NDA102 Controls AS7221 by the DALI Command

Example Code Introduction for 32-bit NuMicro® Family

Information

Application	This example code is to add UART interface to the original DALI application project for communication with AS7221, so that DALI Command can control the AS7221 module.
BSP Version	102_207_209_NDA102_200707_Lib
Hardware	DALI_SLAVE_NDA102EC1 V1.0

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

www.nuvoton.com

1 Overview

This example code can receive the instructions sent by the DALI (Digital Addressable Lighting Interface) host, and then send commands to AS7221 through the UART1 interface to control the lamp behavior in the module, such as adjusting brightness, color temperature, gradient time and so on.

1.1 Principle

AS7221 is an intelligent lighting management chip, which adopts embedded three-color CIE XYZ color sensor technology. It can realize the precise induction of illumination and color temperature, and then control its own PWM output to achieve the precise control of brightness and color temperature. By combining with the original DALI application solution, accurate control of lamps can be realized with DALI instructions.

The DALI host sends dimming and color instruction through the bus. NDA102 receives and parses the instruction as the slave, and then sends the corresponding control command to the AS7221 through UART1, so as to realize the precise regulation of lamps by DALI instruction.
The format of the AS7221 command is:

Set command: command string with parameters, such as "ATcmd= xxx". This will return the string "OK" on success and " ERROR " on failure.

Query command: command string, such as: "ATcmd", the query success returns the string "value OK", failure returns the string "ERROR".

Note: " ATcmd " is the command defined by AS7221, "xxx" is the value to be set, and "value" is the value to be found.

Please refer to "AS7221 Smart Lighting Command Set 3V0V1" for detailed command introduction of AS7221.

1.2 Operation Process

The AS7221 is powered on first, then the host is powered on, and then the slave device DALI_SLAVE_NDA102EC1 V1.0 is powered on. After one second, press the key SW1 on the slave to switch the mode of AS7221. At this time, dimming and color can be controlled by the host and the lamps on AS7221 can be seen to change accordingly.

The host used in this example is Tridonic DALI USB. It is necessary to install a PC host software, which can be found in the start menu after installation, as shown in Figure 1-1 Start
The download path of software is:
https://www.tridonic.com/com/en/software-masterconfigurator.asp

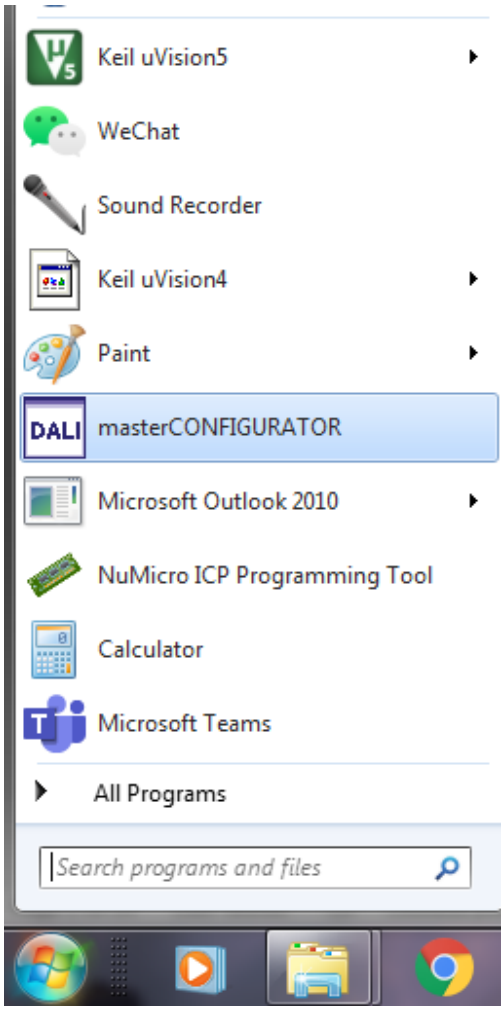

Figure 1-1 Start Menu

Open the PC software of the host and click the Addressing button to search the slave device connected to the bus, as shown in Figure 1-2 PC Software Interface of the .

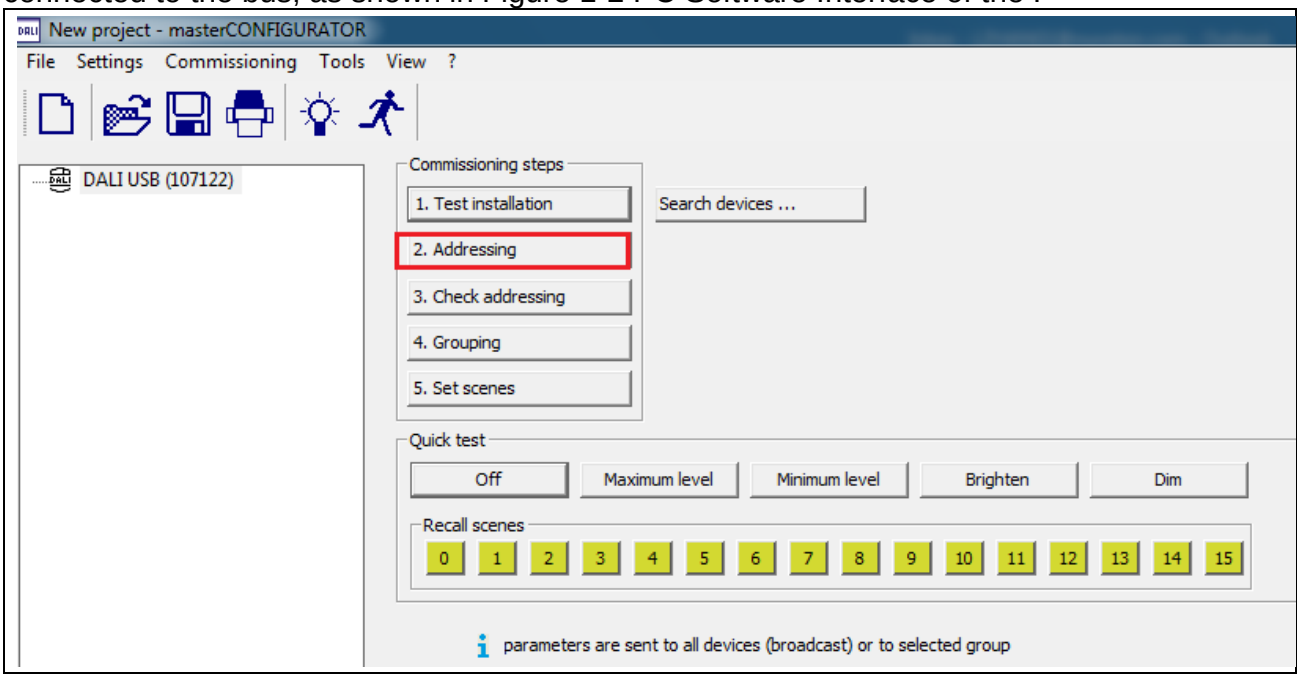

Figure 1-2 PC Software Interface of the Host

Then choose to search the slave device again, and click the Next button, as shown in Figure 1-3 Search Slave .

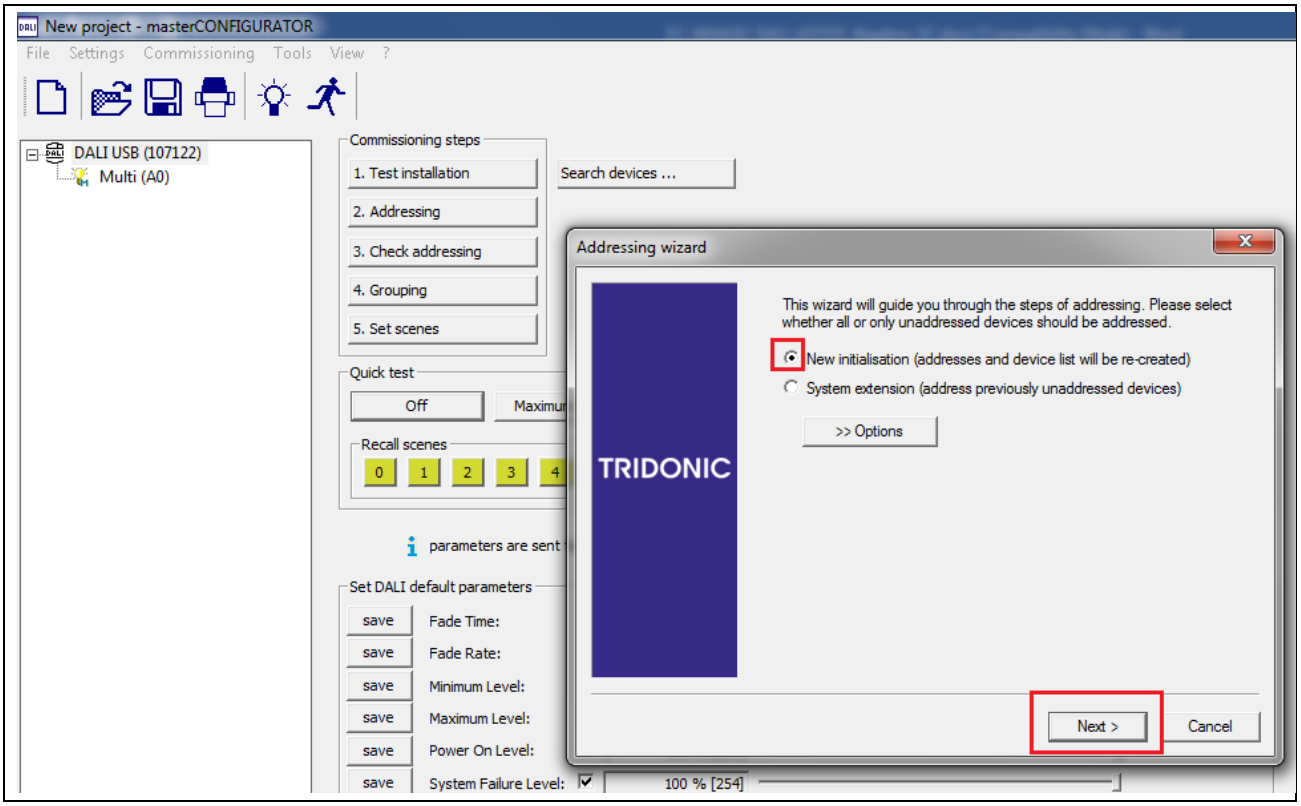

Figure 1-3 Search Slave Device

When a slave device is searched, the slave device will be displayed on the PC interface. Then click the displayed slave device (Multi (A0)) to obtain the information of the slave device and display it, as shown in Figure 1-4 Slave Device

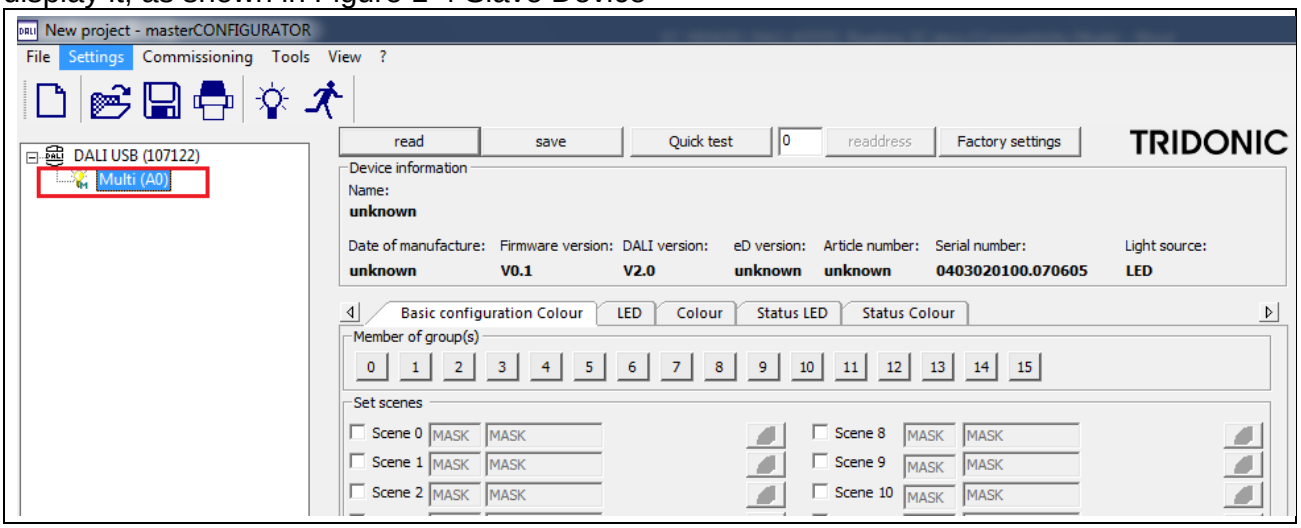

Figure 1-4 Slave Device Information

Then click the “Quick Test” button, and a quick debugging interface will be displayed. Click the “Color Temperature” button on the interface, so that the brightness and color temperature can be adjusted conveniently. The slider above adjusts the brightness, and the slider below adjusts the color temperature. Click the corresponding “Send” button to send the selected brightness percentage and color temperature, as shown in Figure 1-5 Quick Test

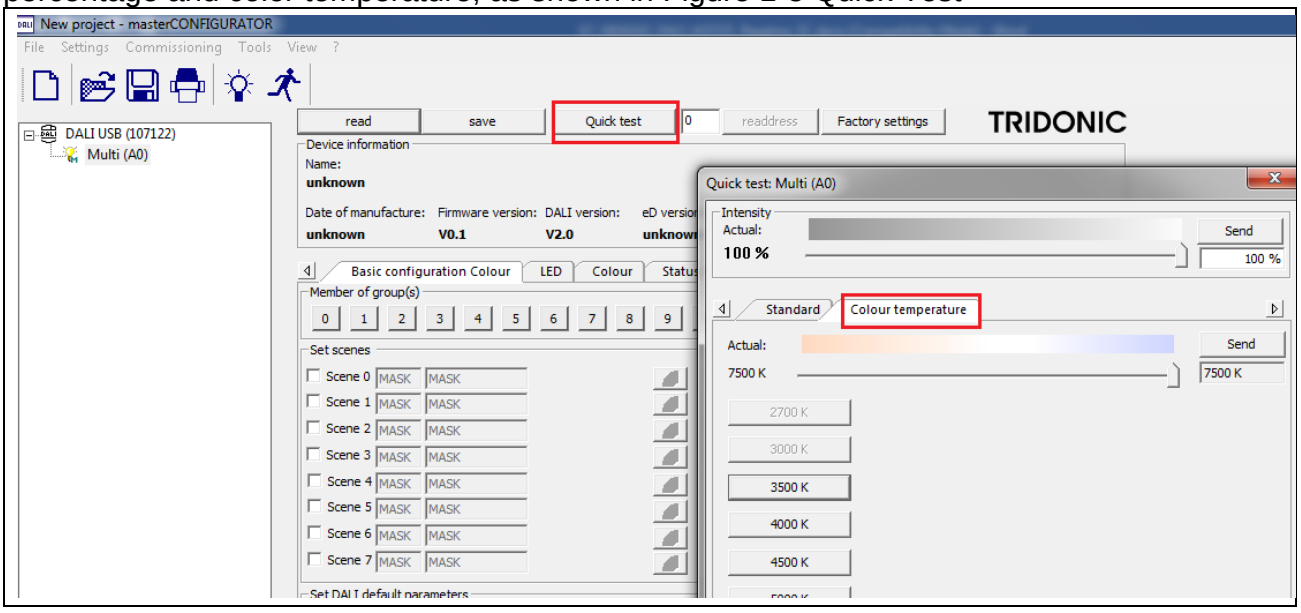

Figure 1-5 Quick Test Interface

The slave device used in this example is DALI_SLAVE_NDA102EC1 V1.0. Please use Nu-Link-Me to supply power to the board from the burning port, and also the burning firmware. It should be noted that UART1 is used to send serial instructions to AS7221. So the TX and RX

on the burning interface should be connected to the serial port of AS7221. In addition, DALI - and DALI + on the board are connected to DALI bus, as shown in Figure 1-6 Demo Board of

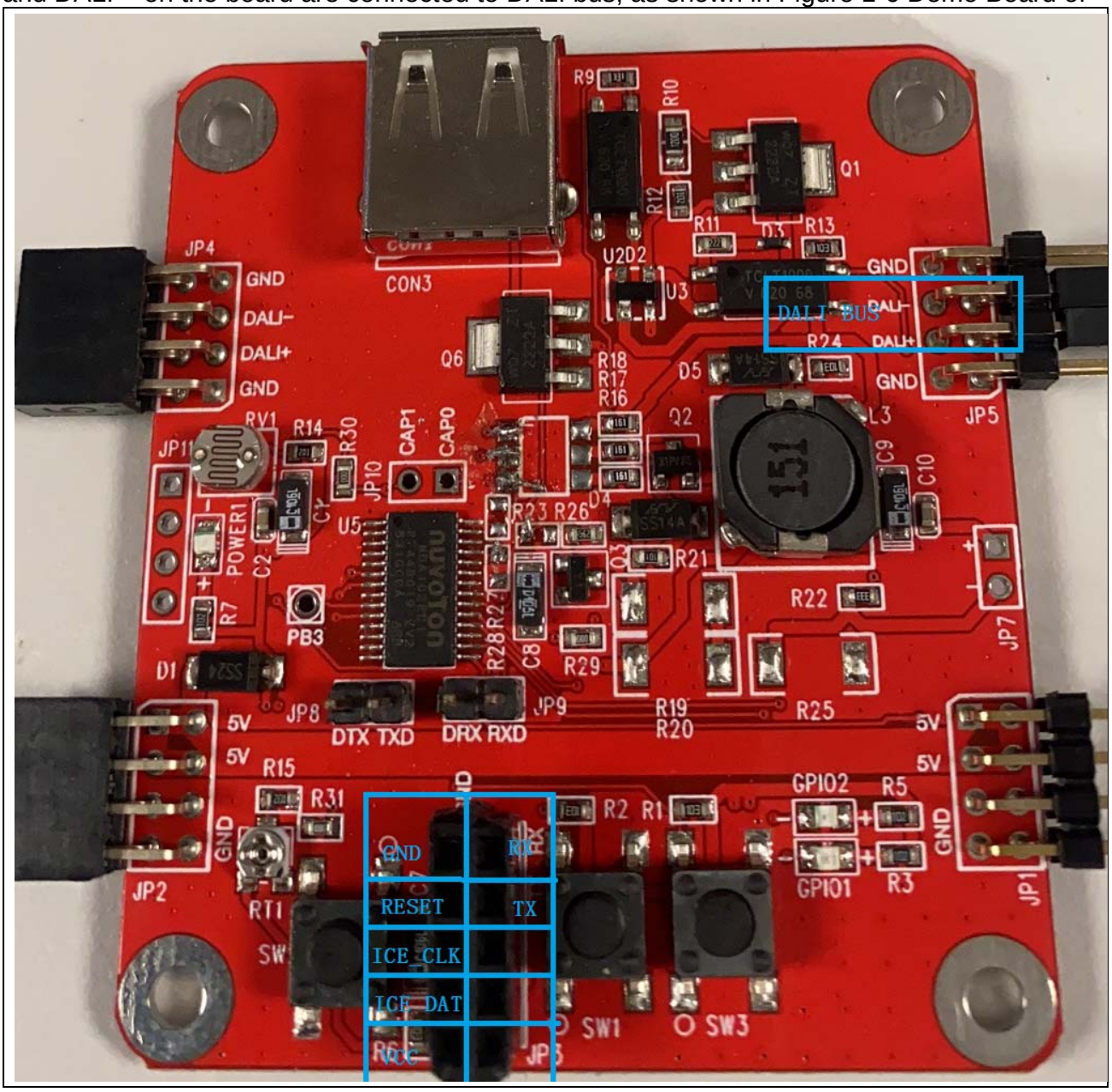

Figure 1-6 Demo Board of Slave

The AS7221 dimming has two modes, one is to output the duty cycle of PWM and the other is the target brightness. Thus, it is necessary to send the serial port instruction in accordance with the mode of AS7221. In this example, the duty cycle is sent by default. To avoid invalid instruction, due to the inconsistency with the mode of AS7221. For example, when sending dimming command to AS7221 but the lamps on AS7221 remain unchanged, switch the mode of AS7221 by pressing SW1, and query the mode of AS7221 by SW3, so as to modify the instruction mode sent to make it consistent with the mode of AS7221.

The fade mode of AS7221 is different from that of DALI protocol, so the fade instruction of DALI

is invalid for AS7221. The fade instruction of AS7221 is set by sending a value to set the fade time through a specific instruction. If you want to set the fade time of AS7221, you can use the extension instruction of DALI and then send it to the slave with parameters. When the command and parameters are recognized by the slave device, they are sent to AS7221 through the serial port.

Note: Currently, this example does not support this function. Since the fade instruction setting of AS7221 needs the support of host operation interface, it also needs to add extension instructions in DALI library. This function can be added later for specific customers.

In the process of dimming, the fade time of the three-color LED lamp on the DALI_SLAVE_NDA102EC1 V1.0 board follows the DALI protocol specification. It is different from the fade time of AS7221, but the fade time of NDA102 and AS7221 can be adjusted by different parameters.

1.3 Demo Result

Host sends 7500K color temperature, as shown in Figure 1-7 Set the Color Temperature to 7500K

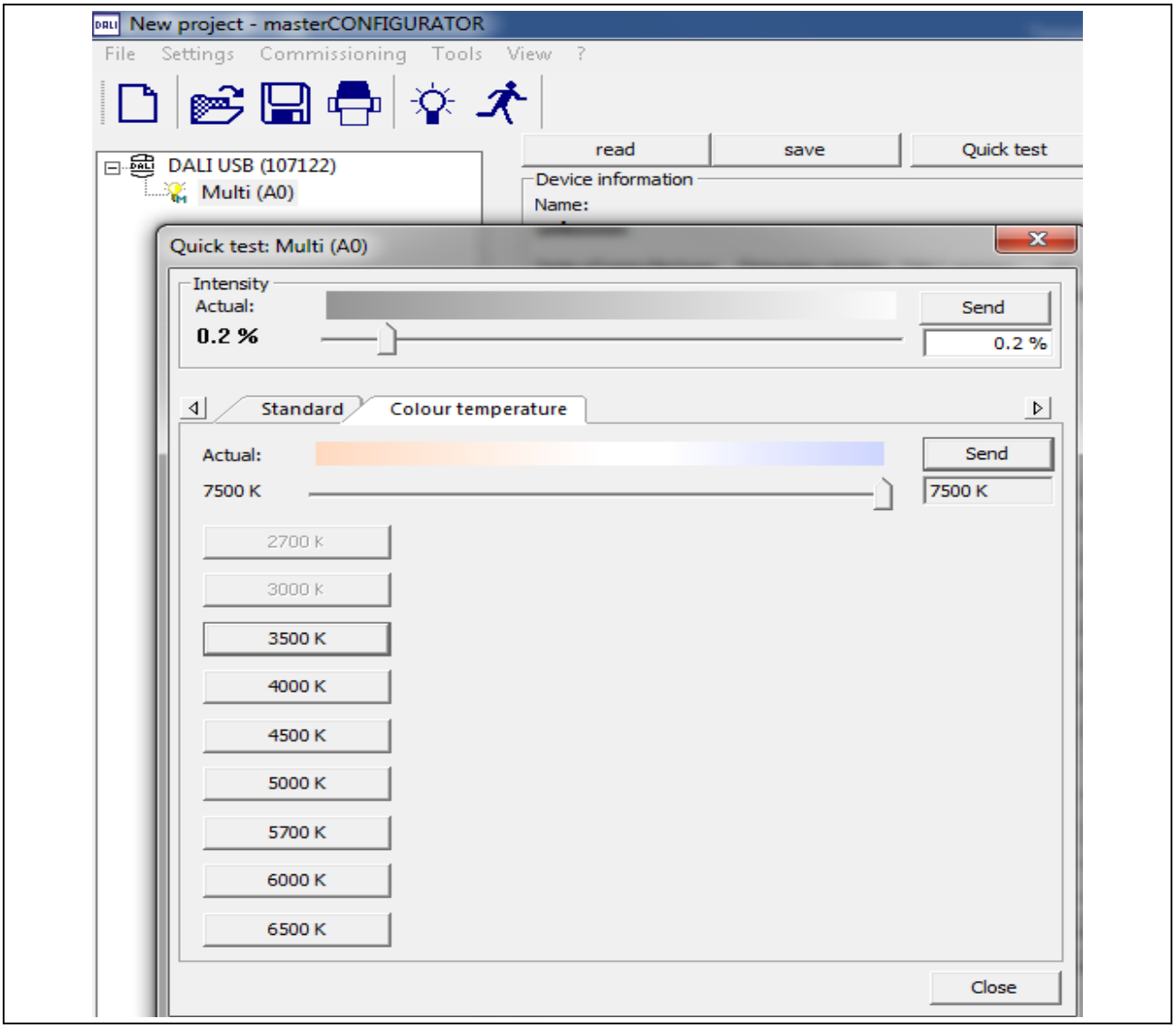

Figure 1-7 Set the Color Temperature to 7500K

The effect of lamps is shown in Figure 1-8 Effect of 7500K.

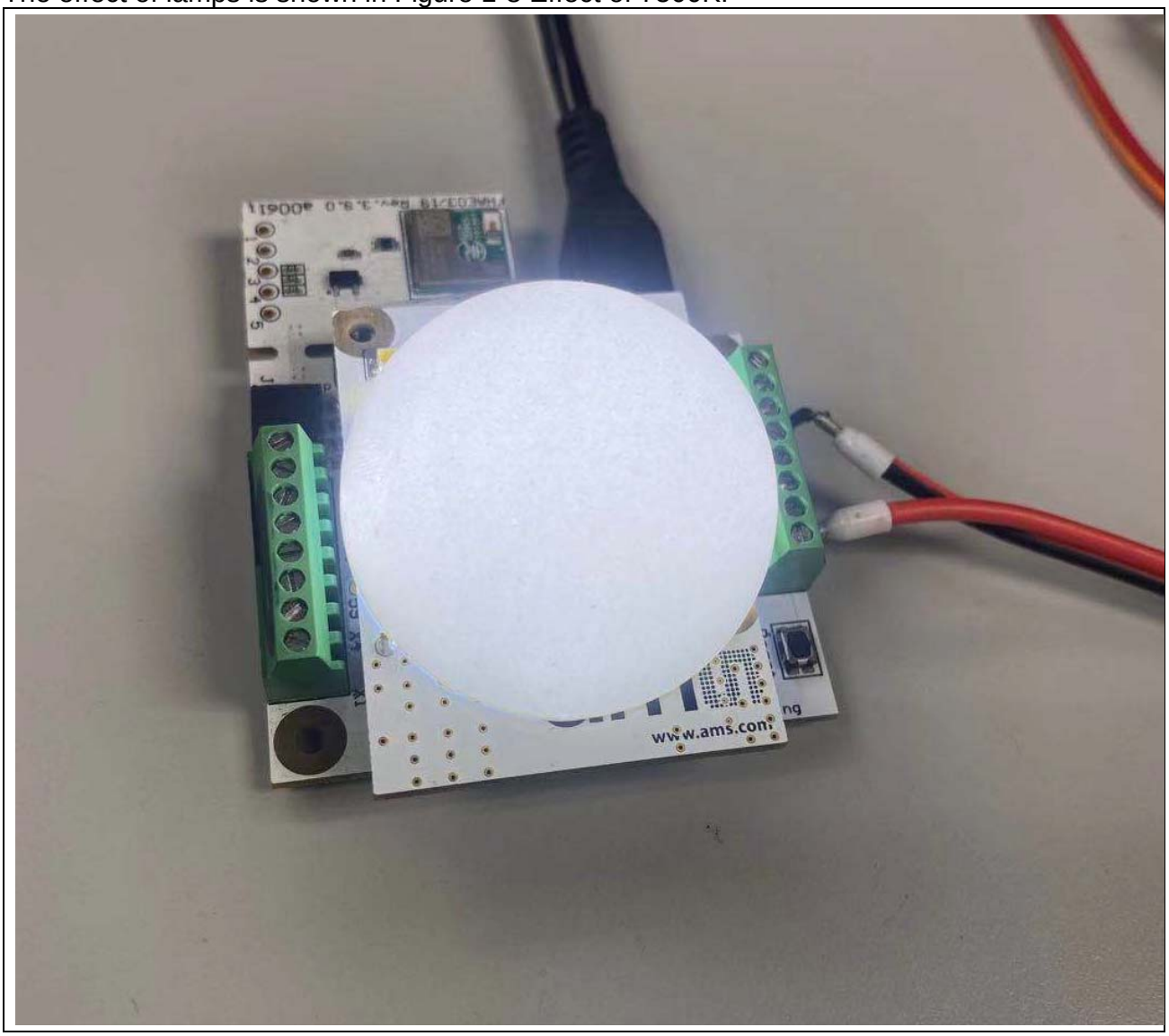

Figure 1-8 Effect of 7500K

Host sends 3000K color temperature, as shown in Figure 1-9 Set the Color Temperature to 3000K

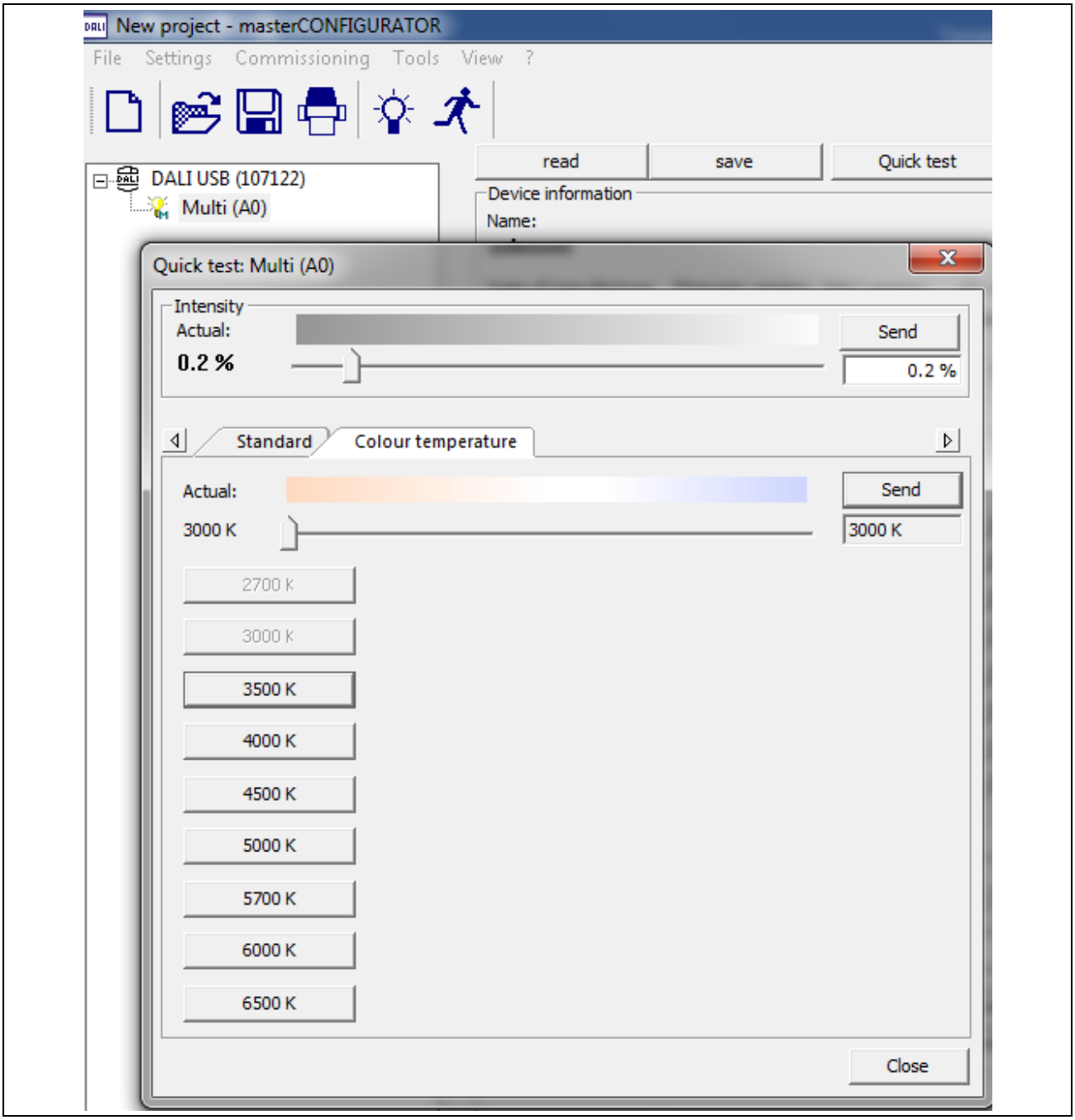

Figure 1-9 Set the Color Temperature to 3000K

The effect of lamps is shown in Figure 1-10 Effect of 3000K

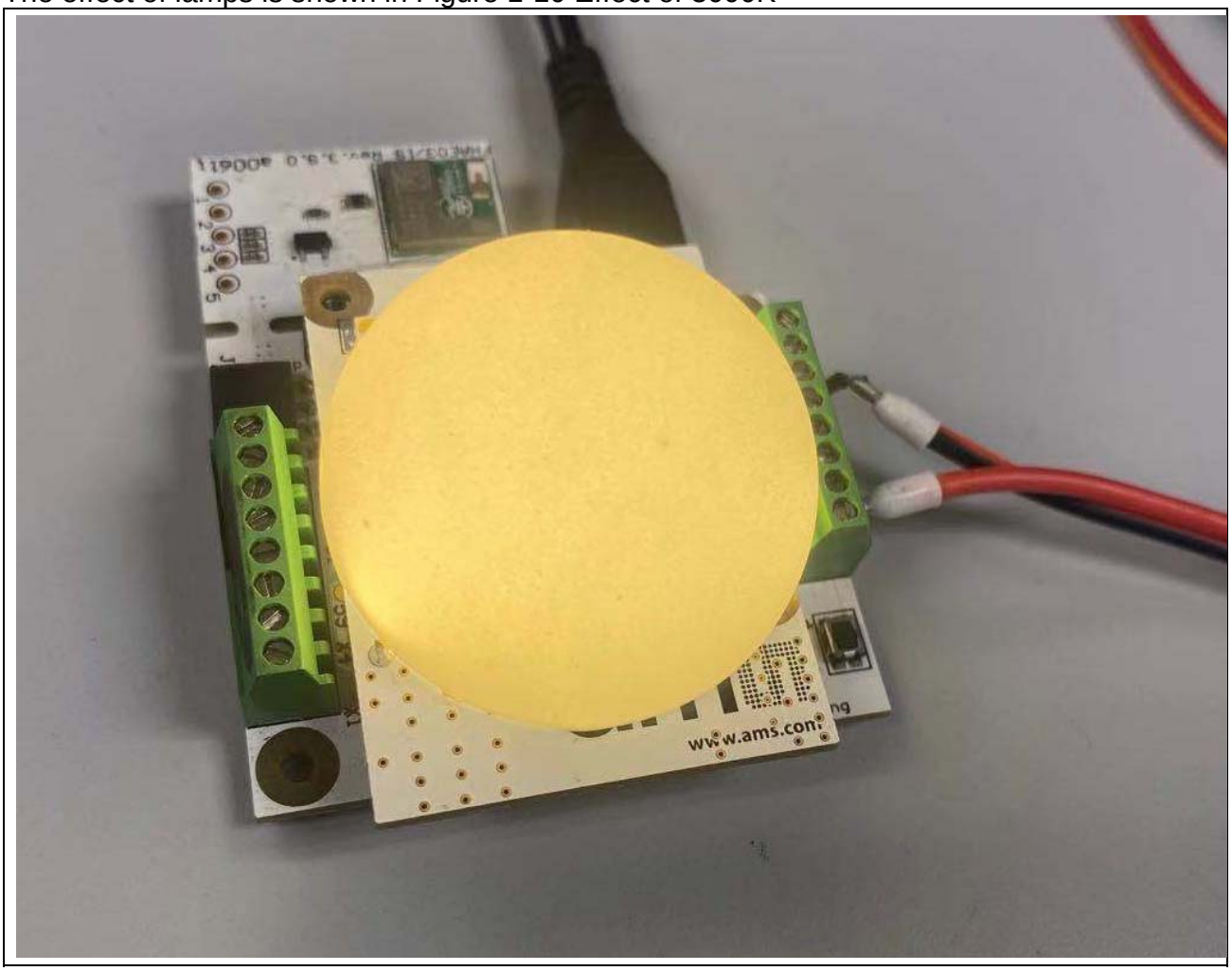

Figure 1-10 Effect of 3000K

2 Code Description

Luminance control interface function, the brightness level or target illumination value to be set is sent to AS7221:

```
void AS7221_Adjust_Brightness(uint8_t level)
{
    if (AS7221_Prelevel != level)
    {
        if (AS7221_ATDL_flag)
        {
            //ATLUXT
            AS7221_Prelevel = AS7221_level = level;
            AS7221_Send_SetCmd(ATLUXT, level);
        }
        else
        {
            //ATDIM
            AS7221_Prelevel = AS7221_level = level;
            AS7221_Send_SetCmd(ATDIM, level);
        }
    }
}
```

The color temperature control interface function, the color temperature value to be set is sent to AS7221:

```
void AS7221_Adjust_Color(uint16_t colorvalue)
{
    uint32_t AS7221_K_colorValue;
    uint32_t AS7221_percentPWM2;

    if (AS7221_precolorTC != colorvalue)
    {
        if (AS7221_ATCT_flag)
        {
            AS7221_colorTC = colorvalue;
            AS7221_precolorTC = colorvalue;
            AS7221_K_colorValue = HDIV_Div(1000000, AS7221_colorTC);
            AS7221_Send_SetCmd(ATCCTT, AS7221_K_colorValue);
        }
        else
```

```
        {
            Tcolorcool = colorvalue - var_TcCoolest;
            Tcolorwarm = var_TcWarmest - colorvalue;
            Color_Range = var_TcWarmest - var_TcCoolest;

            AS7221_percentPWM2 = HDIV_Div(Tcolorwarm * 100, Color_Range);
            AS7221_Send_SetCmd(ATLED23M, AS7221_percentPWM2);
        }
    }
}
```

Command and control interface, the corresponding control command is converted into the corresponding command string and then sent to AS7221 through the serial port:

```
void AS7221_Send_SetCmd(uint8_t cmd, uint32_t value)
{
    uint32_t lightnumb;
    uint32_t numb = 0;

    numb = value;
    memset(charbuf, 0, sizeof(charbuf));

    switch (cmd)
    {
        case ATLIGHT:
            sprintf(charbuf, "ATLIGHT=%d\r\n", numb);
            UUART_Write(UUART1, (uint8_t *)charbuf, ASS7221_Charbuf_Size);
            break;

        case ATDIM:
            lightnumb = value * 100;
            lightnumb = lightnumb >> 8;
            sprintf(charbuf, "ATDIM=%d\r\n", lightnumb);
            UUART_Write(UUART1, (uint8_t *)charbuf, ASS7221_Charbuf_Size);
            break;

        case ATLUXT:
            lightnumb = value * 100;
            sprintf(charbuf, "ATLUXT=%d\r\n", lightnumb);
            UUART_Write(UUART1, (uint8_t *)charbuf, ASS7221_Charbuf_Size);
            break;
```

```
        case ATLED23M:
            lightnumb = value;
            sprintf(charbuf, "ATLED23M=%d\r\n", lightnumb);
            UUART_Write(UUART1, (uint8_t *)charbuf, ASS7221_Charbuf_Size);
            break;

        case ATCCTT:
            lightnumb = value;
            sprintf(charbuf, "ATCCTT=%d\r\n", lightnumb);
            UUART_Write(UUART1, (uint8_t *)charbuf, ASS7221_Charbuf_Size);
            break;

        case ATCT:
            sprintf(charbuf, "ATCT=%d\r\n", value);
            UUART_Write(UUART1, (uint8_t *)charbuf, ASS7221_Charbuf_Size);
            break;

        case Get_ATCT:
            sprintf(charbuf, "ATCT\r\n");
            UUART_Write(UUART1, (uint8_t *)charbuf, ASS7221_Charbuf_Size);
            break;
```

3 Software and Hardware Requirements

3.1 Software Requirements

- BSP version
  - 102_207_209_NDA102_200707_Lib
- IDE version
  - Keil uVersion 5.32

3.2 Hardware Requirements

- Circuit components
  - DALI_SLAVE_NDA102EC1 V1.0
  - TRIDONIC DALI USB
  - a0061i_CSS SLIK AS7221
  - a0062b_CSS SLIK AS3834 LED

- Pin Connect is shown in Figure 3-1 Connection .

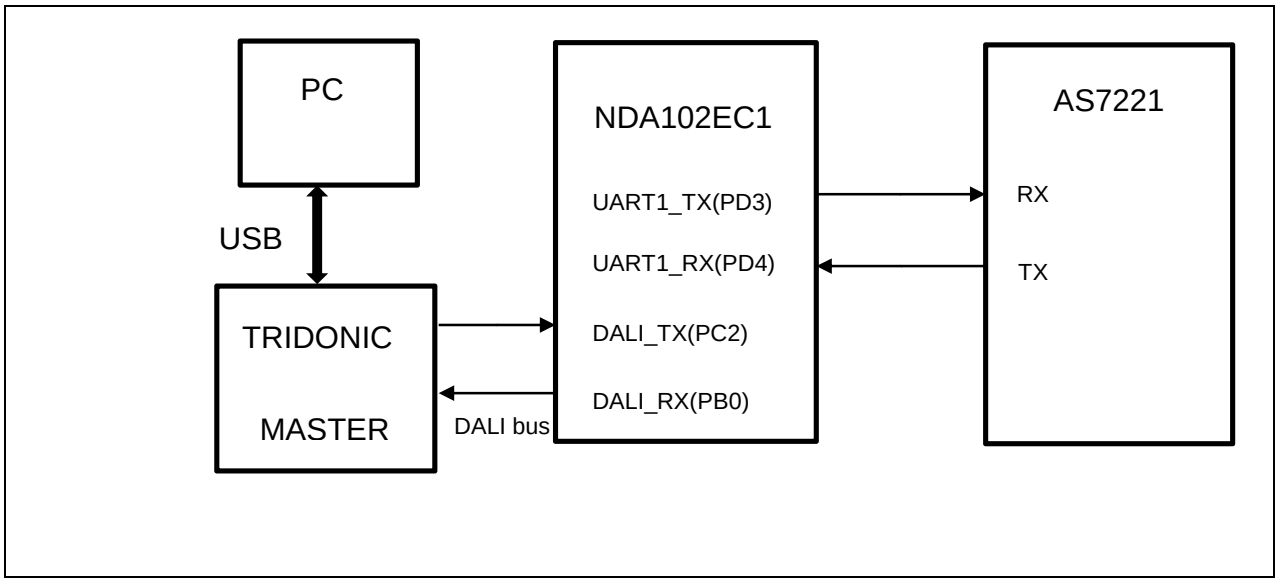

Figure 3-1 Connection Diagram

4 Directory Information

The directory structure as shown in Figure4-1 Directory .

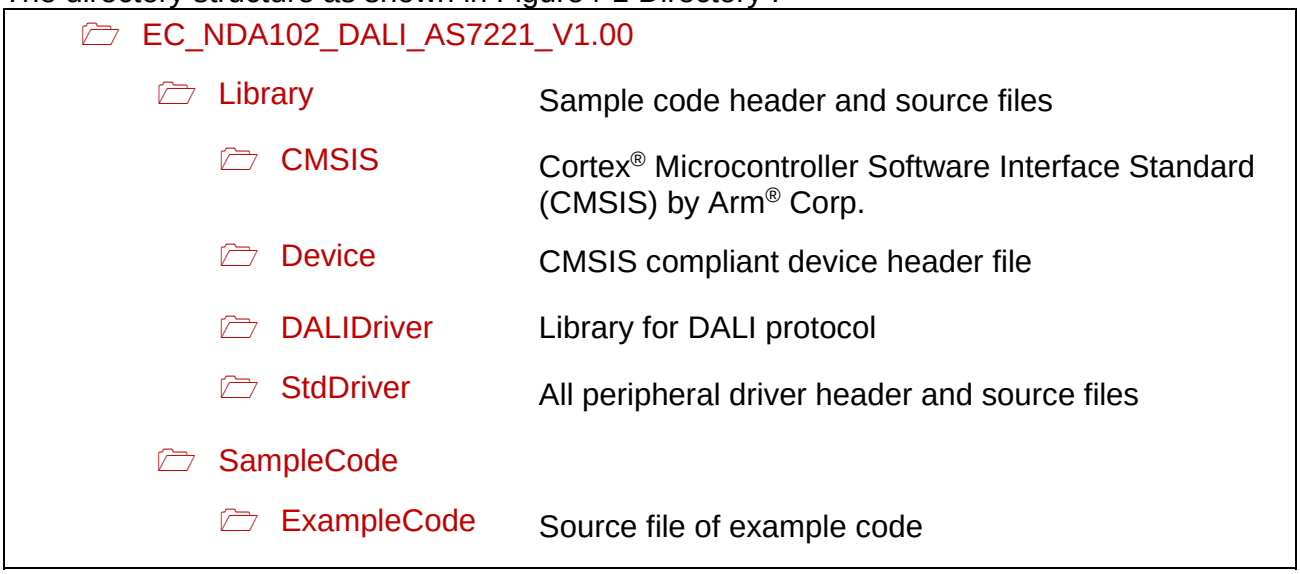

Figure4-1 Directory Structure

5 Example Code Execution

1. Browse the sample code folder as described in the Directory Information section and double-click NDA102_DALI_AS7221.uvprojx.
2. Enter Keil compile mode.
   - Build
   - Download
   - Start/Stop debug session
3. Enter debug mode.
   - Run

6 Revision History

Date	Revision	Description
2021.06.01	1.00	1. Initially issued.

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.