

User's guidance

N32H473CEU7_STB Development Board Hardware Usage Guide

Introduction

The purpose of this document is to allow users to quickly familiarize themselves with the N32H473CEU7_STB development board, understand the functions, instructions and precautions of the development board, so as to conduct MCU debugging and development based on the development board.

CONTENS

1	HARDWARE DEVELOPMENT INSTRUCTIONS.....	1
1.1	<i>Briefly</i>	<i>1</i>
1.2	<i>Development board function.....</i>	<i>1</i>
1.3	<i>Development board layout.....</i>	<i>2</i>
1.4	<i>Development Board Jumper Instructions</i>	<i>4</i>
1.5	<i>Development board schematic</i>	<i>6</i>
2	VERSION HISTORY	11
3	NOTICE.....	12

1 Hardware Development Instructions

1.1 Briefly

The N32H473CEU7_STB development board is used for sample development of the 32-bit N32H473CEU7 chip of National Technology Co., Ltd. This document describes in detail the functions, usage instructions and precautions of the N32H473CEU7_STB development board.

1.2 Development board function

The main MCU chip model of the development board is N32H473CEU7, with UQFN48 pin package. The development board connects all functional interfaces to facilitate customer development.

1.3 Development board layout

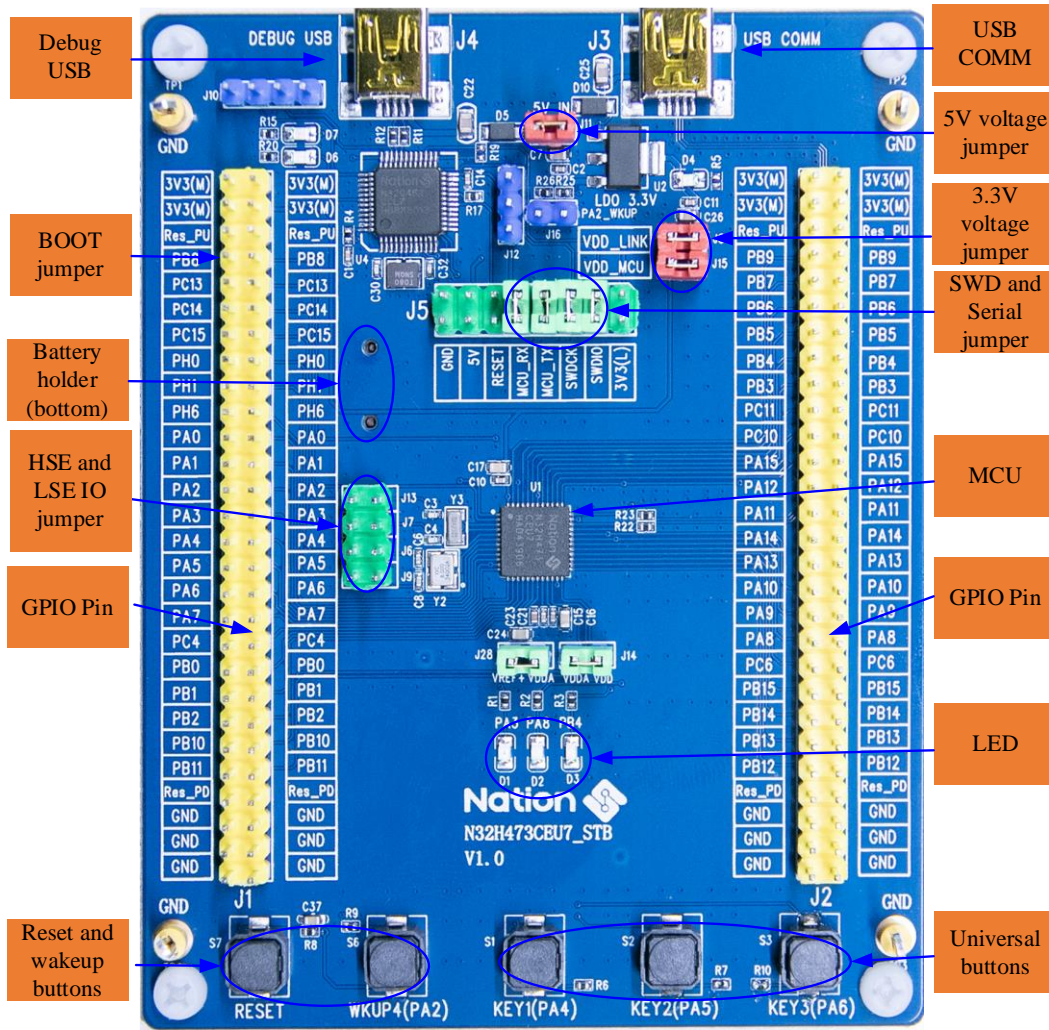


Figure 1-1 Development board layout

1) Power supply for the development board

The development board can be powered by USB COMM interface (J3) or DEBUG USB (J4), and is connected to the 3.3V LDO input port through the J11 jumper.

2) USB COMM interface (J3)

Through Mini USB interface (J3) is used to connect the DP and DM signals of the main MCU (U1) for USB interface communication of the main MCU.

3) Debug USB (J4)

Through the DEBUG USB interface of the NS-LINK chip (U4), the main MCU program download and debugging function can be provided, and the serial port of the MCU can also be connected to provide the USB to serial port function.

4) SWD interface and Serial port (J5)

SWD interface: SWDIO and SWDCK, used for downloading and debugging the main MCU program. ULINK2 or JLINK can be used to download and debug the MCU. You can also use a jumper cap to short-circuit the SWDIO signal pin and SWDCK signal pin to download the MCU through DEBUG USB. debug .

Serial port: MCU_TX and MCU_RX are used as external serial port signals. PA9 (TX) and PA10 (RX) of MCU are used as serial ports. They can be connected to external serial port devices separately, or the jumper cap can be shorted to the MCU_TX signal pin and MCU_RX signal pin. Through NS-LINK on the development board, the USB port is converted into a serial port for the convenience of customers;

5) Reset and Wake Buttons (S7, S6)

S7 and S6 are the reset button and wake-up button respectively, which are connected to the NRST pin and PA2 pin of the chip respectively for chip reset and wake-up functions.

6) Universal keys (S1, S2, S3)

S1, S2, and S3 are general buttons, which are connected to the chip PA4, PA5, and PA6 pins respectively.

7) LED lights

D1, D2, and D3 are LED lights, which are connected to the chip PA3, PA8, and PB4 pins respectively.

8) BOOT (J1 PIN7/8)

The J1 PIN7/8 PB8 pin is the BOOT0 pin, which can be shorted to power and ground through a jumper cap as needed.

9) GPIO□ (J1, J2)

All chip GPIO interfaces are lead out, and 3.3V voltage and GND pins are also reserved on the pins to facilitate testing. For the specific definition of the interface, please refer to "UM_N32H47X Series User Manual V1.0".

1.4 Development Board Jumper Instructions

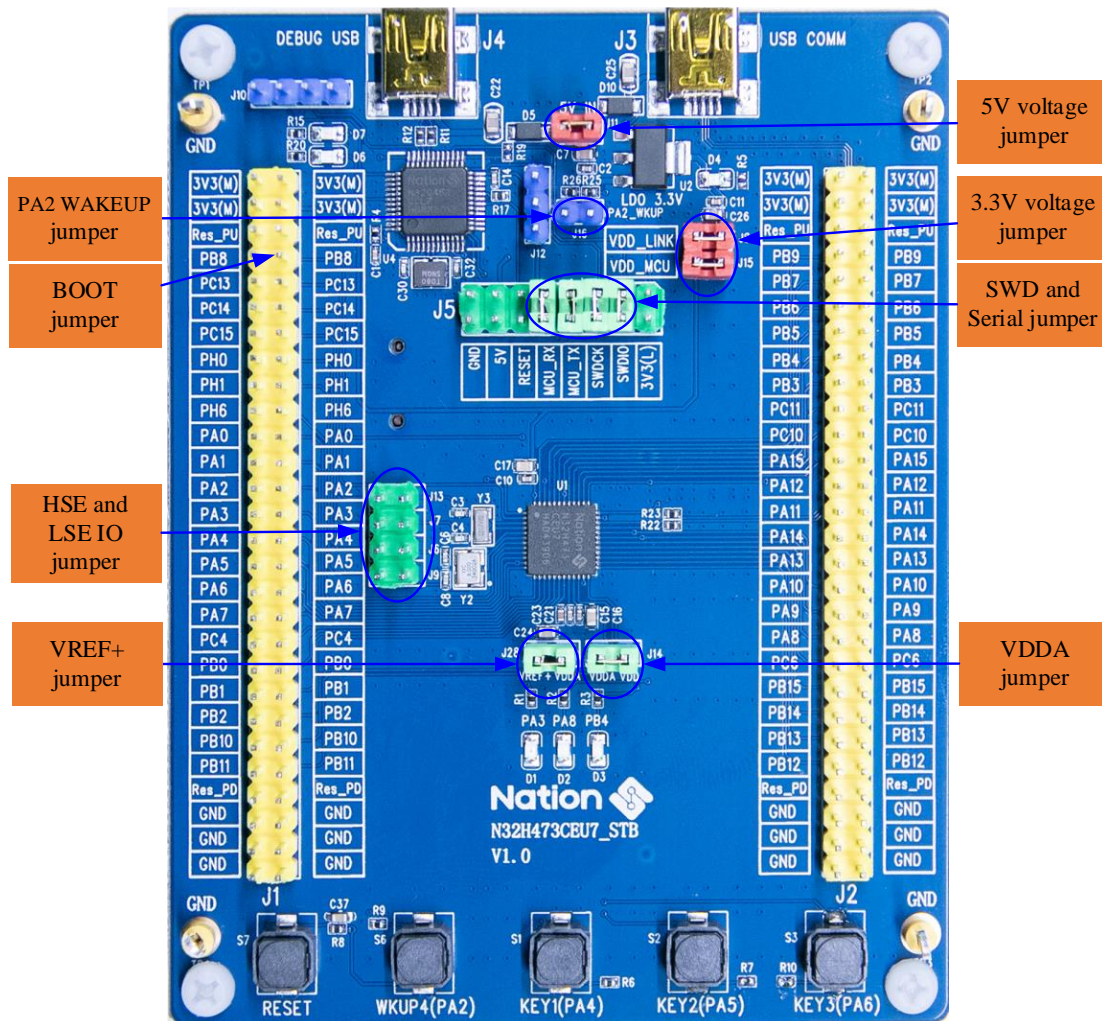


Figure 1-2 Development Board Jumper Description

Table 1-1 Development Board Jumper Description List

No.	Jumper bit number	Jumper function	Instructions for use
1	J11	5V input voltage jumper	The J11 jumper is used to connect the two USB interfaces J3 and J4 to supply power to the LDO3.3V input port.
2	J8、J15	3.3V Power supply jumper	J8: Power Supply 3.3V to NS-LINK MCU chip. J15: Power supply 3.3V to the main MCU chip.
3	J5	SWD jumper	Use NS-LINK to download the program to the MCU through the USB DEBUG port. You need to short-circuit the SWDIO signal pin and the SWDCK signal pin..
	J5	Serial jumper	When using NS-LINK as a serial port through the USB DEBUG port, you need to short-circuit the MCU_TX signal pin and the MCU_RX signal pin..
4	J1 PIN7/8	BOOT jumper	J1 PIN7/8: BOOT0.
5	J16	PA2 WAKEUP jumper	J16: Short this jumper and when the USB interface is inserted, wake up the MCU through PA2 (set the PA2 bit to the WKUP signal).
6	J14	VDDA jumper	J99: Short this jumper, VDDA is directly connected to VDD for power supply.
7	J28	VREF+ jumper	J103: Short this jumper and VREF uses external VDD as the reference source.
8	J7、J13	LSE pin connected to J1 IO jumper	J13, J7: Short this jumper and connect the PC14 and PC15 pins of LSE to the PIN11/12 and PIN13/14 pins of J1 for external debugging.
9	J6、J9	HSE pin connected to J1 IO jumper	J6, J9: Short this jumper and connect the PH0 and PH1 pins of HSE to the PIN15/16 and PIN17/18 pins of J1 for external debugging.

The schematic diagram of the N32H473CEU7_STB development board is described as follows (see "N32H473CEU7_STB_V1.0" for details):

Refer to Figure 1-3 for the MCU connection schematic diagram. Each VDD pin of the MCU is connected to a capacitor, and all GPIOs are connected to the J1, J2 and J14 pins for easy debugging.



Refer to Figure 1-4 for the power design schematic. The PCB supplies 5V through USB (J3/J4), and then outputs 3.3V voltage through the LDO to power the entire PCB board.

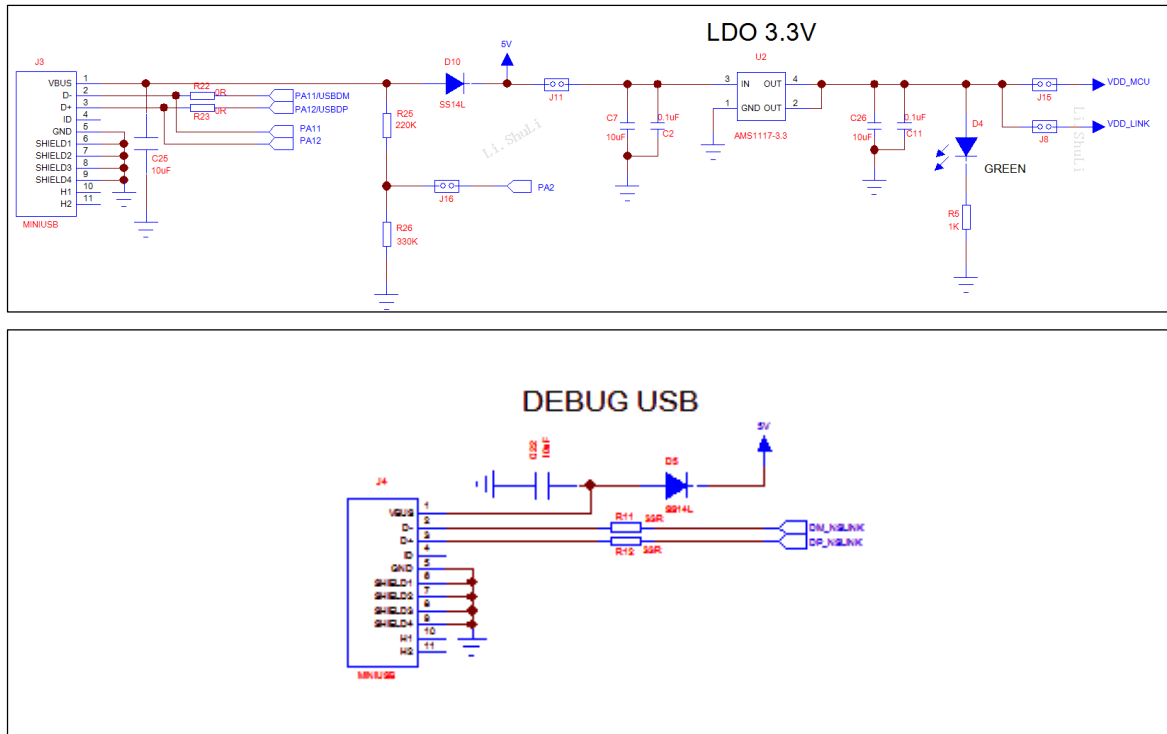


Figure 1-4 Power Design

3) Button design

Refer to Figure 1-5 for the schematic diagram of button design. There are 5 buttons in total, including 3 general buttons, MCU wake-up button and reset button.

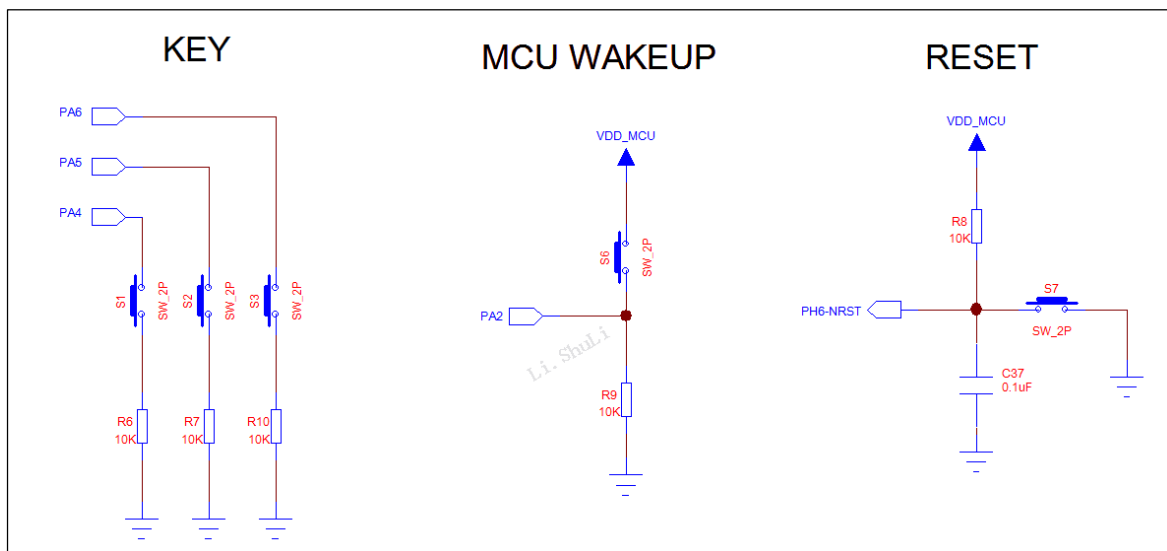


Figure 1-5 Button Design

4) LED light design

Refer to Figure 1-6 for the LED light design schematic. There are 5 LED lights in total. D1, D2, and D3 are connected to PA3, PA8, and PB4 of the main MCU respectively, and can be used for debugging. D6 and D7 are used for NS-LINK MCU control and monitoring of NS-LINK operating status.

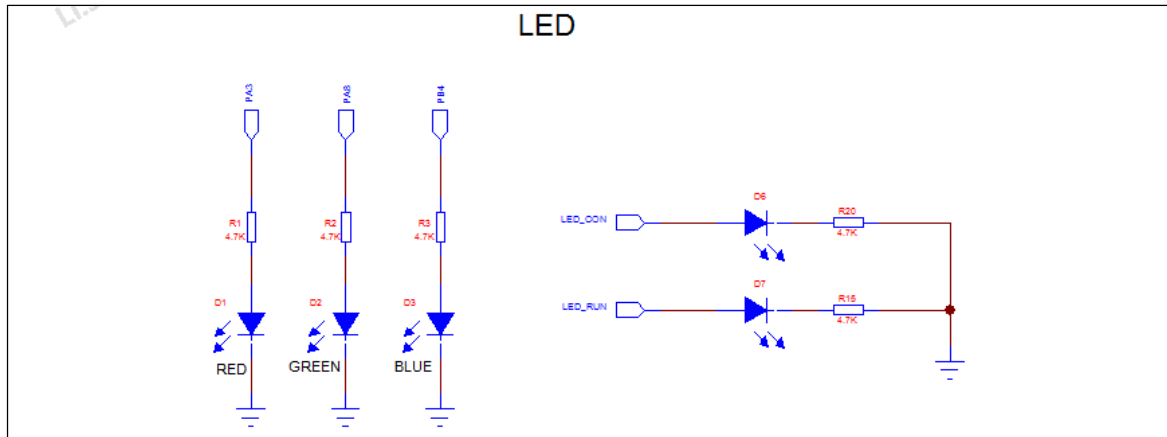


Figure 1-6 LED Light Design

5) Crystal

Refer to Figure 1-7 for the crystal connection diagram. The chip has two external crystals, 32.768KHz and 8MHz respectively.

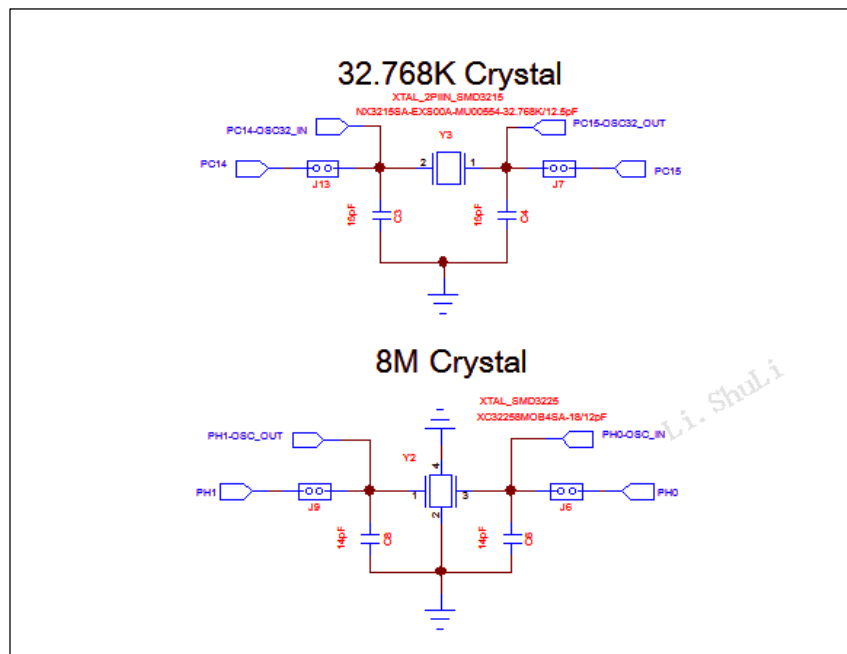


Figure 1-7 crystal design

6) Backup battery BAT

Figure 1-8 shows the schematic diagram of the BAT battery external connection. The VBAT pin can be powered by an external CR1220 battery through the external battery holder on the PCB board.

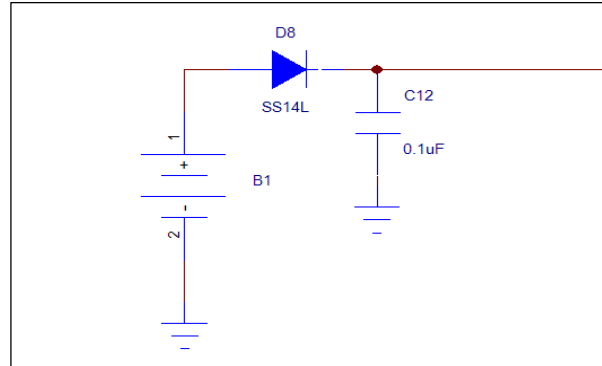


Figure 1-8 crystal design

7) NS-LINK

Refer to Figure 1-9 for the NS-LINK schematic diagram. Users can directly connect the USB cable through the DEBUG USB port to download the program, eliminating the need for a ULINK or JLINK programmer. It can also be debugged through the DEBUG USB analog serial port.

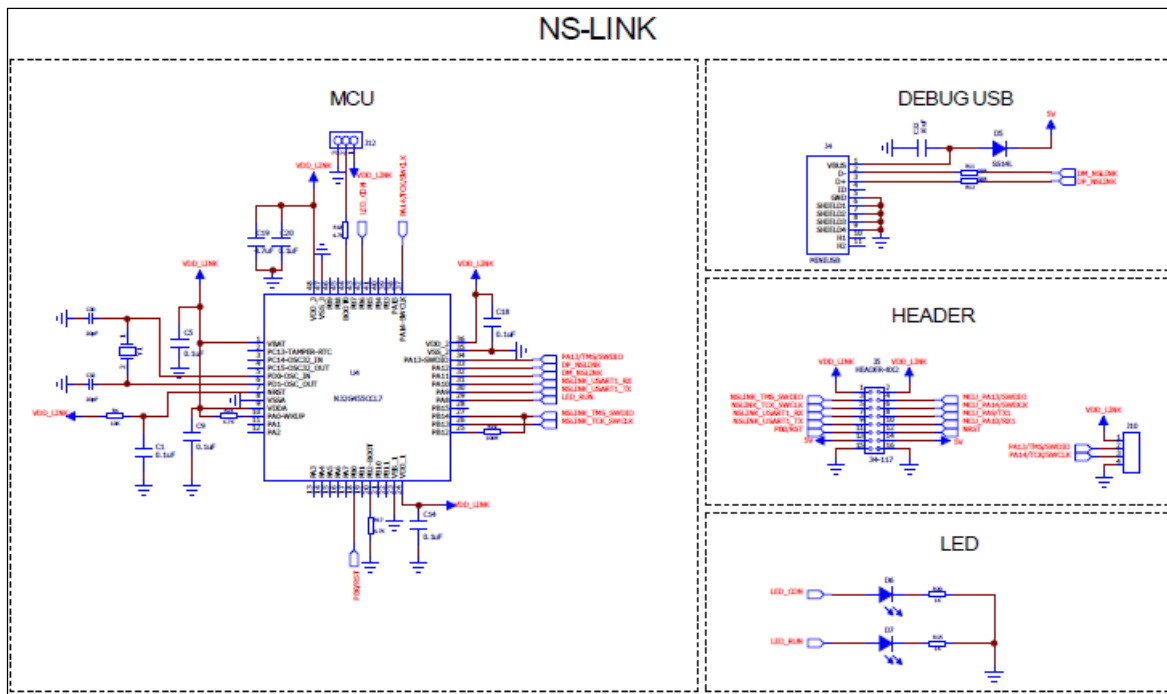


Figure 1-9 NS-LINK

● **MCU peripheral device description:**

- 1) When designing the PCB LAYOUT, place two capacitors nearby VDD (PIN144), 10uF and 0.1uF respectively, and place 0.1uF capacitors nearby the other VDD pins.
- 2) It is recommended to place a 0.1uF and a 10uF capacitor on the VDDA input pin.
- 3) When VERF+ uses the built-in reference source VREFBUF, it is recommended to place a 0.1uF and a 1uF capacitor nearby the VREF+ pin. When VERF+ is powered externally, it is recommended to place a 0.1uF and a 10uF capacitor nearby the VREF+ pin.
- 4) PC14-OSC32_IN, PC15-OSC32_OUT: When there is a need for an external high-precision RTC clock, an external 32.768KHz crystal needs to be connected close to the pin. If there is no need, it can be omitted.

2 Version history

Version	Date	Modify
V1.0	2024-4-2	Initial version

3 Notice

This document is the exclusive property of Nations Technologies Inc. (Hereinafter referred to as NATIONS). This document, and the product of NATIONS described herein (Hereinafter referred to as the Product) are owned by NATIONS under the laws and treaties of the People's Republic of China and other applicable jurisdictions worldwide.

NATIONS does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. Names and brands of third party may be mentioned or referred thereto (if any) for identification purposes only.

NATIONS reserves the right to make changes, corrections, enhancements, modifications, and improvements to this document at any time without notice. Please contact NATIONS and obtain the latest version of this document before placing orders.

Although NATIONS has attempted to provide accurate and reliable information, NATIONS assumes no responsibility for the accuracy and reliability of this document.

It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. In no event shall NATIONS be liable for any direct, indirect, incidental, special, exemplary, or consequential damages arising in any way out of the use of this document or the Product.

NATIONS 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, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at user's risk. User shall indemnify NATIONS and hold NATIONS harmless from and against all claims, costs, damages, and other liabilities, arising from or related to any customer's Insecure Usage.

Any express or implied warranty with regard to this document or the Product, including, but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement are disclaimed to the fullest extent permitted by law.

Unless otherwise explicitly permitted by NATIONS, anyone may not use, duplicate, modify, transcribe or otherwise distribute this document for any purposes, in whole or in part