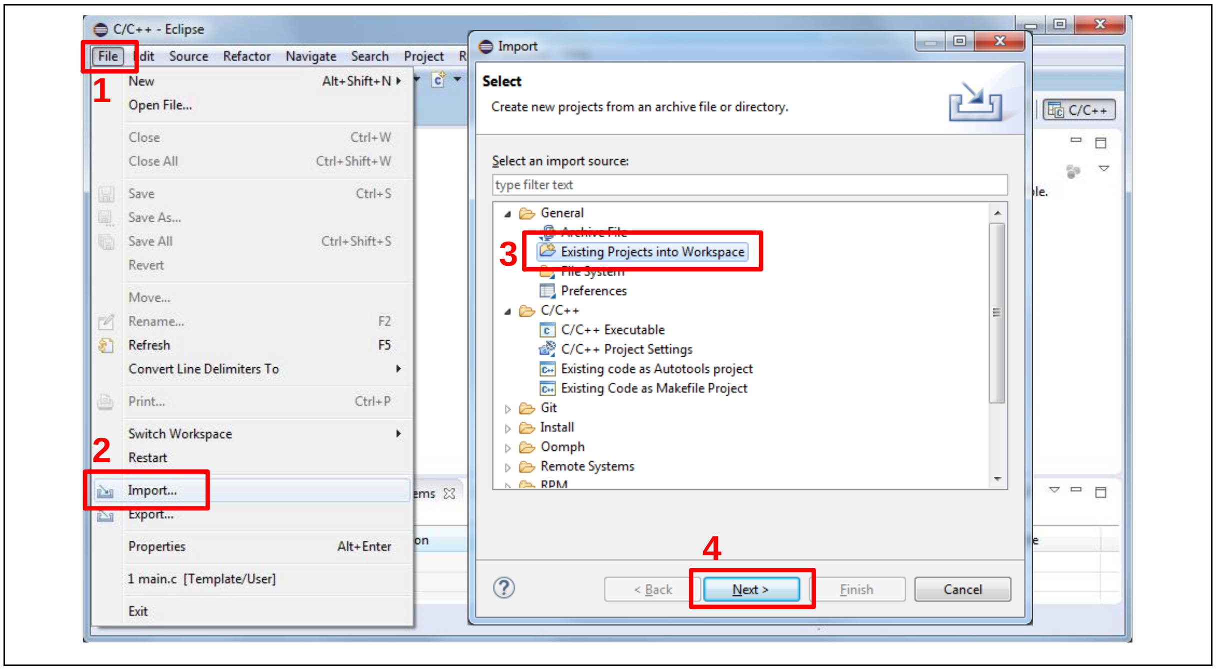


Figure 5.2-36 Import the Project in NuEclipse

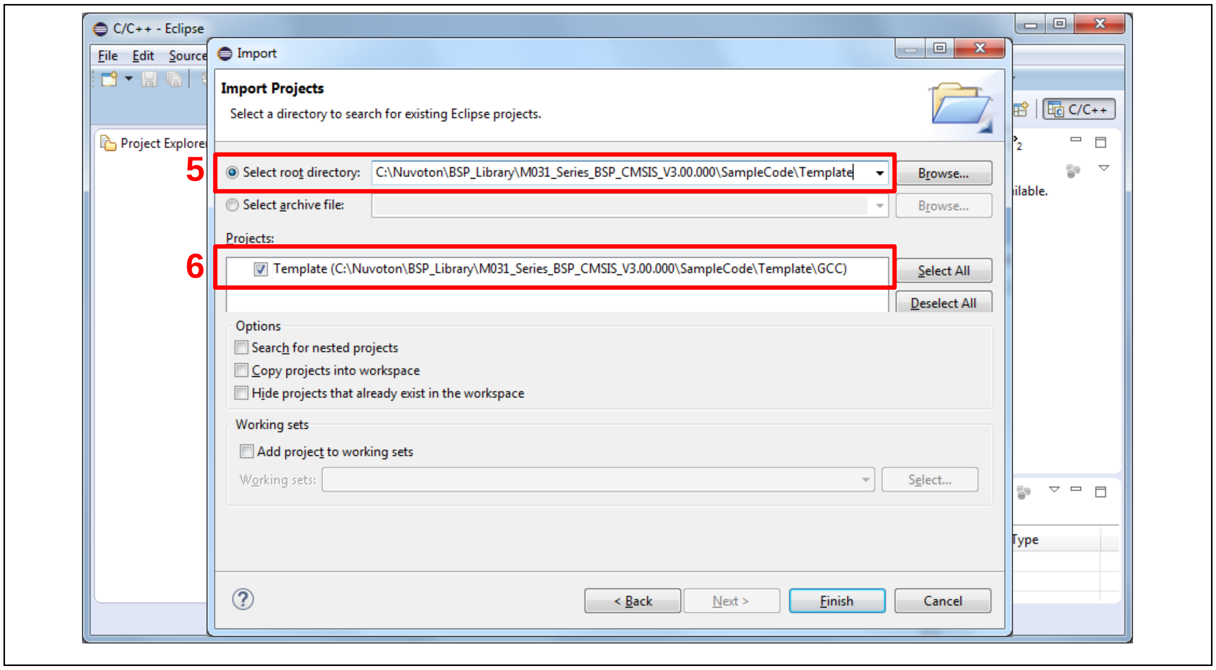


Figure 5.2-37 Import Projects Windows

(4) Click the “Template” project and find the project properties as shown in Figure 5.2-38.

Make sure the settings are the same as settings in Figure 5.2-39.

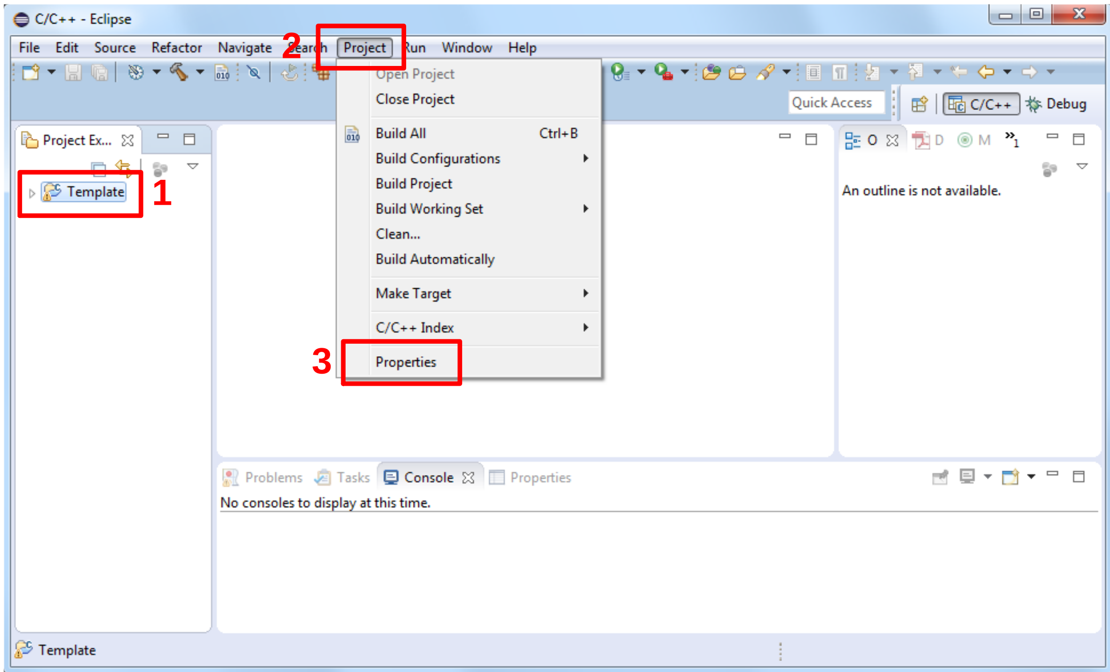


Figure 5.2-38 Open Project Properties Window

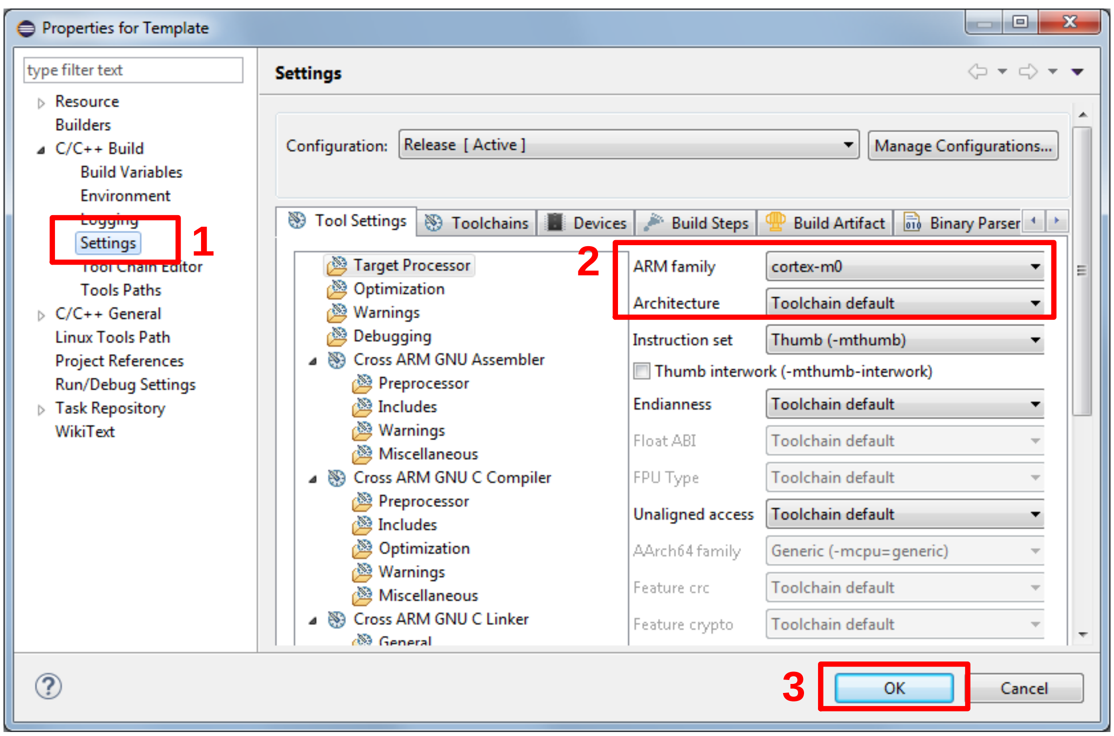


Figure 5.2-39 Project Properties Settings

(5) Click the "Template" project and build the project.

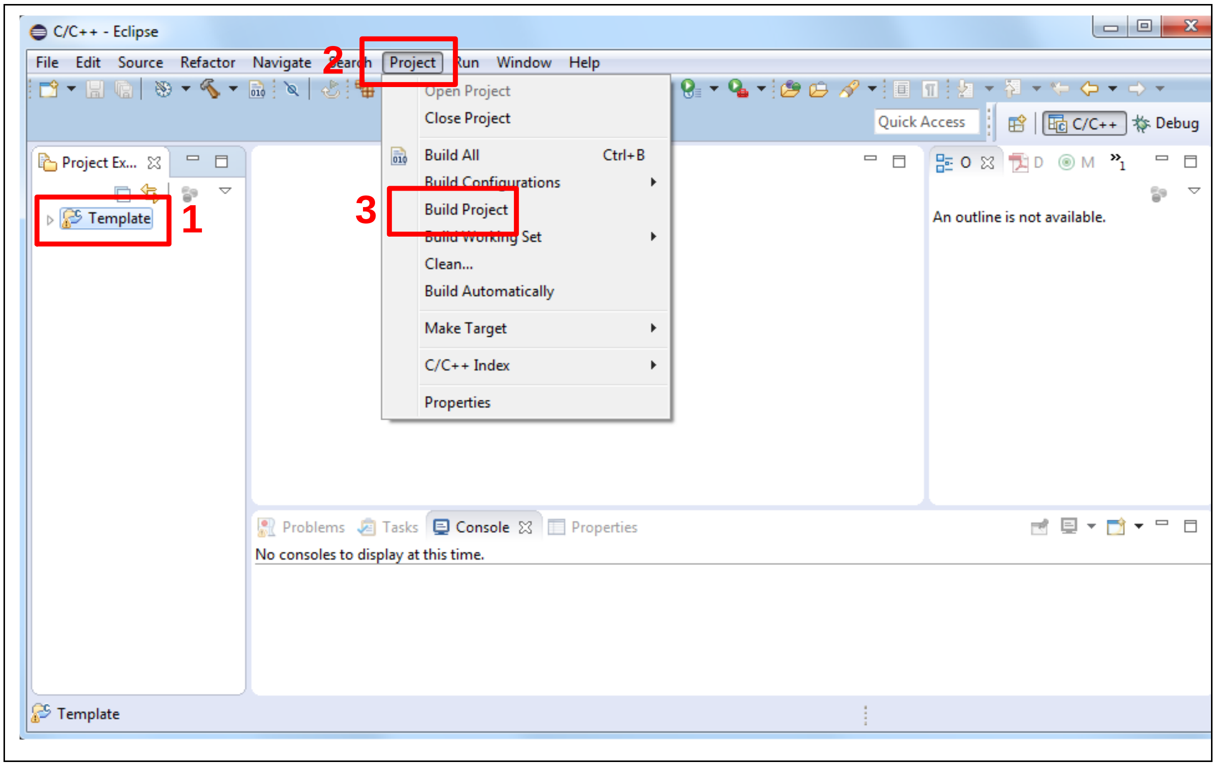


Figure 5.2-40 Build Project

5.2.4.1 *Debugger and Programming Settings:*

(6) After the project is built, click the “Template” project and set the “Debug Configuration” as shown in Figure 5.2-41. Follow the settings presented in Figure 5.2-42, Figure 5.2-43 and Figure 5.2-44 to enter debug mode.

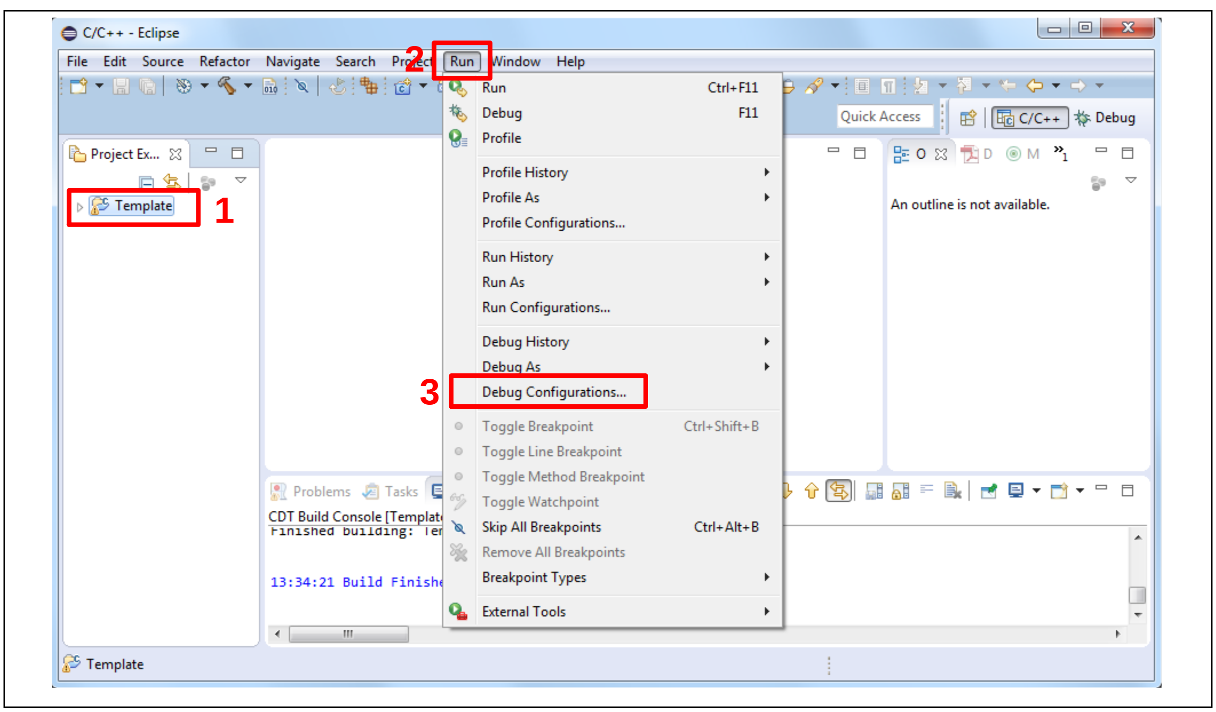
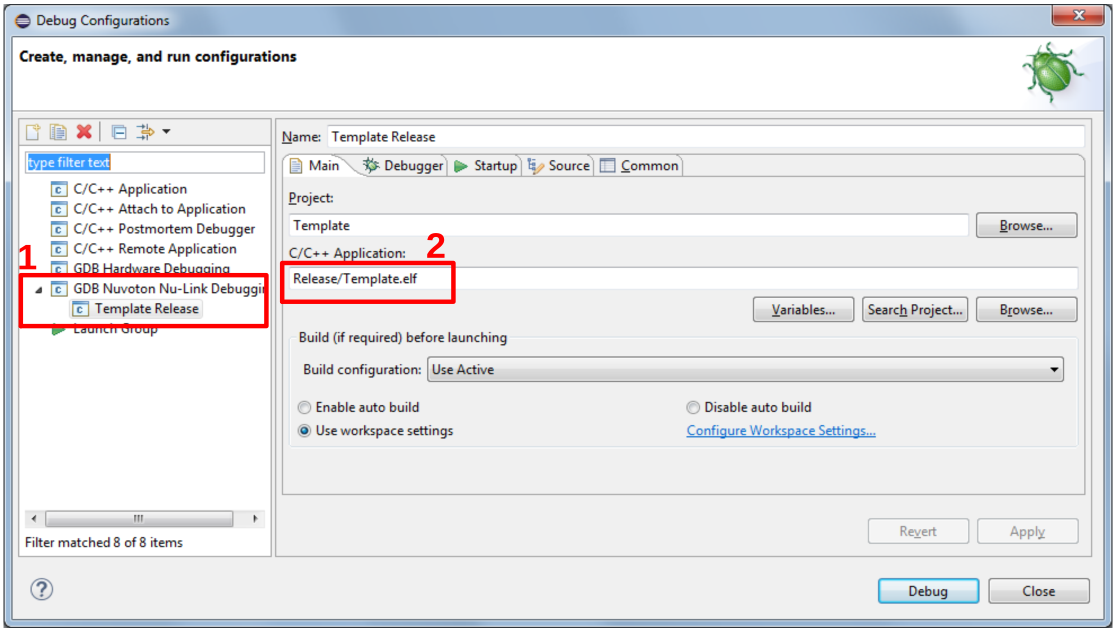


Figure 5.2-41 Open Debug Configuration



Note 1: Double click the “GDB Nuvoton Nu-Link Debugging” to create the subitem.

Note 2: After the project is built, the “*.elf” file will be shown in “C/C++ Application” frame.

Figure 5.2-42 Main Tab Configuration

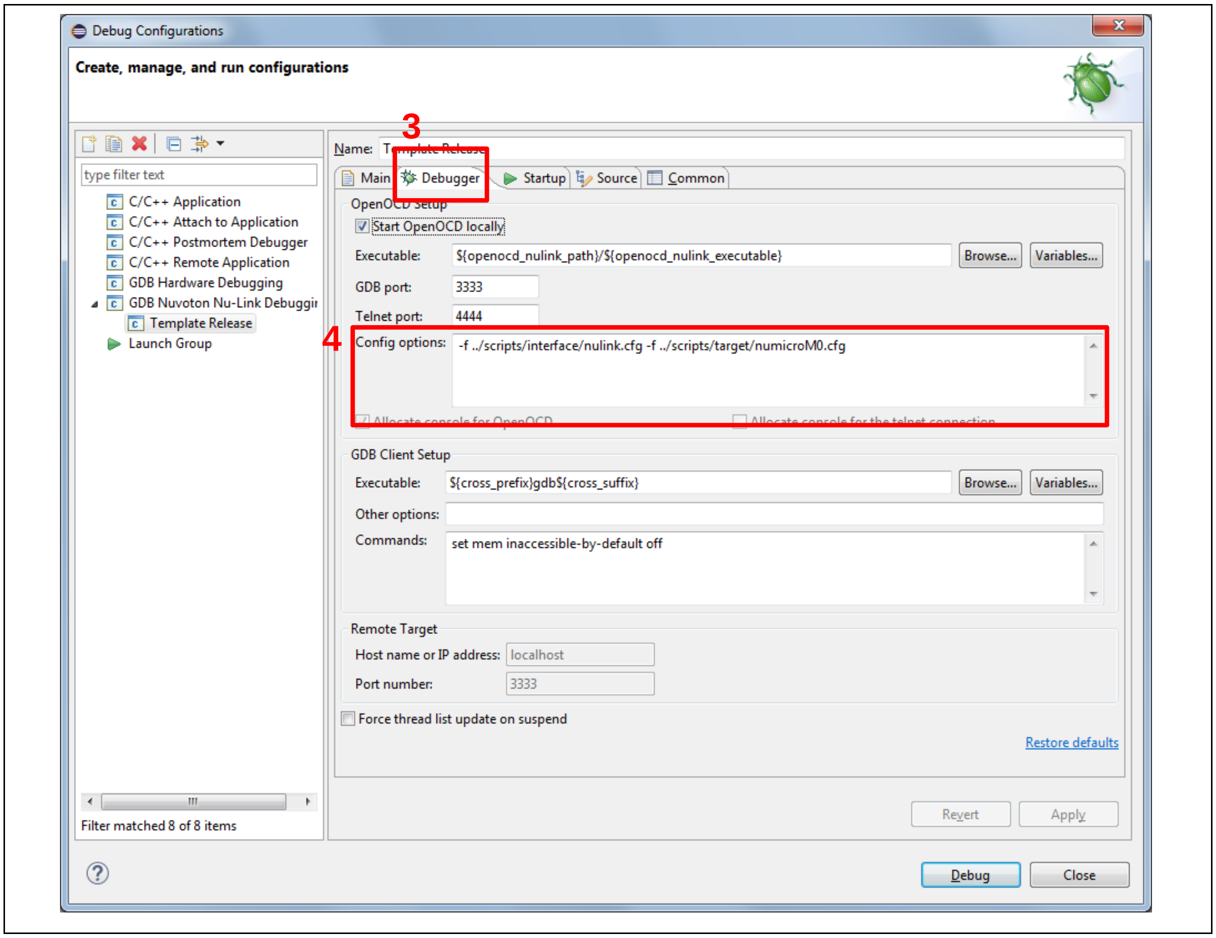
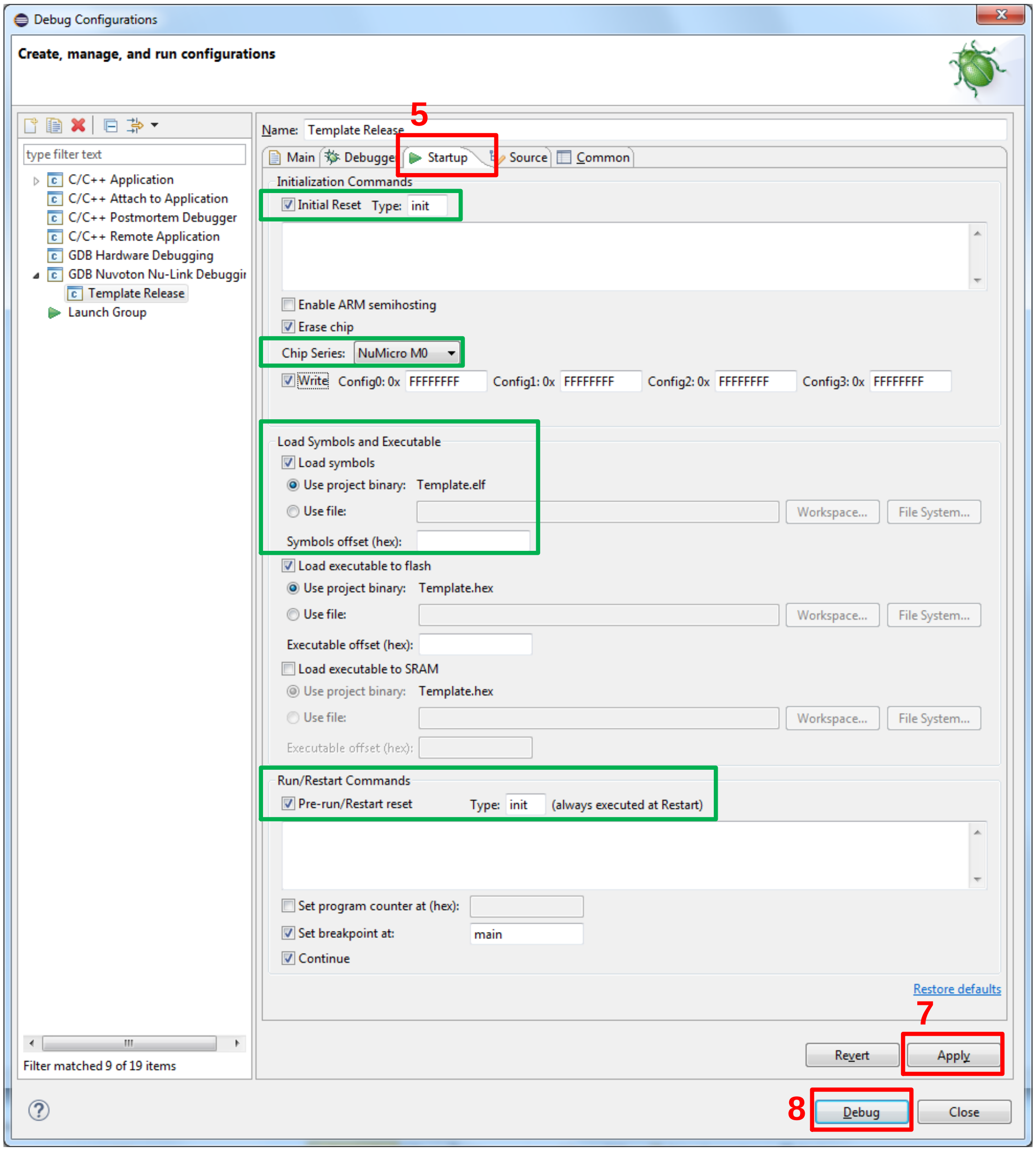


Figure 5.2-43 Debugger Tab Configuration



Note 1: Users must follow those settings highlighted in green.

Note 2: Users can configure other settings depend on the needs.

Figure 5.2-44 Debugger Tab Configuration

(7) Figure 5.2-45 shows the debug mode under NuEclipse. Click "Resume" and the debug message will be printed out as shown in Figure 5.2-46. User can debug the project under debug mode by checking source code, assembly language, peripherals' registers, and setting breakpoint, step run, value monitor, etc. For more information about how to use NuEclipse, please refer to the NuEclipse User Manual.

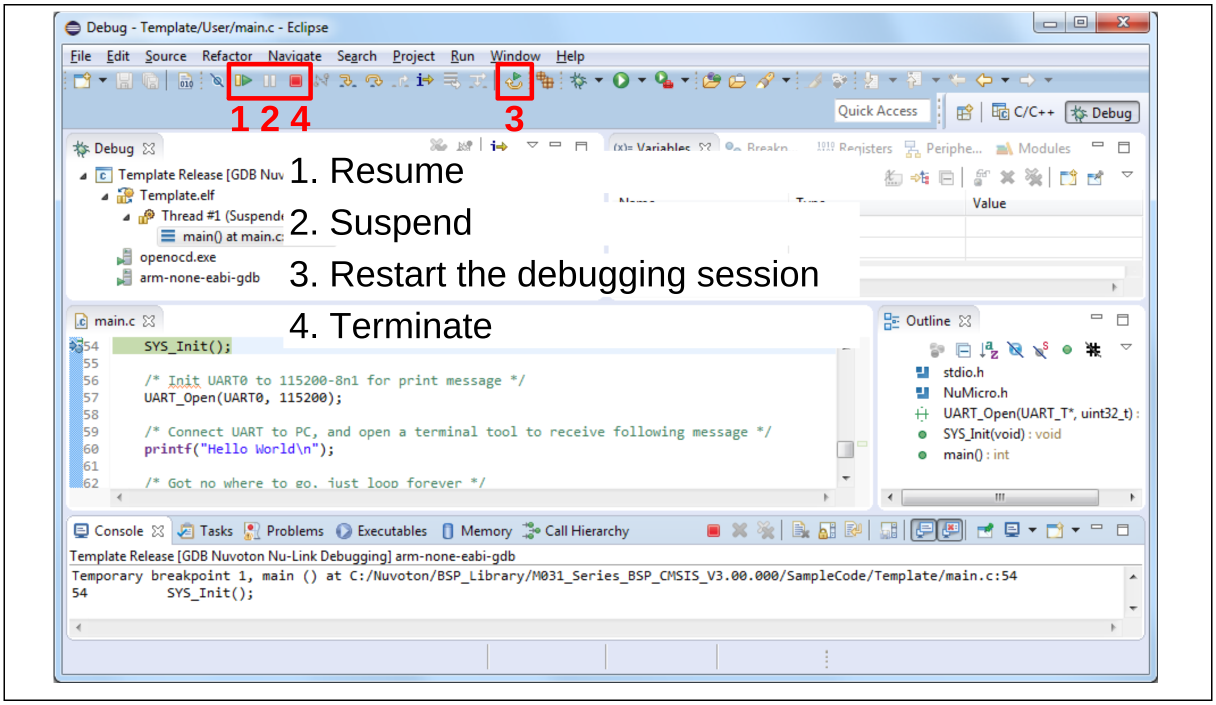


Figure 5.2-45 NuEclipse Debug Mode

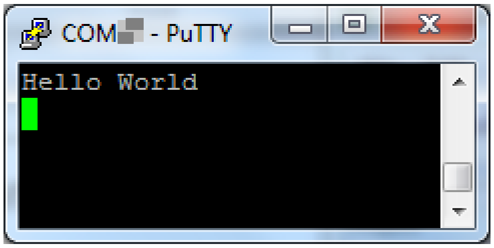


Figure 5.2-46 Debug Message on Serial Port Terminal Windows

5.2.5 ISP Tool

(1) Download and install Nuvoton NuMicro® ISP Programming Tool.

(2) Open the ISP Tool, select one connection interface to connect as shown in Figure 5.2-47

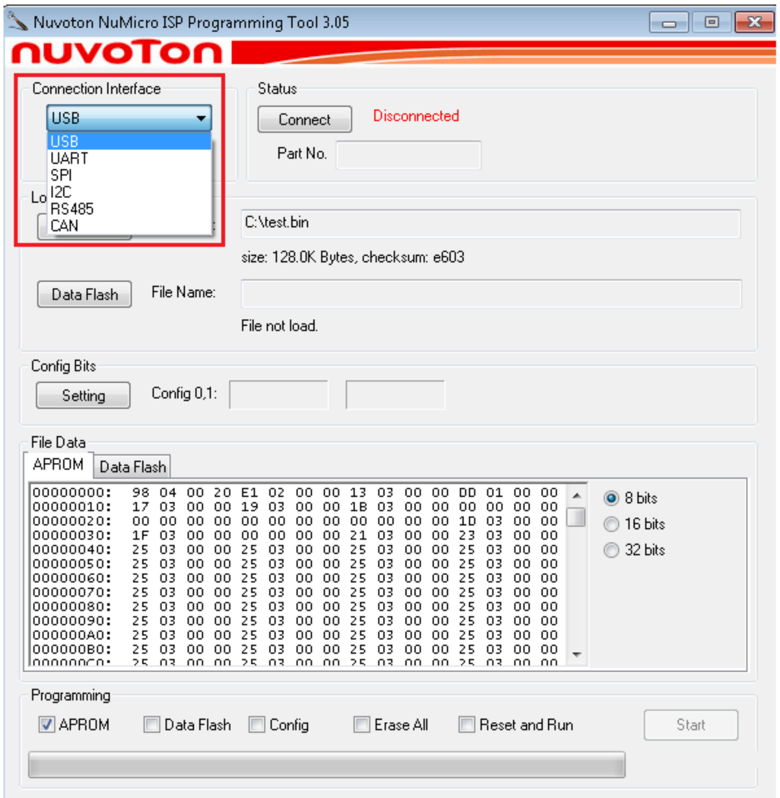


Figure 5.2-47 Startup Screen of ISP Tool

(3) In the ISP Tool window, the connection status is shown as “Disconnected” since the ISP tool has not been connected with the target chip, as shown in Figure 5.2-48.

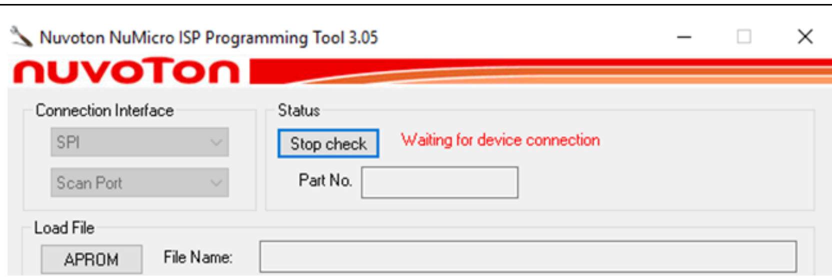


Figure 5.2-48 ISP is not connected to any device of ISP Tool

(4) Refer to section 3.3 to connect pins of Nu-Link2-Pro to target chip depend on connection interface in step 2. Connection interface option of ISP Tool mapping table as shown in Table 5.2-2.

ISP Tool Connection Interface	Pin Connection of Nu-Link2-Pro
UART[1]	Refer to section 3.2.2
I^2C	Refer to section 3.3.2
SPI	Refer to section 3.3.3
RS-485	Refer to section 3.3.4
CAN	Refer to section 3.3.5

Table 5.2-2 Connection Interface Option of ISP Tool

[1] ISP programming UART interface connection method is the same as virtual COM

(5) Download BSP sample code and open ISP firmware of Keil sample code project that it can be found in SampleCode\ISP as shown in Figure 5.2-49. The interfaces of ISP firmware sample code may be different for each NuMicro® chip series.

BSP sample code download link:

https://github.com/OpenNuvoton/

https://gitee.com/OpenNuvoton/

6 \SampleCode\ISP

ISP_CAN	Sample ISP firmware communicated with ISP tool through a CAN interface.
ISP_HID	Sample ISP firmware communicated with ISP tool through a USBD HID interface.
ISP_HID_20	Sample ISP firmware communicated with ISP tool through a HSUSBD HID interface..
ISP_I2C	Sample ISP firmware communicated with ISP tool through an I^2C interface.
ISP_RS485	Sample ISP firmware communicated with ISP tool through a RS485 interface.
ISP_UART	Sample ISP firmware communicated with ISP tool through a UART interface.

Figure 5.2-49 ISP Firmware Sample Code Project

(6) Invoke **Project** → **Options for Target** → **Utilities**, select "**Nuvoton Nu-Link Debugger**" when the **Use Target Driver for Flash Programming** option is enabled, and select the **Update Target before Debugging** option, as shown in Figure 5.2-50

- M480_Series_BSP_CMSIS_V3.XX.XXX
  - SampleCode
    - ISP
      - ISP_SPI

(7) Click the **Settings** button to open the Flash Download form, as shown in Figure 5.2-50

(8) Click the **Configure** button to open the user configuration form and set **Boot Select** option to LDROM and click **OK** button as shown in Figure 5.2-50

(9) Download code to LDROM of target chip.

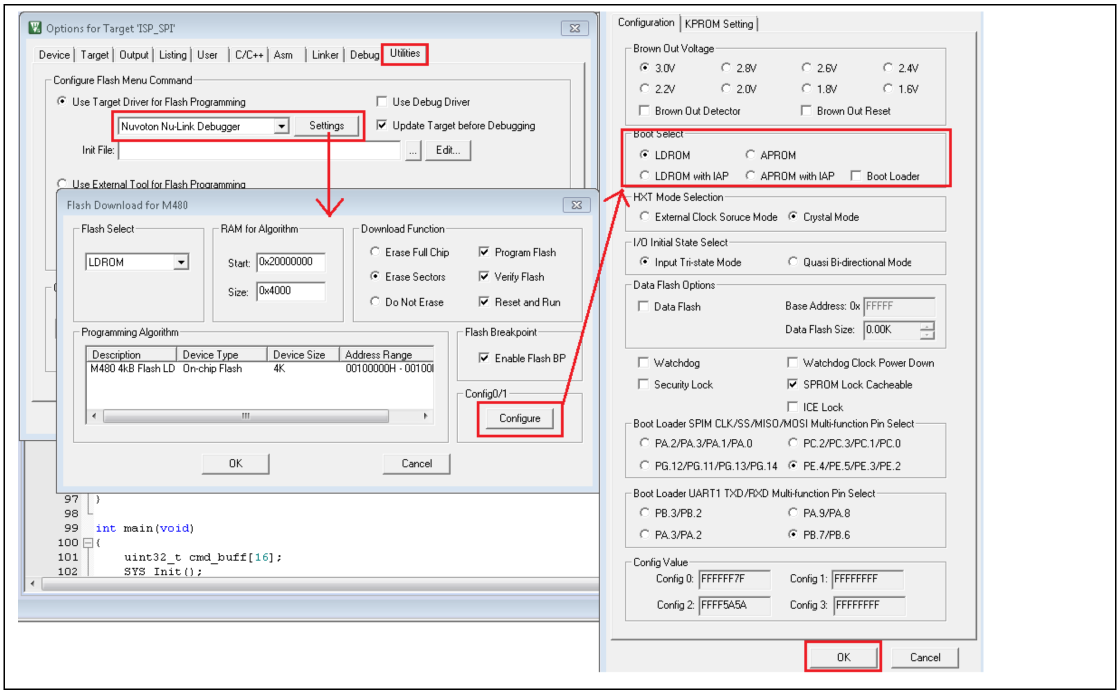


Figure 5.2-50 Boot from LDROM Setting in Keil ISP Firmware Project

(10) Open ISP programming tool, click **Connect** button, and reset the target chip to run ISP code. ISP programming tool will connect to target chip. For example, SPI connection interfaces as shown in Figure 5.2-51. For UART connection interface, user needs to select VCOM port number as shown in Figure 5.2-52. All of connection interfaces that provided by ISP tool can connect to target chip by using above steps except USB interface.

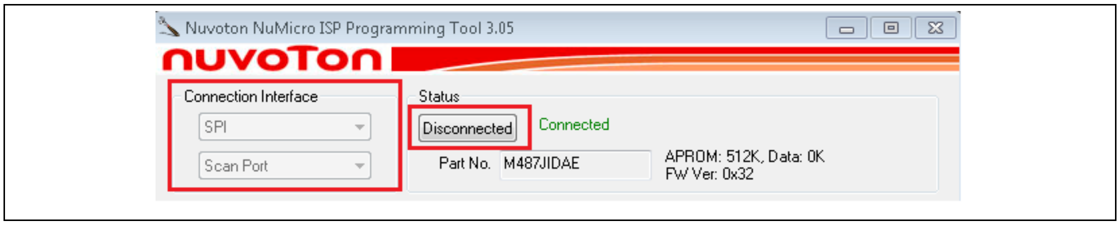


Figure 5.2-51 Connect to Target Chip with SPI Interface

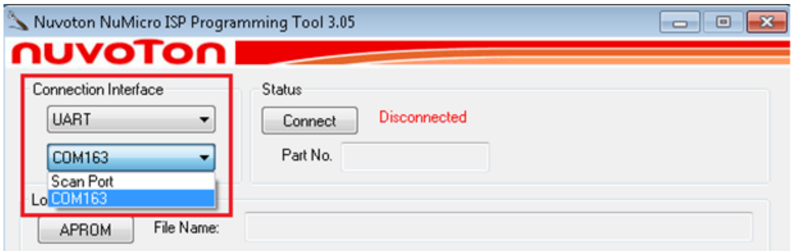


Figure 5.2-52 Select VCOM Port Number with UART Interface

In ISP_HID firmware sample code (USB interface), there is a control pin to control target chip to run APROM or LDROM code. The control pin may be different for each NuMicro® chip series and please refer to each BSP sample code. Target chip will run APROM code when the control pin is in high and target chip will run LDROM code when the control pin is low. User must keep the control pin in low and click **Connect** button of ISP programming tool. ISP programming tool will connect to target chip with USB connection interface.

ISP Interface	Reset and Control Pin
SPI, UART, I^2C, RS-485, CAN	Chip reset to reboot in LDROM, ISP FW to connect ISP tool
USB	Chip reset to reboot in LDROM, ISP FW to check control pin in low state for ISP process

Table 5.2-3 Entering the ISP Conditions for ISP Tool Connection

(11) Load programming file and check the programming option and click **Start** button. ISP Programming Tool start to program file to target chip by user selected connection interface as shown in Figure 5.2-53.

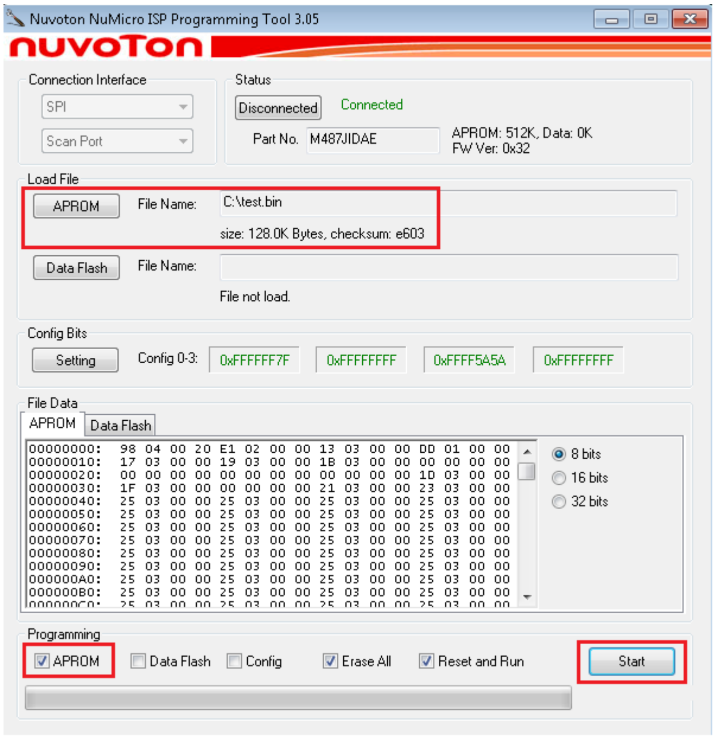


Figure 5.2-53 Program Data to Target Chip with SPI Interface

5.2.6 More Features of Nu-Link2-Pro Firmware

Nu-Link2-Pro also provides different interface to program user code to target chip. These various feature and programming interface correspond to different firmware bin file. User can switch the roles that Nu-Link2-Pro play by re-programming Nu-Link2-Pro to another .bin file.

The brief description of Nu-Link2-Pro firmware as follows:

1. **NuLink2_DAPLink.bin**

   NuLink2_DAPLink.bin is the firmware supported ARM Mbed DAPLink and PyOCD for programming and debugging.

2. **NuLink2_ISPLink2.bin**

   NuLink2_ISPLink2.bin is the Nu-Link2-Pro firmware that can use offline mode to program code with bridge interfaces. User puts the programming data to Nu-Link2-Pro first and then programs data to target chip with bridge interface.

3. **NuLink2_Bus_Monitor.bin**

   NuLink2_Bus_Monitor.bin is the Nu-Link2-Pro firmware can monitor the bus data of NuLink2_ISP_Bridge between two connected devices.

The Nu-Link2-Pro also provides a method to update firmware by USB mass storage. Please follow the steps below:

(1) Hold offline button of Nu-Link2-Pro shown in in Figure 2.3-1, plug in USB cable and release the button.

(2) A "Nu-Link2" disk will show as Figure 5.2-54. (If you see disk name is "NuMicro MCU", it will upgrade DUT firmware instead of Nu-Link2-Pro itself)

(3) Drag and drop Nu-Link2 image .bin into the disk.
(4) Re-plug the USB cable to complete update firmware.

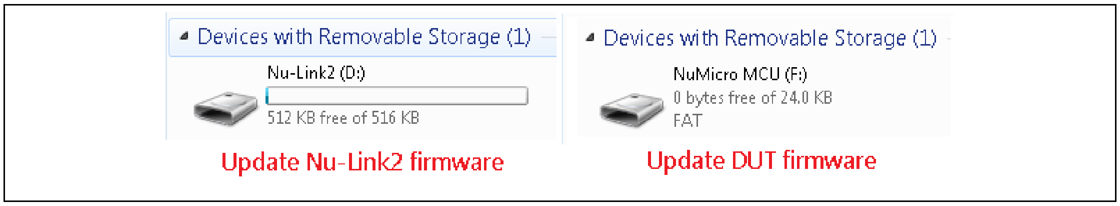


Figure 5.2-54 Update Nu-Link2 Firmware or DUT Firmware

Please click the link below for further information and resource:

https://github.com/OpenNuvoton/Nuvoton_Tools

https://github.com/OpenNuvoton/Nuvoton_Tools/tree/master/Latest_NuLink_Firmware

https://gitee.com/OpenNuvoton/Nuvoton_Tools

https://gitee.com/OpenNuvoton/Nuvoton_Tools/tree/master/Latest_NuLink_Firmware

6 APPENDIX

6.1 Nu-Link2-Pro Operating Current of ICP

When power is supplied via an USB during ICP online programming, the operating current of Nu-Link2-Pro is shown in the Table 6.1-1 below.

SWD I/O Mode Settings	5.0 V	3.3 V	2.5 V	1.8 V
USB Input Voltage (V)	5.0	5.0	5.0	5.0
SWD I/O Voltage (V)	4.66	3.33	2.52	1.82
USB Input Current (mA)	128	117	115	113

Table 6.1-1 Nu-Link2-Pro Operating Current (Online Programming)

When power is supplied from a target board (SWD VCC pin) during offline programming and offline file on SPI flash, the operating current of Nu-Link2-Pro is shown in the Table 6.1-2 below.

Power Supplied from a Target Board	5.0 V	3.3 V	2.5 V	1.8 V
Power Supplied via an USB	Off	Off	Off	Off
SWD VCC Input Voltage (V)	5.01	3.33	2.51	1.82
SWD VCC Input Current (mA)	77.5	127.5	155.4	167.5

Table 6.1-2 Nu-Link2-Pro Operating Current (Offline Programming) of SPI Flash

When power is supplied from a target board (SWD VCC pin) during offline programming and offline file on USB flash drive, the operating current of Nu-Link2-Pro is shown in the Table 6.1-3 below.

Power Supplied from a Target Board	5.0 V	3.3 V	2.5 V	1.8 V
Power Supplied via an USB	Off	Off	Off	Off
SWD VCC Input Voltage (V)	5.00	3.22	2.52	1.82
SWD VCC Input Current (mA)	77.6	123.3	152.6	161.7

Table 6.1-3 Nu-Link2-Pro Operating Current (Offline Programming) of USB Flash

When power is supplied from a target board (SWD VCC pin) during offline programming and offline file on Micro SD card , the operating current of Nu-Link2-Pro is shown in the Table 6.1-4 below.

Power Supplied from a Target Board	5.0 V	3.3 V	2.5 V	1.8 V
Power Supplied via an USB	Off	Off	Off	Off
SWD VCC Input Voltage (V)	5.01	3.28	2.53	1.81
SWD VCC Input Current (mA)	77.3	125.5	154.6	165.2

Table 6.1-4 Nu-Link2-Pro Operating Current (Offline Programming) of Micro SD Card

6.2 Nu-Link2-Pro Operating Current of ISP

The operating current of Nu-Link2-Pro during ISP online programming with power supply via USB is shown in the Table 6.2-1 below.

ISP programming Interface	I^2C	SPI	RS-485	CAN	UART
USB VCC Input Current (mA)	117.1	114.3	151	191	114.2
Target board Input Current (mA)	11.9	15.1	47.1	90.1	15

Table 6.2-1 Operating Current of ISP Online Programming

6.3 Automatic IC Programming System

The automatic IC programming system through individual slot and the Control Bus as Figure 6.3-1.

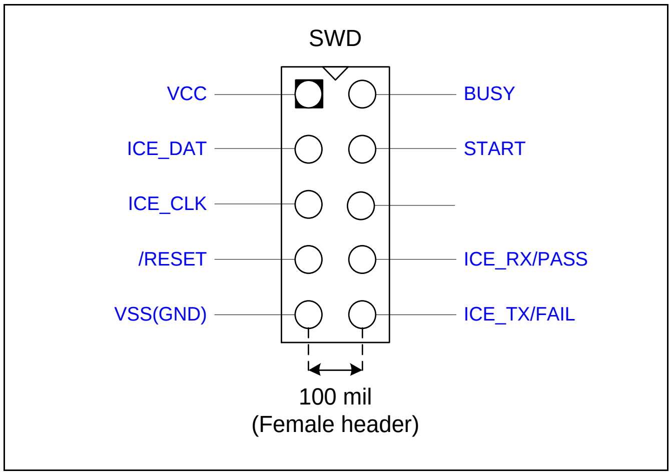


Figure 6.3-1 SWD Connector Pin Diagrams

6.3.1 Operation Sequence and Waveform

1. The Nu-Link2-Pro power on. START, BUSY, PASS, and FAIL are set to logic.
2. To start programming, START needs to be set to logic 0 for TSTART, $50\text{ms} \leq T_{START} \leq 80\text{ms}$
3. Programming start-up. BUSY is set to logic 0, and might toggle during programming.
4. When finish programming, BUSY is set to logic 1, and PASS or FAIL is set to logic 0.
   - When BUSY is set to logic 1, and PASS is set to logic 0, means “PASS”.
   - When BUSY is set to logic 1, and FAIL is set to logic 0, means “FAIL”.

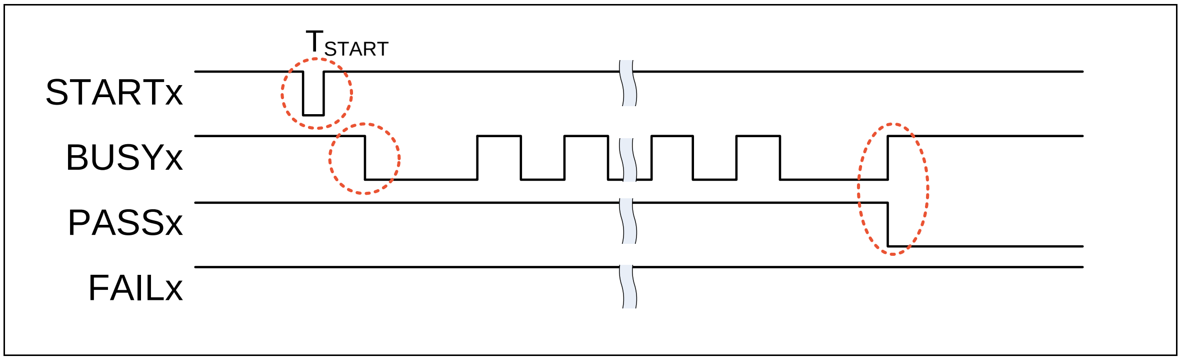


Figure 6.3-2 PASS Waveform

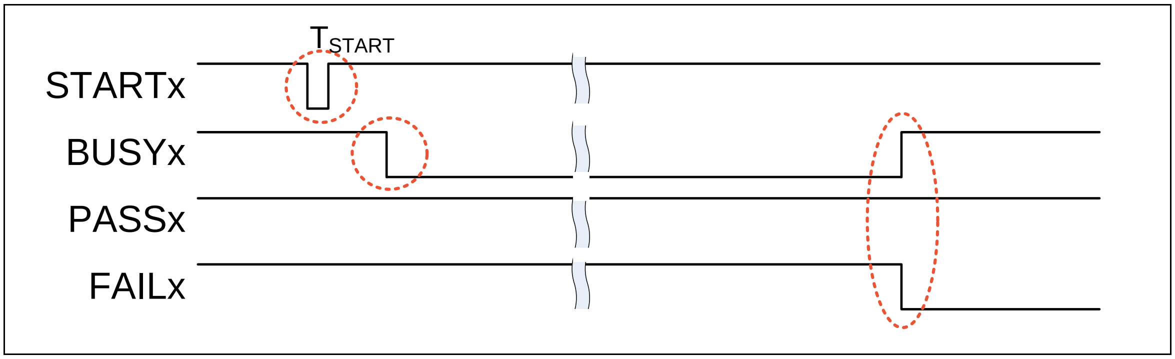


Figure 6.3-3 FAIL Waveform

6.1 Nu-Link Debugger and Programmer Comparison

The Nu-Link Debugger and Programmer series provides an USB connector and a SWD signal interface for connecting to the target chip. The user can connect the Nu-Link Debugger and Programmer to an USB port of a PC to debug and program target chips through the development software tools. As shown in Table 6.1-1, there are three specifications for the Nu-Link Debugger and Programmer and Table 6.1-2 two specifications for the Nu-Link Debugger and Programmer with development board, in which debugging, Online/Offline Programming, and SWD I/O voltage settings may be supported depending on the specifications (refer to the relevant section for details).

Function	Type	Nu-Link2-Pro 	Nu-Link-Pro 	Nu-Link 
Debug	Debug via SWD	✓	✓	✓
	ETM	✓	-	-
	DAPLink/pyOCD	✓	-	-
Program	Online ICP Programming	✓	✓	✓
	Offline ICP-Button	✓	✓	✓
	Offline ICP-Control Bus[1]	✓	-	-
	Drag & drop Flash programming	✓	-	-
	SWD I/O Voltage Support	1.8V, 2.5V, 3.3V, 5.0V	1.8V, 2.5V, 3.3V, 5.0V	5.0V
Upgrade	Online ISP	✓	-	-
Storage	SPI Flash	✓	✓	✓
	SD Card	✓	-	-
	USB Flash Drive	✓	-	-
Bridge	Virtual COM	✓	-	-
Bus Monitor	I^2C, SPI, CAN, RS-485	✓		

Table 6.1-1 Comparison of All Nu-Link Debugger and Programmer

Note:

1. The Nu-Link2-Pro and Nu-Link2-Me can be connected to an automatic IC programming system through the Control Bus.

Function \ Type		Nu-Link2-Me[2] [3] 	Nu-Link-Me 
Debug	Debug via SWD	✓	✓
	ETM	-	-
	DAPLink/pyOCD	-	-
Program	Online ICP Programming	✓	✓
	Offline ICP-Button	✓	-
	Offline ICP-Control Bus[1]	✓	-
	Drag & drop Flash programming	✓	-
	SWD I/O Voltage Support	1.8 V, 3.3 V, 5.0 V[3]	3.3 V, 5.0 V[2]
Upgrade	Online ISP	✓[4]	-
Storage	SPI Flash	✓	-
	SD Card	-	-
	USB Flash Drive	-	-
Bridge	Virtual COM	✓	✓

Table 6.1-2 Comparison of Integrated Nu-Link Debugger and Programmer on Development Board

Note:

1. The Nu-Link2-Pro and Nu-Link2-Me can be connected to an automatic IC programming system through the Control Bus.
2. Adjusted by resistor JPR1.
3. Adjusted by resistor ICEJPR1.
4. Nu-Link2-Me only supports UART interface for ISP update.

7 REVISION HISTORY

Date	Revision	Description
2020.03.13	1.00	1. Initially issued.
2020.03.24	1.01	1. Modify some related descriptions in the introduction section