

**GigaDevice Semiconductor Inc.**

**GD32H759I-START**

**User Guide**

**Rev1.3**

# Tables of Contents

<b>TABLES OF CONTENTS .....</b>	<b>1</b>
<b>LIST OF FIGURES .....</b>	<b>3</b>
<b>LIST OF TABLES .....</b>	<b>4</b>
<b>1. SUMMARY .....</b>	<b>5</b>
<b>2. FUNCTION PIN ASSIGN .....</b>	<b>5</b>
<b>3. GETTING STARTED .....</b>	<b>5</b>
<b>4. HARDWARE LAYOUT OVERVIEW .....</b>	<b>6</b>
4.1. Power supply .....	6
4.2. Boot option .....	6
4.3. LED .....	7
4.4. KEY .....	7
4.5. USART .....	8
4.6. USB .....	8
4.7. Extension .....	9
4.8. GD-Link .....	9
4.9. MCU .....	10
<b>5. ROUTINE USE GUIDE .....</b>	<b>11</b>
5.1. GPIO_Running_LED .....	11
5.1.1. DEMO purpose .....	11
5.1.2. DEMO running result .....	11
5.2. GPIO_Key_Polling_mode .....	11
5.2.1. DEMO purpose .....	11
5.2.2. DEMO running result .....	11
5.3. EXTI_Key_Interrupt_mode .....	12
5.3.1. DEMO purpose .....	12
5.3.2. DEMO running result .....	12
5.4. USART_HyperTerminal_Interrupt .....	12
5.4.1. DEMO purpose .....	12
5.4.2. DEMO running result .....	12
5.5. TIMER_Key_EXTI .....	13
5.5.1. DEMO purpose .....	13
5.5.2. DEMO running result .....	13
5.6. USB_MSC_Device .....	13
5.6.1. DEMO purpose .....	13
5.6.2. DEMO running result .....	14
5.7. USB_MSC_Host .....	14
5.7.1. DEMO purpose .....	14

---

5.7.2.	DEMO running result .....	15
<b>6.</b>	<b>REVISION HISTORY .....</b>	<b>16</b>

## List of Figures

Figure 4-1. Schematic diagram of power supply.....	6
Figure 4-2. Schematic diagram of boot option .....	6
Figure 4-3. Schematic diagram of LED function .....	7
Figure 4-4. Schematic diagram of Key function .....	7
Figure 4-5. Schematic diagram of USART .....	8
Figure 4-6. Schematic diagram of USB .....	8
Figure 4-7. Schematic diagram of Extension.....	9
Figure 4-8. Schematic diagram of GD-Link.....	9
Figure 4-9. Schematic diagram of MCU .....	10

# List of Tables

Table 2-1. Function pin assignment.....	5
Table 6-1. Revision history .....	16

## 1. Summary

GD32H759I-START uses GD32H759IMT6 as the main controller. It uses GD-Link Mini USB interface to supply 5V power. Reset, Boot, K2-User KEY, LED, USB and USART to USB interface are also included. For more details please refer to GD32H759I-START-V1.4 schematic.

## 2. Function Pin Assign

**Table 2-1. Function pin assignment**

Function	Pin	Description
LED	PC9	LED1
	PC10	LED2
	PC11	LED3
	PC12	LED4
RESET		K1-Reset
KEY	PD15	K2-User
USART	USART0_TX	PF4
	USART0_RX	PF5
USB	PA9	USBHS0_VBUS
	PA10	USBHS0_ID
	USBHS0_DM	USBHS0_DM
	USBHS0_DP	USBHS0_DP
	PJ10	USBHS1_ULPI_NXT
	PJ9	USBHS1_ULPI_DIR
	PJ8	USBHS1_ULPI_STP
	PJ11	USBHS1_ULPI_CK
	PG5	USBHS1_ULPI_D7
	PG4	USBHS1_ULPI_D6
	PG3	USBHS1_ULPI_D5
	PG2	USBHS1_ULPI_D4
	PG1	USBHS1_ULPI_D3
	PK2	USBHS1_ULPI_D2
	PK1	USBHS1_ULPI_D1
	PK0	USBHS1_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 J-Link tool or GD-Link tool 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 the power supply is OK.

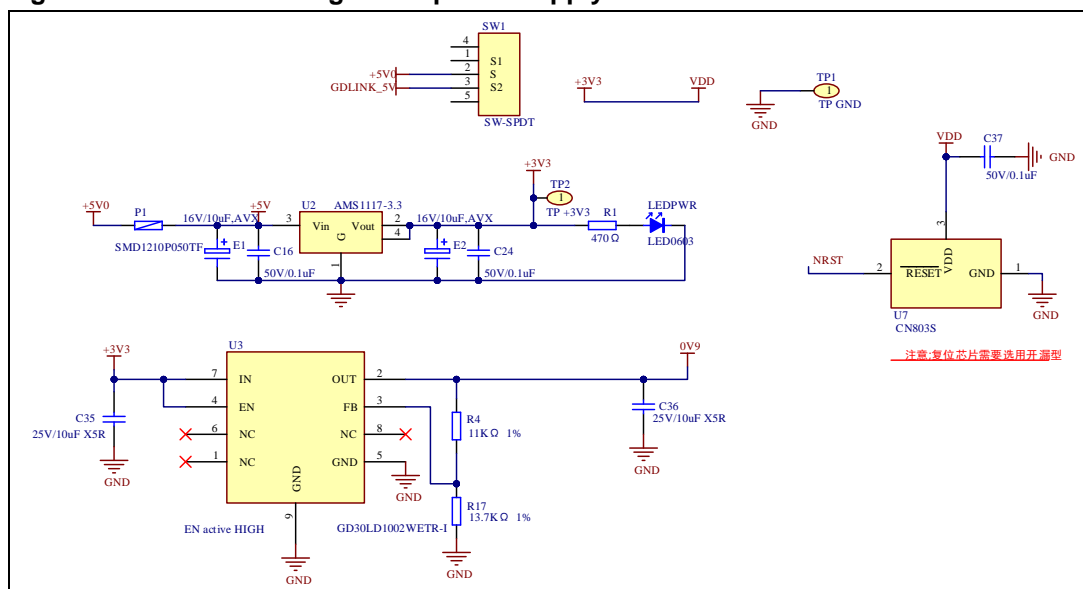
There are Keil version and IAR version of all projects. Keil version of the projects are created based on Keil MDK-ARM 5.29 uVision5. IAR version of the projects are created based on IAR Embedded Workbench for ARM 8.32.1. During use, the following points should be noted:

1. If you use Keil uVision5 to open the project. In order to solve the "Device Missing (s)" problem, you can install GigaDevice.GD32H7xx\_DFP.1.2.0.pack.
2. If you use IAR to open the project, install IAR\_GD32H7xx\_ADDON\_1.2.0.exe to load the associated files.

## 4. Hardware layout overview

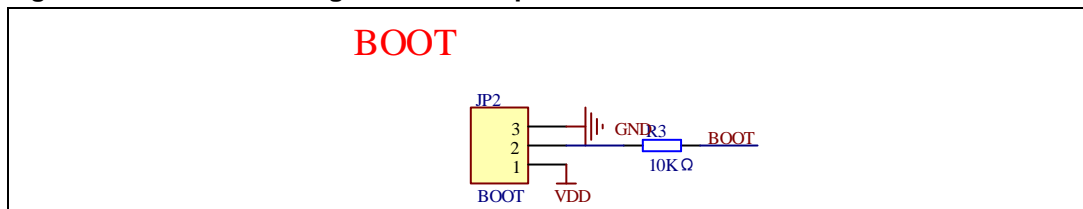
#### 4.1. Power supply

**Figure 4-1. Schematic diagram of power supply**



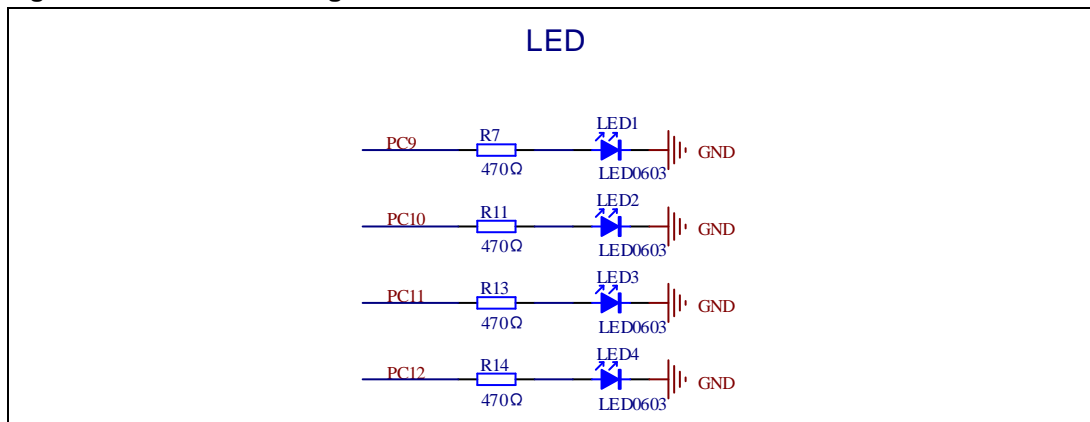
## 4.2. Boot option

**Figure 4-2. Schematic diagram of boot option**



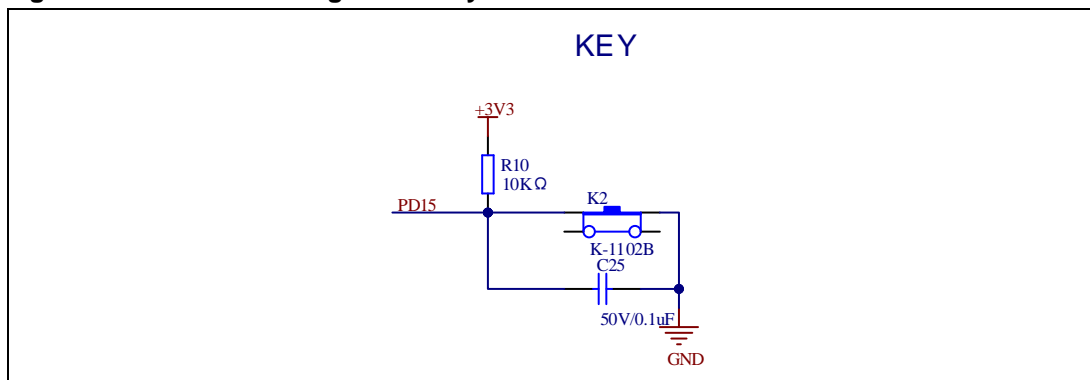
### 4.3. LED

Figure 4-3. Schematic diagram of LED function



### 4.4. KEY

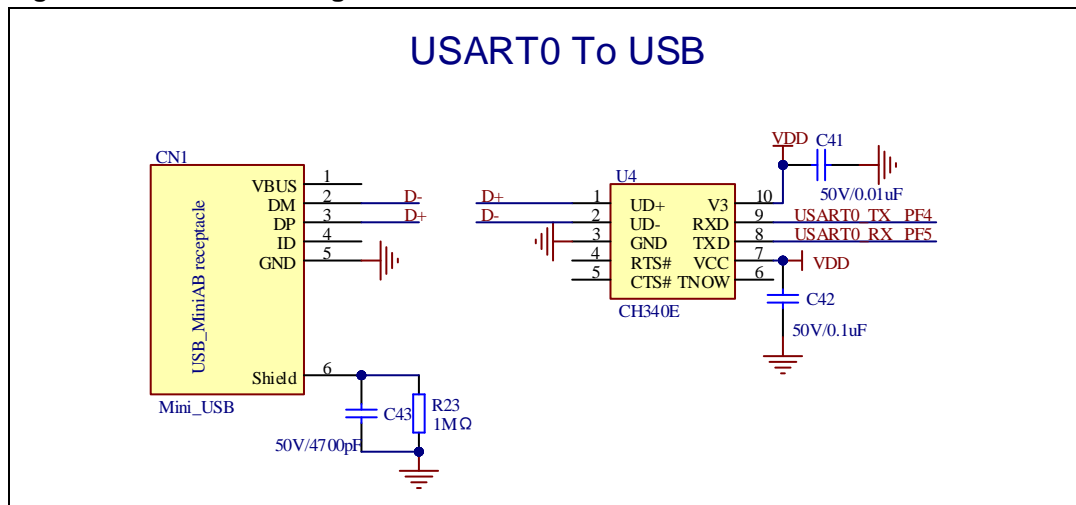
Figure 4-4. Schematic diagram of Key function





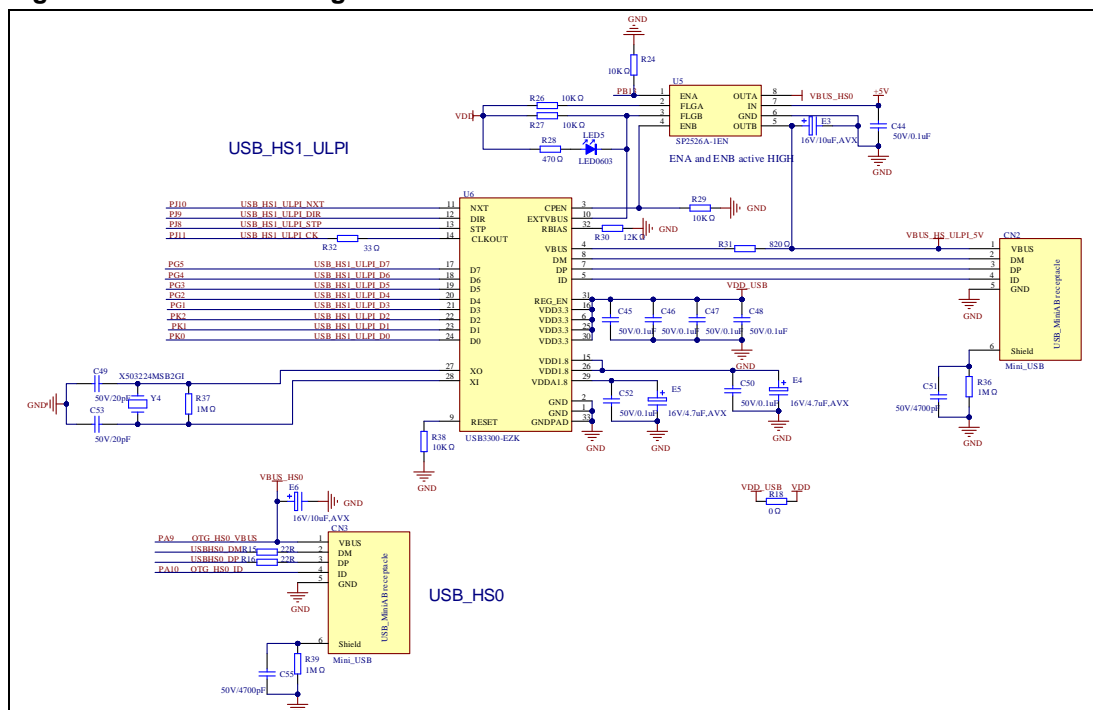
## 4.5. USART

### Figure 4-5. Schematic diagram of USART



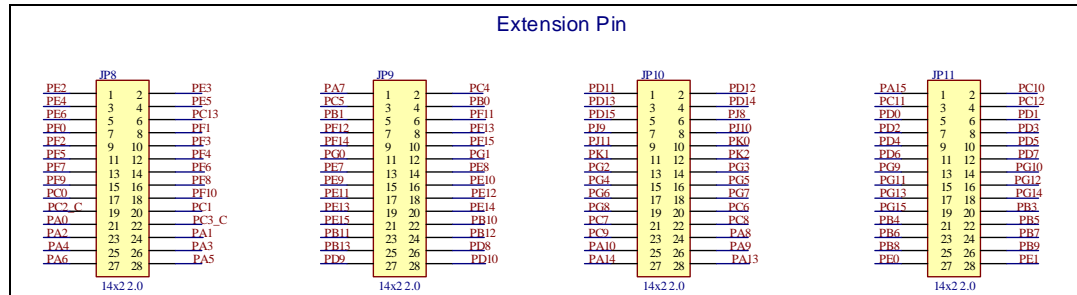
## 4.6. USB

**Figure 4-6. Schematic diagram of USB**



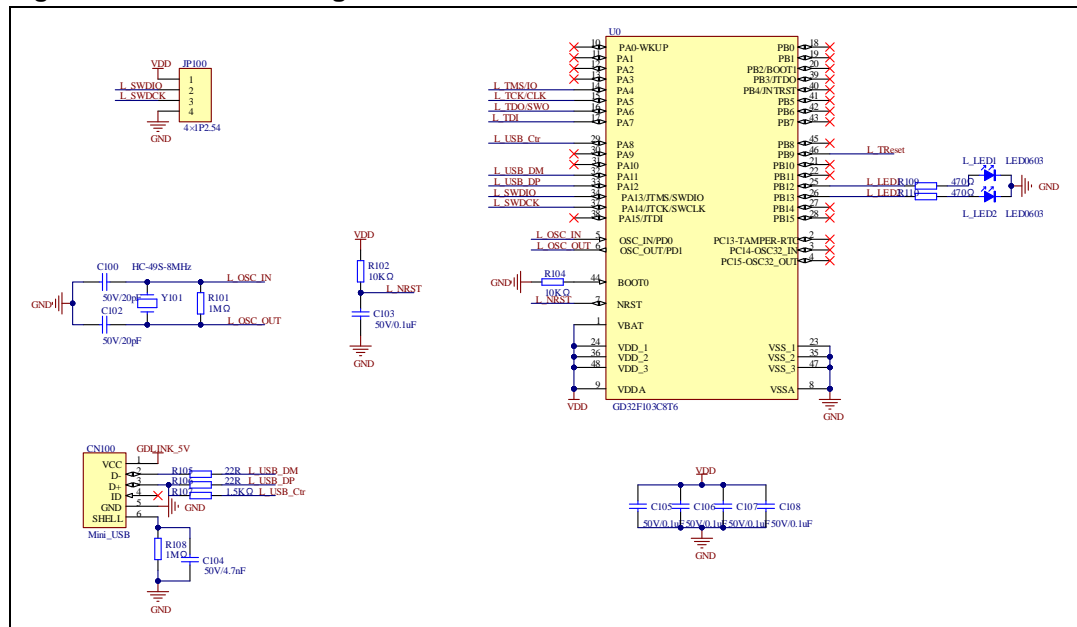
## 4.7. Extension

Figure 4-7. Schematic diagram of Extension



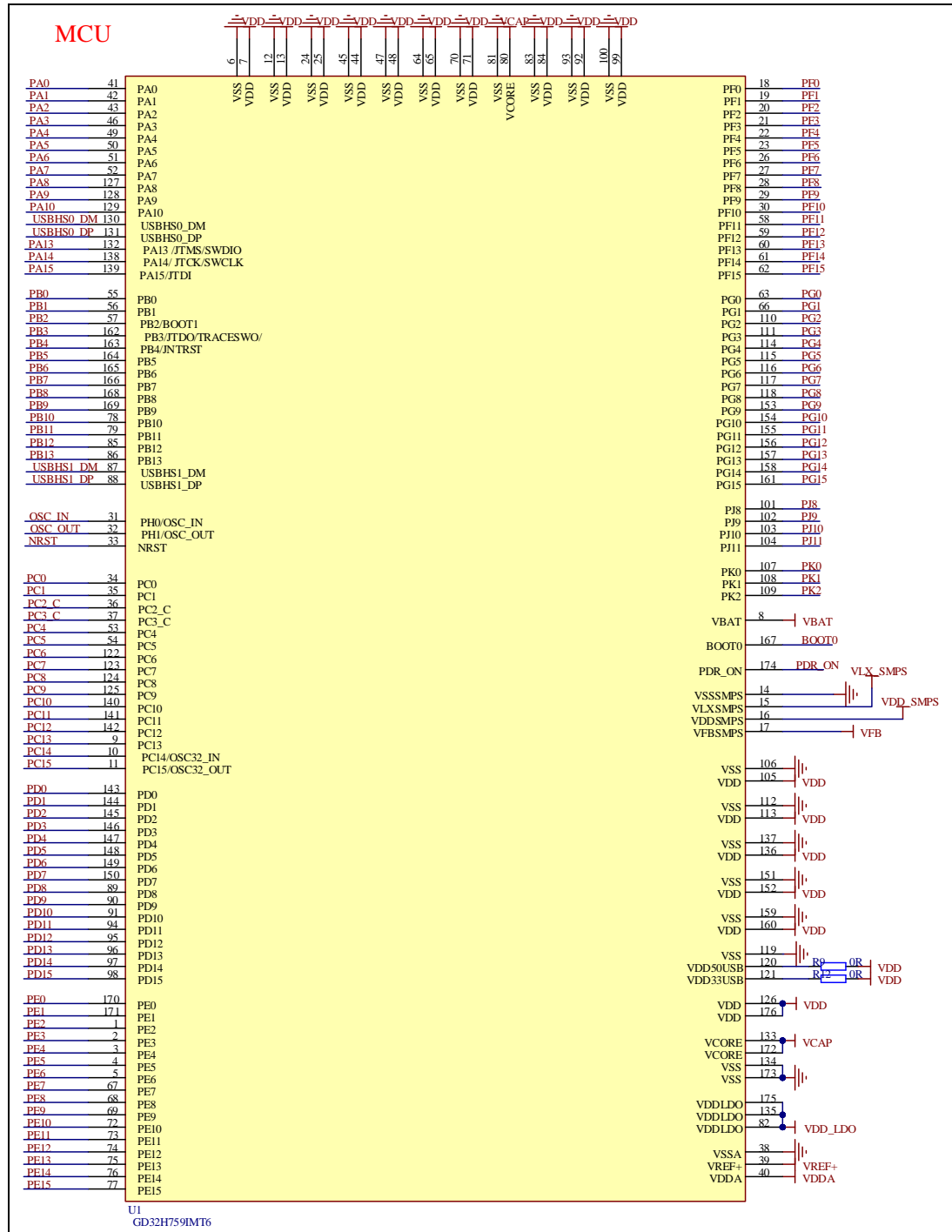
## 4.8. GD-Link

Figure 4-8. Schematic diagram of GD-Link



## 4.9. MCU

Figure 4-9. Schematic diagram of MCU



## 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.

GD32H759I-START-V1.4 board has four LEDs. The LEDs 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, four LEDs will change the state like running water and then repeat the whole process over and over again.

### 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.

GD32H759I-START-V1.4 board has one key and four LEDs. The key is User key. The LEDs are controlled by GPIO.

This demo will show how to use the User key to control the LED2. When press down the User Key, it will check the input value of the IO port. If the value is 0 and will wait for 100ms. Check the input value of the IO port again. If the value still is 0, it indicates that the button is pressed successfully and toggle LED2.

#### 5.2.2. DEMO running result

Download the program <02\_GPIO\_Key\_Polling\_mode> to the START board. Press down the User Key, LED2 will be turned on. Press down the User Key again, LED2 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

GD32H759I-START board has one key and four LEDs. The key is User 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 User 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. After startup, the LED2 flash once, press down the User key, LED2 will be turned on, press down the User 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 HyperTerminal.

### 5.4.2. DEMO running result

Download the program <04\_USART\_HyperTerminal\_Interrupt> to the START board and connect serial cable to USART. Firstly, all the LEDs are turned on and off for test. Then, the USART sends the tx\_buffer array (from 0x00 to 0xFF) to the hyperterminal and waits for receiving data from the hyperterminal that you must send. The string that you have sent is stored in the rx\_buffer array. The receive buffer have a BUFFER\_SIZE bytes as maximum. 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, LED4 toggle together.

The output information via the HyperTerminal 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 GPIO control the LED and the KEY
- Learn to use EXTI to generate external interrupt
- Learn to use TIMER to generate PWM

GD32H759I-START board has one key and four LEDs. The key is User key. The LED1, LED2, LED3 and LED4 are controlled by GPIO.

This demo will show how to use the TIMER PWM to trigger EXTI interrupt to toggle the state of LED2 and EXTI interrupt line to control the LED1. When press down the User 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, all the LED1 and LED2 are flashed once for test, press down the User Key, LED1 will be turned on. Press down the User Key again, LED1 will be turned off. Connect PA5 (TIMER1\_CH0) and PA4 with DuPont line. The LED2 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 USBHS
- 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

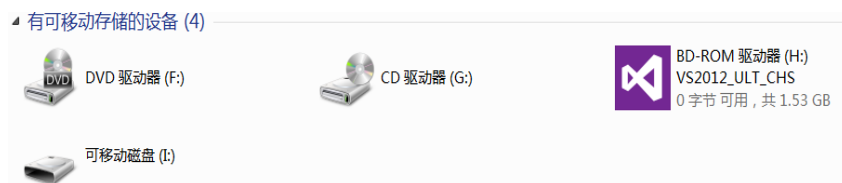
and mobile devices, which allow a universal serial bus (USB) device to access a host computing device, file transfer between them, mainly including mobile hard disk, mobile U disk drive, etc. The MSC device must have a storage medium, and this demo uses the MCU's internal SRAM as the storage medium. 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

Download the program < 06\_USB\_MSC\_Device > to the START board and run. When the start board is connected 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.

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

GD32H759I-START-V1.4 board integrates USBHS0 module and the USBHS1 module, and the modules can be used as USB device, USB host or OTG device. This demo mainly shows how to use the USBHS0 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 User 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.
> High speed device detected.
> Device Attached.
VID: FFFFh
PID: 5678h
> Mass storage device connected.
Manufacturer: USB
Product: Disk 2.0
Serial Number: 9207302211445624486
> Enumeration completed.
>To see the disk information:
>Press User Key...
> File System initialized.
> Disk capacity: 4026531328d Bytes.
> Exploring disk flash ...
|__System Volume Information
|__GD32.TXT
|__RECYCLER
|__PORT.JPG
> Writing File to disk flash ...
> GD32.TXT created in the disk.
> The MSC host demo is end.
```



## 6. Revision history

Table 6-1. Revision history

Revision No.	Description	Date
1.0	Initial Release	Mar.31, 2023
1.1	Modify description in chapter <b><u>Summary</u></b> and <b><u>Table 2-1.</u></b> <b><u>Function pin assignment.</u></b>	Dec.05, 2023
1.2	Update schematic.	Jan.12, 2024
1.3	Update schematic.	Feb.17, 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 under the intellectual property laws and treaties of the People's Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. 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.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. 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. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do 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 Products. Customers shall indemnify and hold the Company 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 Products.

Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and Products and services described herein at any time, without notice.