

GigaDevice Semiconductor Inc.

GD32F527V-START

Arm[®] Cortex[®]-M33 32-bit MCU

User Guide

Revision 1.2

(Aug. 2025)

Table of Contents

TABLE OF CONTENTS.....	1
LIST OF FIGURES	3
LIST OF TABLES	4
1. SUMMARY	5
2. FUNCTION PIN ASSIGNMENT	5
3. GETTING STARTED	5
4. HARDWARE LAYOUT OVERVIEW	7
4.1. Power supply	7
4.2. Boot option	7
4.3. LED	8
4.4. KEY	8
4.5. USART	8
4.6. USBFS	9
4.7. USBHS	10
4.8. Extension	10
4.9. GD-Link	11
4.10. MCU	12
4.11. Arduino	13
5. ROUTINE USE GUIDE	14
5.1. GPIO_Running_LED	14
5.1.1. DEMO purpose	14
5.1.2. DEMO running result	14
5.2. GPIO_Key_Polling_mode	14
5.2.1. DEMO purpose	14
5.2.2. DEMO running result	14
5.3. EXTI_Key_Interrupt_mode	15
5.3.1. DEMO purpose	15
5.3.2. DEMO running result	15
5.4. USART_HyperTerminal_Interrupt	15
5.4.1. DEMO purpose	15
5.4.2. DEMO running result	15
5.5. TIMER_Key_EXTI	16
5.5.1. DEMO purpose	16
5.5.2. DEMO running result	16
5.6. USB_MSC_Device	16
5.6.1. DEMO Purpose	16
5.6.2. DEMO Running Result	17

5.7. USB_MSC_Host	18
5.7.1. DEMO purpose	18
5.7.2. DEMO running result.....	18
6. REVISION HISTORY	20

List of Figures

Figure 4-1 Schematic diagram of power supply.....	7
Figure 4-2 Schematic diagram of boot option	7
Figure 4-3 Schematic diagram of LED function	8
Figure 4-4 Schematic diagram of Key function	8
Figure 4-5 Schematic diagram of USART function	8
Figure 4-6 Schematic diagram of USBFS function	9
Figure 4-7 Schematic diagram of USBHS function	10
Figure 4-8 Schematic diagram of Extension Pin	10
Figure 4-9 Schematic diagram of GD-Link.....	11
Figure 4-10 Schematic diagram of MCU.....	12
Figure 4-11 Schematic diagram of Arduino	13

List of Tables

Table 2-1 Function pin assignment	5
Table 6-1 Revision history	20

1. Summary

GD32F527V-START uses GD32F527VST6 as the main controller. It uses GD-Link Mini USB interface to supply 5V power. SWD, Reset, Boot, User button key, LED, USART, USBFS, USBHS, Arduino, GD-Link and Extension Pins are also included. For more details please refer to GD32F527V-START-V1.0 schematic.

2. Function pin assignment

Table 2-1 Function pin assignment

Function	Pin	Description
LED	PB11	LED1
	PB12	LED2
	PB14	LED3
	PD8	LED4
RESET		K1-Reset
KEY	PA0	K2-Wakeup
USART	PC6	USART5_TX
	PC7	USART5_RX
USB_FS	PA9	USBFS_VBUS
	PA11	USBFS_DM
	PA12	USBFS_DP
	PA10	USBFS_ID
USB_HS	PC3	USB_HS_ULPI_NXT
	PC2	USB_HS_ULPI_DIR
	PC0	USB_HS_ULPI_STP
	PA5	USB_HS_ULPI_CK
	PB5	USB_HS_ULPI_D7
	PB13	USB_HS_ULPI_D6
	PB12	USB_HS_ULPI_D5
	PB11	USB_HS_ULPI_D4
	PB10	USB_HS_ULPI_D3
	PB1	USB_HS_ULPI_D2
	PB0	USB_HS_ULPI_D1
	PA3	USB_HS_ULPI_D0

3. Getting started

The START board uses GD-Link Mini USB connector to get power DC +5V, which is the hardware system normal work voltage. A GD-Link on board is necessary in order to

download and debug programs. Select the correct boot mode and then power on, the LEDPWR will turn on, which indicates that the power supply is OK.

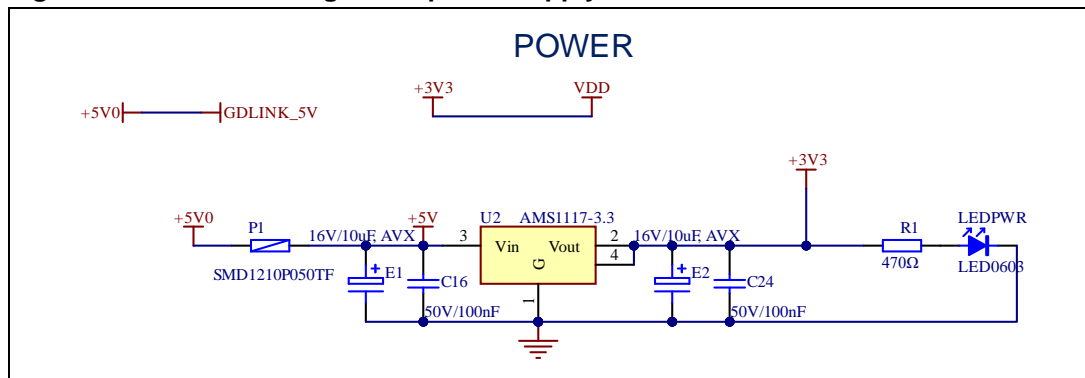
There are Keil version and IAR version of all projects. Keil version of the projects are created based on Keil MDK-ARM 5.28 uVision5. IAR version of the projects are created based on IAR Embedded Workbench for ARM 8.32.1. During use, the following points should be noted:

1. If you use Keil uVision5 to open the project, you need to install the latest version of GigaDevice.GigaDevice.GD32F527_DFP to load the related files.
2. If you use IAR to open the project, you need to install the latest version of IAR_GD32F527_ADDON to load the related files.

4. Hardware layout overview

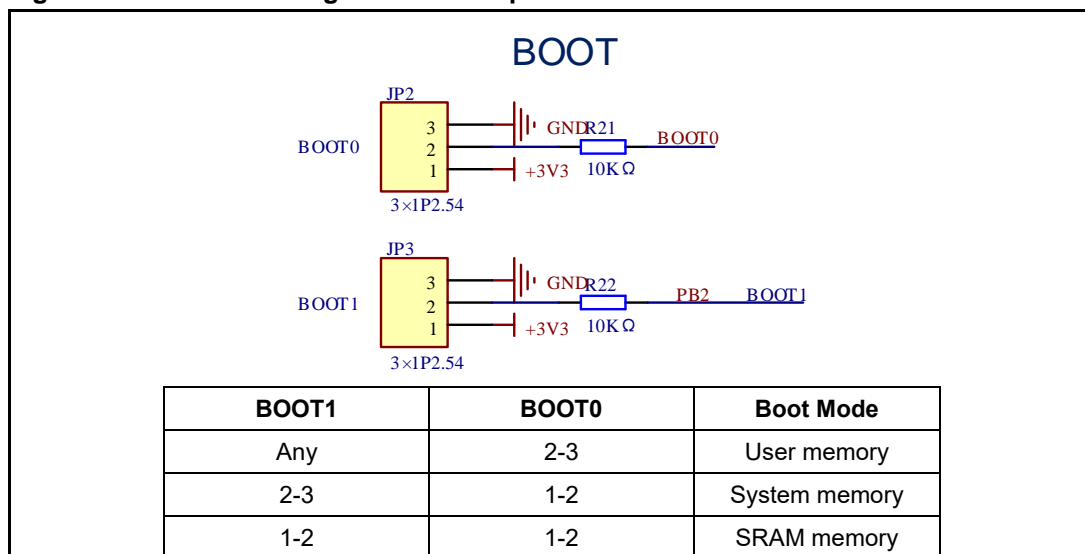
4.1. Power supply

Figure 4-1 Schematic diagram of power supply



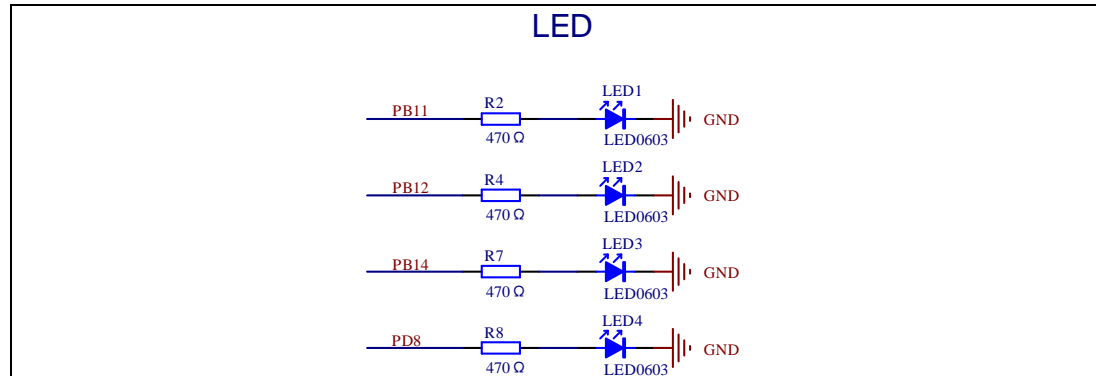
4.2. Boot option

Figure 4-2 Schematic diagram of boot option



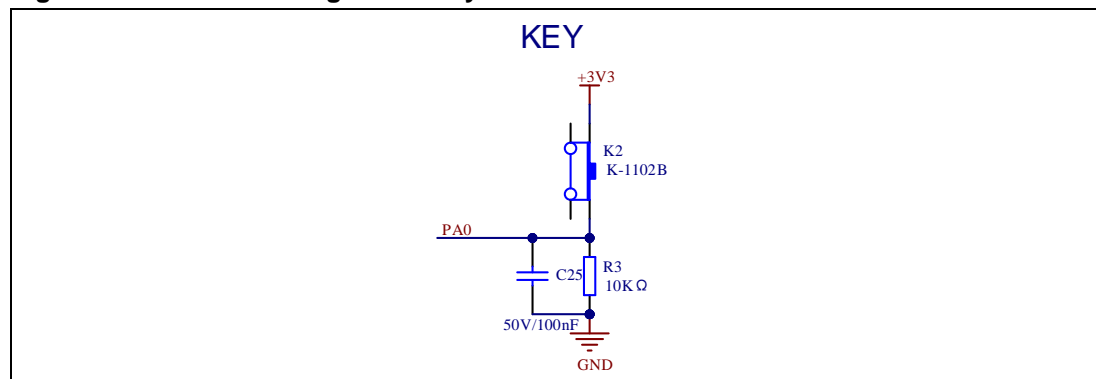
4.3. LED

Figure 4-3 Schematic diagram of LED function



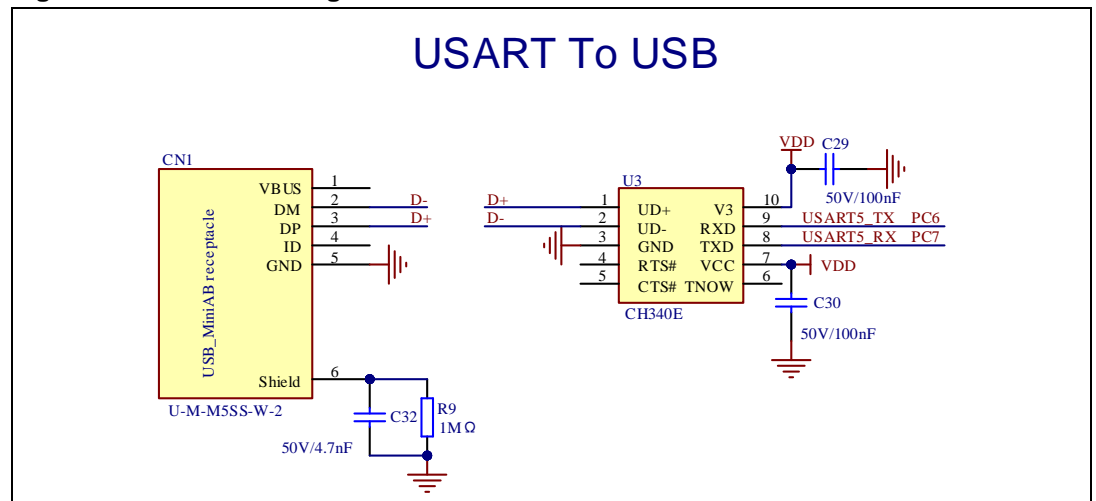
4.4. KEY

Figure 4-4 Schematic diagram of Key function



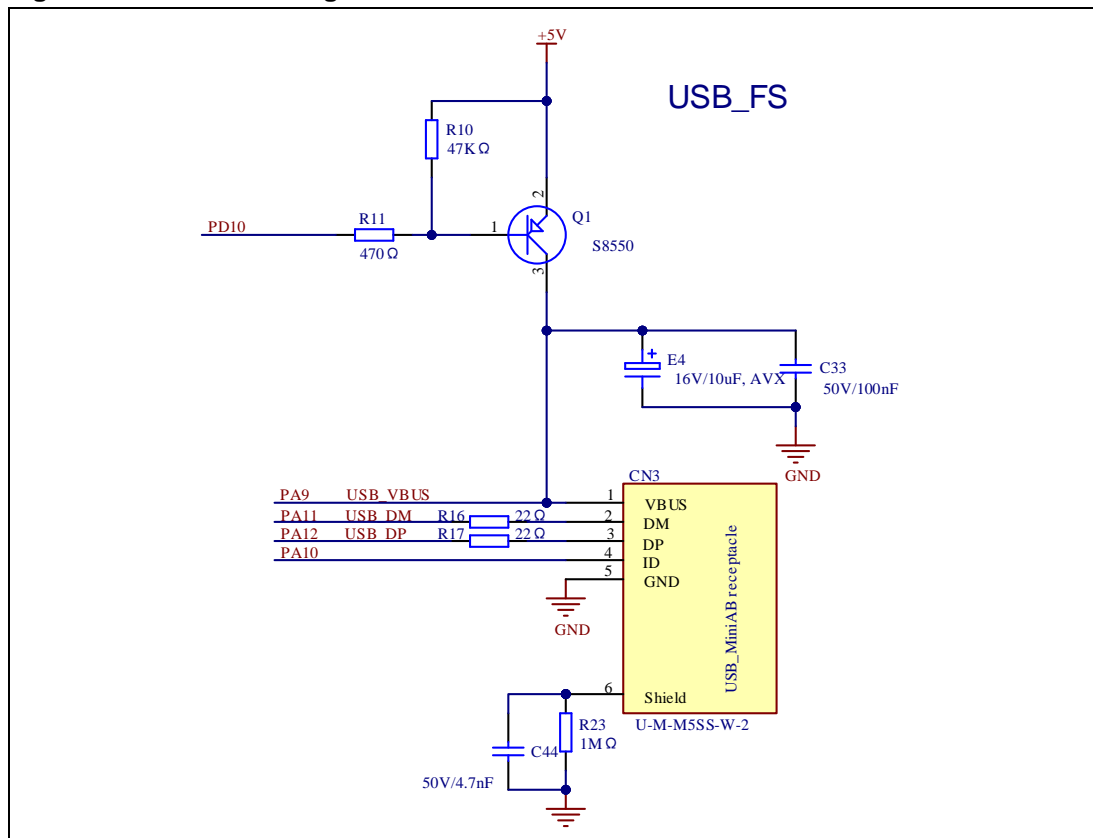
4.5. USART

Figure 4-5 Schematic diagram of USART function



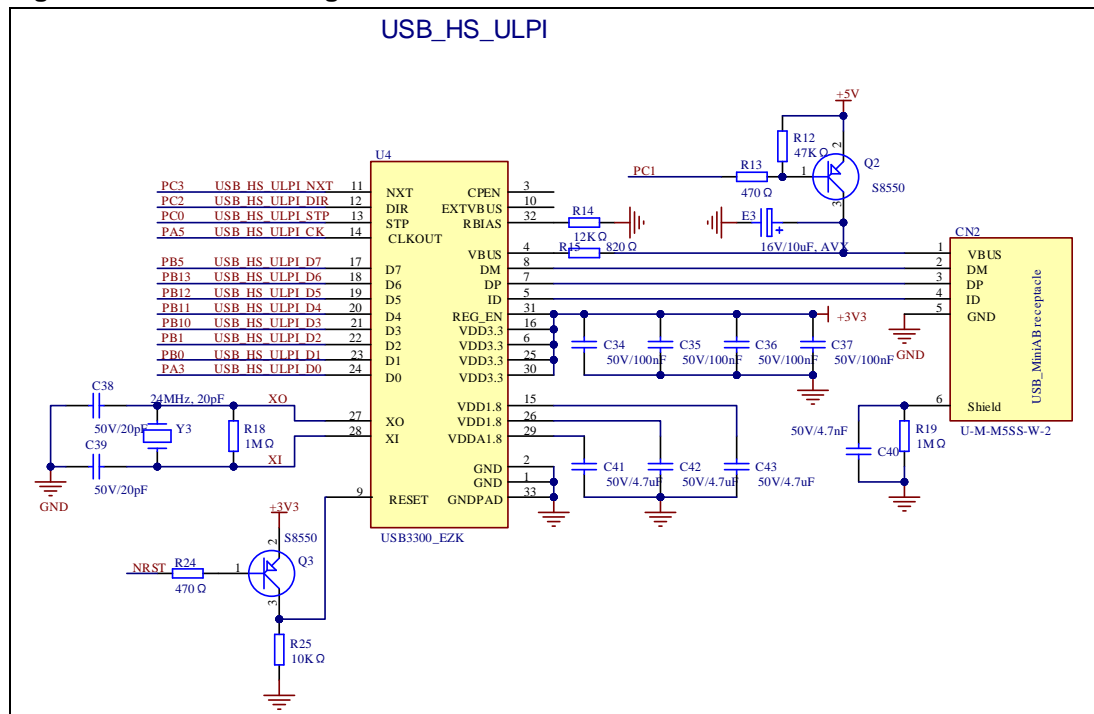
4.6. USBFS

Figure 4-6 Schematic diagram of USBFS function



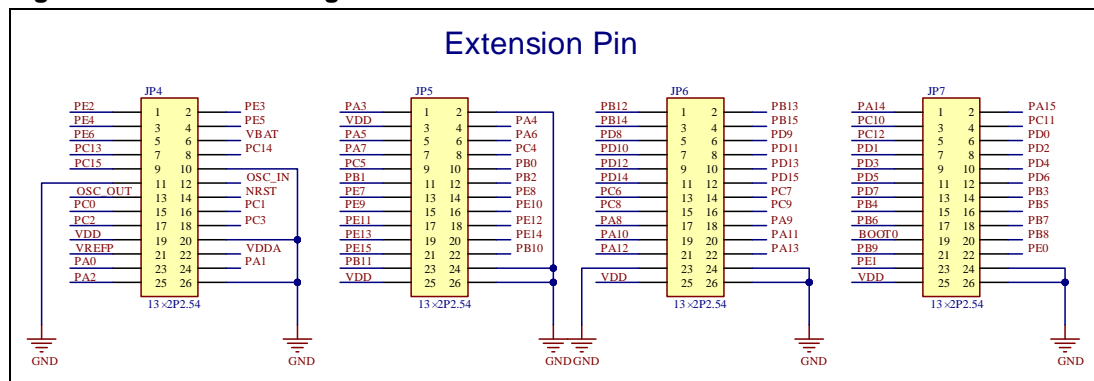
4.7. USBHS

Figure 4-7 Schematic diagram of USBHS function



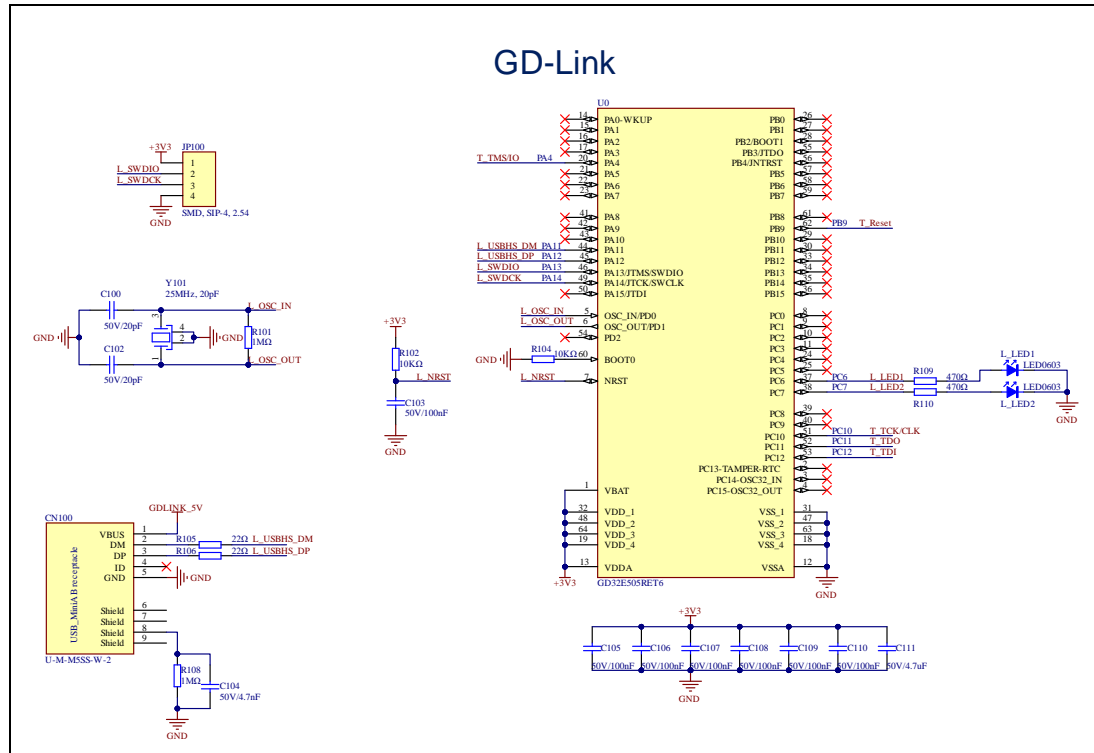
4.8. Extension

Figure 4-8 Schematic diagram of Extension Pin



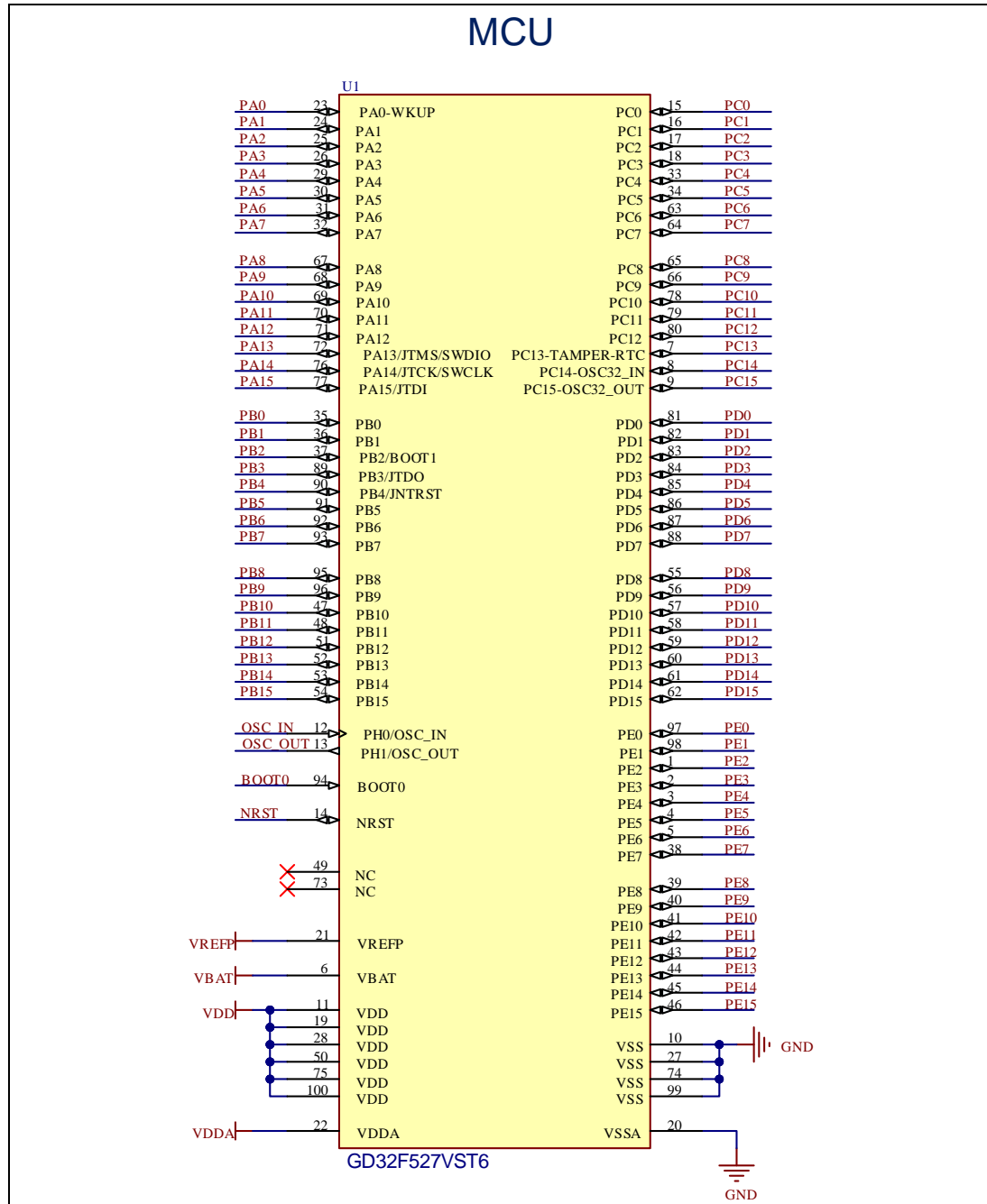
4.9. GD-Link

Figure 4-9 Schematic diagram of GD-Link



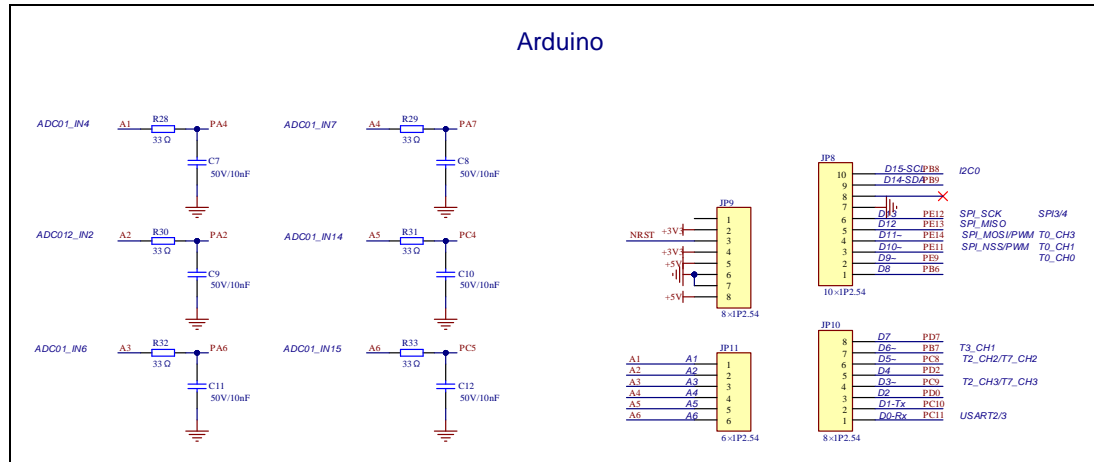
4.10. MCU

Figure 4-10 Schematic diagram of MCU



4.11. Arduino

Figure 4-11 Schematic diagram of Arduino



5. Routine use guide

5.1. GPIO_Running_LED

5.1.1. DEMO purpose

This demo includes the following functions of GD32 MCU:

- Learn to use GPIO control the LED.
- Learn to use SysTick to generate 1ms delay.

GD32F527V-START board has four LEDs. The LED1, LED2 LED3 and LED4 are controlled by GPIO. This demo will show how to light the LEDs.

5.1.2. DEMO running result

Download the program <01_GPIO_Running_LED> to the START board, LED1, LED2 LED3 and LED4 will turn on in sequence with interval of 1000ms, and repeat the process.

5.2. GPIO_Key_Polling_mode

5.2.1. DEMO purpose

This demo includes the following functions of GD32 MCU:

- Learn to use GPIO control the LED and the Key.
- Learn to use SysTick to generate 1ms delay.

GD32F527V-START board has two keys and four LEDs. The two keys are Reset key and Wakeup key. The LED1, LED2, LED3 and LED4 are controlled by GPIO.

This demo will show how to use the Wakeup key to control the LED1. When press down the Wakeup Key, it will check the input value of the IO port. If the value is 1 and will wait for 100ms. Check the input value of the IO port again. If the value still is 1, it indicates that the button is pressed successfully and toggle LED1.

5.2.2. DEMO running result

Download the program <02_GPIO_Key_Polling_mode> to the START board, Press down the Wakeup Key, LED1 will be turned on. Press down the Wakeup Key again, LED1 will be turned off.

5.3. EXTI_Key_Interrupt_mode

5.3.1. DEMO purpose

This demo includes the following functions of GD32 MCU:

- Learn to use GPIO control the LED and the KEY.
- Learn to use EXTI to generate external interrupt.

GD32F527V-START board has two keys and four LEDs. The two keys are Reset key, and Wakeup key. The LED1, LED2, LED3 and LED4 are controlled by GPIO.

This demo will show how to use the EXTI interrupt line to control the LED2. When press down the Wakeup Key, it will produce an interrupt. In the interrupt service function, the demo will toggle LED2.

5.3.2. DEMO running result

Download the program <03_EXTI_Key_Interrupt_mode> to the START board, LED2 is turned on and off for test. Press down the Wakeup Key, LED2 will be turned on. Press down the Wakeup Key again, LED2 will be turned off.

5.4. USART_HyperTerminal_Interrupt

5.4.1. DEMO purpose

This demo includes the following functions of GD32 MCU:

- Learn to use the USART transmit and receive interrupts to communicate with the serial terminal tool.

5.4.2. DEMO running result

Download the program < 04_USART_HyperTerminal_Interrupt > to the START board, connect serial cable to USART. Firstly, all the LEDs flash 2 times for test. Then, the USART sends the tx_buffer array (from 0x00 to 0xFF) to the serial terminal tool supporting hex format communication and waits for receiving data of BUFFER_SIZE bytes from the serial terminal. The data MCU has received is stored in the rx_buffer array. After that, compare tx_buffer with rx_buffer. If tx_buffer is same with rx_buffer, LED1, LED2, LED3, LED4 flash by turns. Otherwise, LED1, LED2, LED3 and LED4 turn on.

The output information via the serial port is as following.

```

00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B
1C 1D 1E 1F 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35 36 37
38 39 3A 3B 3C 3D 3E 3F 40 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53
54 55 56 57 58 59 5A 5B 5C 5D 5E 5F 60 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F
70 71 72 73 74 75 76 77 78 79 7A 7B 7C 7D 7E 7F 80 81 82 83 84 85 86 87 88 89 8A 8B
8C 8D 8E 8F 90 91 92 93 94 95 96 97 98 99 9A 9B 9C 9D 9E 9F A0 A1 A2 A3 A4 A5 A6 A7
A8 A9 AA AB AC AD AE AF B0 B1 B2 B3 B4 B5 B6 B7 B8 B9 BA BB BC BD BE BF C0 C1 C2 C3
C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF
E0 E1 E2 E3 E4 E5 E6 E7 E8 E9 EA EB EC ED EE EF F0 F1 F2 F3 F4 F5 F6 F7 F8 F9 FA FB
FC FD FE FF

```

5.5. TIMER_Key_EXTI

5.5.1. DEMO purpose

This demo includes the following functions of GD32 MCU:

- Learn to use EXTI to generate external interrupt
- Learn to use TIMER to generate PWM

This demo will show how to use the TIMER PWM to trigger EXTI interrupt to toggle the state of LED1 and EXTI interrupt line to control the LED1. When press down the Wakeup key, it will produce an interrupt. In the interrupt service function, the demo will toggle LED1.

5.5.2. DEMO running result

Download the program <05_TIMER_Key_EXTI> to the START board, the LED1 is flashed once for test, press down the Wakeup key, LED1 will be turned on. Press down the Wakeup key again, LED1 will be turned off. Connect PA6 (TIMER2_CH0) and PB11(LED1) with DuPont line. The LED1 will be toggled every 500ms.

5.6. USB_MSC_Device

5.6.1. DEMO Purpose

This demo includes the following functions of GD32 MCU:

- Learn how to use the USBFS/USBHS peripheral mode
- Learn how to implement USB MSC (mass storage) device

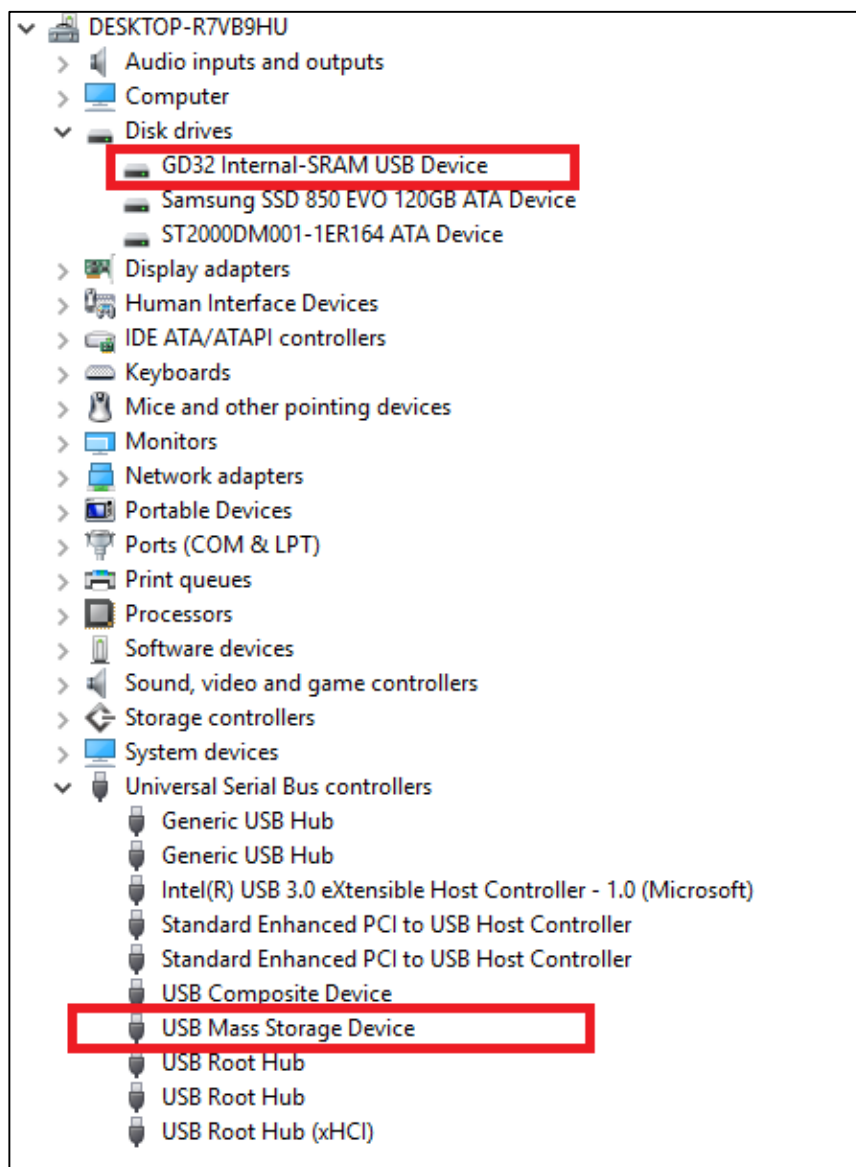
This demo mainly implements a U disk. U disk is currently very widely used removable MSC devices. MSC, the Mass Storage Device Class, is a transport protocol between a computer and mobile devices, which allow a universal serial bus (USB) equipment to access a host computing device, file transfer between them, mainly including mobile hard disk, mobile U disk drive, etc... The MSC device must have a storage medium, and this Demo uses the MCU's internal SRAM as the storage medium. For more details of the MSC protocol please refer to the MSC protocol standard.

MSC device will use a variety of transport protocols and command formats for communication, so it need to choose the appropriate protocol and command format in the realization of the

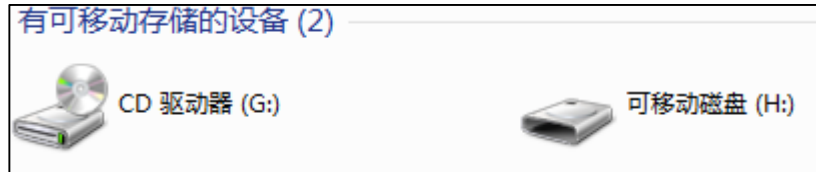
application. This Demo selects the BOT (bulk only transport) protocol and the required SCSI (small computer interface) command, and is compatible with a wide variety of Window operating systems. Specific BOT protocol and SCSI command specification please refer to the standard of their agreement.

5.6.2. DEMO Running Result

After doing this, download the program <06_USB_MSC_Device> to the START board and run. When the START board connect to the PC, you will find a USB large capacity storage device is in the universal serial bus controller, and there is 1 more disk drives in the equipment manager of PC, as shown below:



Then, after opening the resource manager, you will see more of the 1 disk, as shown in the following diagram:



At this point, the write/read/formatting operation can be performed as the other mobile devices.

5.7. USB_MSC_Host

5.7.1. DEMO purpose

This demo includes the following functions of GD32 MCU:

- Learn to use the USBFS/USBHS as a MSC host
- Learn the operation between the MSC host and the Udisk

GD32F527V-START board integrates USBFS module and the USBHS module, and the modules can be used as USB device, USB host or OTG device. This demo mainly shows how to use the USBFS and USBHS as a USB MSC host to communicate with external Udisk.

5.7.2. DEMO running result

Insert the OTG cable to USB port. Then, download the program <07_USB_MSC_Host> to the START board and run.

If an Udisk has been attached, the user will see the information of Udisk enumeration on the serial assistant. Pressing the Wakeup key, the user will see the root content of the Udisk, then the MSC Host write file to the Udisk, and the user will see information that the MSC host demo is end.

```
++++USB host library started++++

> Reset the USB device.
> Full speed device detected.

> Device Attached.
VID: 30DEh
PID: 6544h
> Mass storage device connected.
Manufacturer: KIOXIA
Product: TransMemory
Serial Number: 0022CFF6BD8EC761D322083E

> Enumeration completed.
_____

> To see the disk information and write file to the Udisk:
> Press Wakeup Key...

> File System initialized.
> Disk capacity: 31001148928 Bytes.
> Exploring disk flash ...
    |__System Volume Information
    |__GD32.TXT
    |__LED.BIN
    |__TEST.TXT

> Writing File to disk flash ...
> GD32.TXT be opened for write.
> File content compare: SUCCESS.
> The MSC host demo is end.
```

6. Revision history

Table 6-1 Revision history

Revision No.	Description	Date
1.0	Initial Release	Mar.18, 2024
1.1	Update chapter 3	Jan.25, 2025
1.2	Update chapter 3	Aug.08, 2025

Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the "Product"), is owned by the Company according to the laws of the People's Republic of China and other applicable laws. The Company reserves all rights under such laws and no Intellectual Property Rights are transferred (either wholly or partially) or licensed by the Company (either expressly or impliedly) herein. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

To the maximum extent permitted by applicable law, the Company makes no representations or warranties of any kind, express or implied, with regard to the merchantability and the fitness for a particular purpose of the Product, nor does the Company assume any liability arising out of the application or use of any Product. Any information provided in this document is provided only for reference purposes. It is the sole responsibility of the user of this document to determine whether the Product is suitable and fit for its applications and products planned, and properly design, program, and test the functionality and safety of its applications and products planned using the Product. The Product is designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only, and the Product is not designed or intended for use in (i) safety critical applications such as weapons systems, nuclear facilities, atomic energy controller, combustion controller, aeronautic or aerospace applications, traffic signal instruments, pollution control or hazardous substance management; (ii) life-support systems, other medical equipment or systems (including life support equipment and surgical implants); (iii) automotive applications or environments, including but not limited to applications for active and passive safety of automobiles (regardless of front market or aftermarket), for example, EPS, braking, ADAS (camera/fusion), EMS, TCU, BMS, BSG, TPMS, Airbag, Suspension, DMS, ICMS, Domain, ESC, DCDC, e-clutch, advanced-lighting, etc.. Automobile herein means a vehicle propelled by a self-contained motor, engine or the like, such as, without limitation, cars, trucks, motorcycles, electric cars, and other transportation devices; and/or (iv) other uses where the failure of the device or the Product can reasonably be expected to result in personal injury, death, or severe property or environmental damage (collectively "Unintended Uses"). Customers shall take any and all actions to ensure the Product meets the applicable laws and regulations. The Company is not liable for, in whole or in part, and customers shall hereby release the Company as well as its suppliers and/or distributors from, any claim, damage, or other liability arising from or related to all Unintended Uses of the Product. Customers shall indemnify and hold the Company, and its officers, employees, subsidiaries, affiliates as well as its suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Product.

Information in this document is provided solely in connection with the Product. The Company reserves the right to make changes, corrections, modifications or improvements to this document and the Product described herein at any time without notice. The Company shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. Information in this document supersedes and replaces information previously supplied in any prior versions of this document.