

# S726

## I2C Configuration Guide

Issue 1.0



**Copyright © Neoway Technology Co., Ltd 2022. All rights reserved.**

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Neoway Technology Co., Ltd.

**neoway**有方 is the trademark of Neoway Technology Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

**Notice**

This document provides guide for users to use S726.

This document is intended for system engineers (SEs), development engineers, and test engineers.

THIS GUIDE PROVIDES INSTRUCTIONS FOR CUSTOMERS TO DESIGN THEIR APPLICATIONS. PLEASE FOLLOW THE RULES AND PARAMETERS IN THIS GUIDE TO DESIGN AND COMMISSION. NEOWAY WILL NOT TAKE ANY RESPONSIBILITY OF BODILY HURT OR ASSET LOSS CAUSED BY IMPROPER OPERATIONS.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE DUE TO PRODUCT VERSION UPDATE OR OTHER REASONS.

EVERY EFFORT HAS BEEN MADE IN PREPARATION OF THIS DOCUMENT TO ENSURE ACCURACY OF THE CONTENTS, BUT ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT DO NOT CONSTITUTE A WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

Neoway provides customers complete technical support. If you have any question, please contact your account manager or email to the following email addresses:

Sales@neoway.com

Support@neoway.com

**Website:** <http://www.neoway.com>

# Contents

1 Overview.....	5
2 Pinmap Configuration .....	6
3 DTS Configuration .....	7
3.1 Confirming the Aliases Node.....	7
3.2 Confirming the I2C Node.....	7
3.3 Adding an I2C Device Node .....	7
4 Configuration of I2Cs .....	9
4.1 Configuring I2C-2 .....	9
4.2 Configuring I2C-3 .....	10
4.3 Configuring I2C-4 .....	10
4.4 Configuring I2C-5.....	11

# About This Document

## Scope

This document is applicable to the S726 series.




## Audience

This document is intended for [system engineers \(SEs\)](#), [development engineers](#), and [test engineers](#).

## Change History

Issue	Change	Changed By
1.0	Initial draft	Leo Shen

## Conventions

Symbol	Indication
	Indicates danger or warning. This information must be followed. Otherwise, a catastrophic module or user device failure or bodily injury may occur.
	Indicates caution. This symbol alerts the user to important points about using the module. If these points are not followed, the module or user device may fail.
	Indicates instructions or tips. This symbol provides advices or suggestions that may be useful when using the module.

# 1 Overview

Inter-Integrated Circuit (I2C) is a protocol developed by Philips. Because of its small number of pins, simple hardware implementation and strong scalability, I2C does not require external transceivers that use communication protocols such as USART and CAN. I2C is now widely used for communication between multiple integrated circuits (ICs) within a system.

The S726 platform provides four I2Cs that can be used by customers as desired: I2C-2, I2C-3, I2C-4, and I2C-5.

I2C ID	Pin	GPIO ID	Remarks
I2c-2	pin-91, pin-92	gpio127, gpio128	Used as the sensor I2C by default.
I2c-3	pin-47, pin-48	gpio146, gpio147	Used as the touchscreen I2C by default.
I2c-4	pin-168, pin-167	gpio154, gpio155	Default SIM2 function
I2c-5	pin-168, pin-167	gpio38, gpio39	Used as UART5 by default.

I2Cs must be configured in the pinmap and device tree. For the specific configuration method, see the following steps.

## 2 Pinmap Configuration

S726 uses the pinmap to configure the pin functions. To use the I2C function, you need to check the table below and configure the I2C function for the corresponding pin.

Path of the pinmap file:

bsp\bootloader\u-boot15\board\spreadtrum\S726\pinmap-sp9863a.c

Cell Name	Pin Name	Function0	Type	Function1	Type	Function2	Type	Function3	Type
SPSCBC2_8X_I IC_HL	SCL2	SCL2	I/O /T					GPIO 127	I/O /T
SPSCBC2_8X_I IC_HL	SDA2	SDA2	I/O /T					GPIO 128	I/O /T
SPSCBC2_8X_ W_IIC_HL	SCL3	SCL3	I/O /T			EXT_XTL _EN0	I	GPIO 146	I/O /T
SPSCBC2_8X_ W_IIC_HL	SDA3	SDA3	I/O /T					GPIO 147	I/O /T
SPPDWUWSW SIM_V	SIMCLK2	SIMCLK2	O	SCL4	I/O /T	SE_GPIO1 1	I/O /T	GPIO 154	I/O /T
SPPDWUWSW SIM_V	SIMDAT2	SIMDAT2	I/O /T	SDA4	I/O /T	SE_GPIO1 2	I/O /T	GPIO 155	I/O /T
SPSCBC2_8X_I IC_HL	U5TXD	U5TXD	O	SCL5	I/O /T			GPIO 38	I/O /T
SPSCBC2_8X_I IC_HL	U5RXD	U5RXD	I	SDA5	I/O /T			GPIO 39	I/O /T

## 3 DTS Configuration

### 3.1 Confirming the Aliases Node

Add the configuration of the I2C node (for example I2C-3) under the aliases node. Open the following file:

bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\sharkl3.dtsi

Add the following code:

```
aliases {  
  
    i2c3 = &i2c3;  
  
};
```

### 3.2 Confirming the I2C Node

Add the configuration of the I2C node (for example I2C-3) under the SOC node. The file location is as follows:

bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\sharkl3.dtsi

Generally, the I2C node is already configured, and you only need to check and confirm the configuration.

```
i2c3: i2c@70800000 {  
    compatible = "sprd,sharkl3-i2c";  
    reg = <0x70800000 0x1000>; /*i2c register address*/  
    interrupts = <GIC_SPI 14 IRQ_TYPE_LEVEL_HIGH>;  
    clock-names = "enable","i2c", "source"; /*i2c clock configuration*/  
    clock-frequency = <400000>; /*i2c master mode clock frequency*/  
    #address-cells = <1>;  
    #size-cells = <0>;  
    status = "disabled"; /*enable i2c when using: okay*/  
};
```

### 3.3 Adding an I2C Device Node

After configuring the I2C node, add the corresponding I2C device node, for example, touchscreen node. Open the following file:

bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\S726\_uis8581e5h10-overlay.dts

Configure the following code:

```
&i2c3 {
    status = "okay"; /*enable i2c: okay*/
    goodix@14 {
        compatible = "goodix,gt1x";
        reg = <0x14>; /*7-bit slave device*/
        goodix,irq-gpio = <&ap_gpio 144 GPIO_ACTIVE_HIGH>;
        goodix,reset-gpio = <&ap_gpio 145 GPIO_ACTIVE_HIGH>;
    };
};
```



## 4 Configuration of I2Cs

### 4.1 Configuring I2C-2

**Step 1:** Configure the pinmap. Check the function table, which shows that function 0 is I2C-2. Open the following file:

```
bsp\bootloader\u-boot15\board\spreadtrum\S726\pinmap-sp9863a.c
```

Configure the following code:

```
// i2c-2, scl
{REG_PIN_SCL2,      BITS_PIN_AF(0)},
{REG_MISC_PIN_SCL2,
BITS_PIN_DS(1)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_CM4|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
// i2c-2, sda
{REG_PIN_SDA2,      BITS_PIN_AF(0)},
{REG_MISC_PIN_SDA2,
BITS_PIN_DS(1)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_CM4|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
```

**Step 2:** Confirm the aliases node, and open the following file:

```
bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\sharkl3.dtsi
```

confirm the following code:

```
aliases {
...
    i2c2 = &i2c2;
...
};
```

```
i2c2: i2c@70700000 {
    compatible = "sprd,sharkl3-i2c";
    reg = <0x70700000 0x1000>;
    interrupts = <GIC_SPI 13 IRQ_TYPE_LEVEL_HIGH>;
    clock-names = "enable","i2c", "source";
    clock-frequency = <100000>;
    #address-cells = <1>;
    #size-cells = <0>;
    status = "disabled";
};
```

## 4.2 Configuring I2C-3

**Step 1:** Configure the pinmap. Check the function table, which shows that function 0 is I2C-3. Open the following file:

bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\sharkl3.dtsi

Configure the following code:

```
// i2c-3, scl
{REG_PIN_SCL3,      BITS_PIN_AF(0)},
{REG_MISC_PIN_SCL3,
BITS_PIN_DS(3)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_AP|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
// i2c-3, sda
{REG_PIN_SDA3,      BITS_PIN_AF(0)},
{REG_MISC_PIN_SDA3,
BITS_PIN_DS(3)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_AP|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
```

**Step 2:** Confirm the aliases node, and open the following file:

bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\sharkl3.dtsi

Confirm the following code:

```
aliases {
...
    i2c3 = &i2c3;
...
};
```

```
i2c3: i2c@70800000 {
    compatible = "sprd,sharkl3-i2c";
    reg = <0x70800000 0x1000>;
    interrupts = <GIC_SPI 14 IRQ_TYPE_LEVEL_HIGH>;
    clock-names = "enable","i2c", "source";
    clock-frequency = <400000>;
    #address-cells = <1>;
    #size-cells = <0>;
    status = "disabled";
};
```

## 4.3 Configuring I2C-4

**Step 1:** Configure the pinmap. Check the function table, which shows that function 1 is I2C-4. Open the following file:

bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\sharkl3.dtsi

Configure the following code:

```
// I2C-4, scl
{REG_PIN_SIMCLK0,   BITS_PIN_AF(1)},
```

```
{REG_MISC_PIN_SIMCLK0,
BITS_PIN_DS(1)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_AP|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
// I2C-4, sda
{REG_PIN_SD2_D2,          BITS_PIN_AF(1)},
{REG_MISC_PIN_SD2_D2,
BITS_PIN_DS(1)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_AP|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
```

**Step 2:** Confirm the aliases node, and open the following file:

bsp\kernel\kernel4.14\arch\arm64\boot\dtb\sprd\sharkl3.dtsi

Confirm the following code:

```
aliases {
...
    i2c4 = &i2c4;
...
};
```

```
i2c4: i2c@70900000 {
    compatible = "sprd,sharkl3-i2c";
    reg = <0x70900000 0x1000>;
    interrupts = <GIC_SPI 15 IRQ_TYPE_LEVEL_HIGH>;
    clock-names = "enable","i2c", "source";
    clock-frequency = <100000>;
    #address-cells = <1>;
    #size-cells = <0>;
    status = "disabled";
};
```

## 4.4 Configuring I2C-5

**Step 1:** Configure the pinmap. Check the function table, which shows that function 1 is I2C-5. Open the following file

bsp\kernel\kernel4.14\arch\arm64\boot\dtb\sprd\sharkl3.dtsi

Configure the following code:

```
// I2C-5, scl
{REG_PIN_U5TXD,          BITS_PIN_AF(1)},
{REG_MISC_PIN_U5TXD,
BITS_PIN_DS(1)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_AP|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
// I2C-5, sda
{REG_PIN_U5RXD,          BITS_PIN_AF(1)},
{REG_MISC_PIN_U5RXD,
BITS_PIN_DS(1)|BIT_PIN_WPUS|BIT_PIN_WPU|BIT_PIN_SLP_AP|BIT_PIN_SLP_WPU|BIT_PIN_SLP_Z},
```

**Step 2:** Confirm the aliases node, and open the following file:

bsp\kernel\kernel4.14\arch\arm64\boot\dts\sprd\sharkl3.dtsi

Confirm the following code:

```
aliases {  
...  
    i2c5 = &i2c5;  
...  
};
```

```
i2c5: i2c@71500000 {  
    compatible = "sprd,sharkl3-i2c";  
    reg = <0x71500000 0x1000>;  
    interrupts = <GIC_SPI 116 IRQ_TYPE_LEVEL_HIGH>;  
    clock-names = "enable","i2c", "source";  
    clock-frequency = <100000>;  
    #address-cells = <1>;  
    #size-cells = <0>;  
    status = "disabled";  
};
```