User Manual

APM32F407 EVAL Board

Version: V1.0

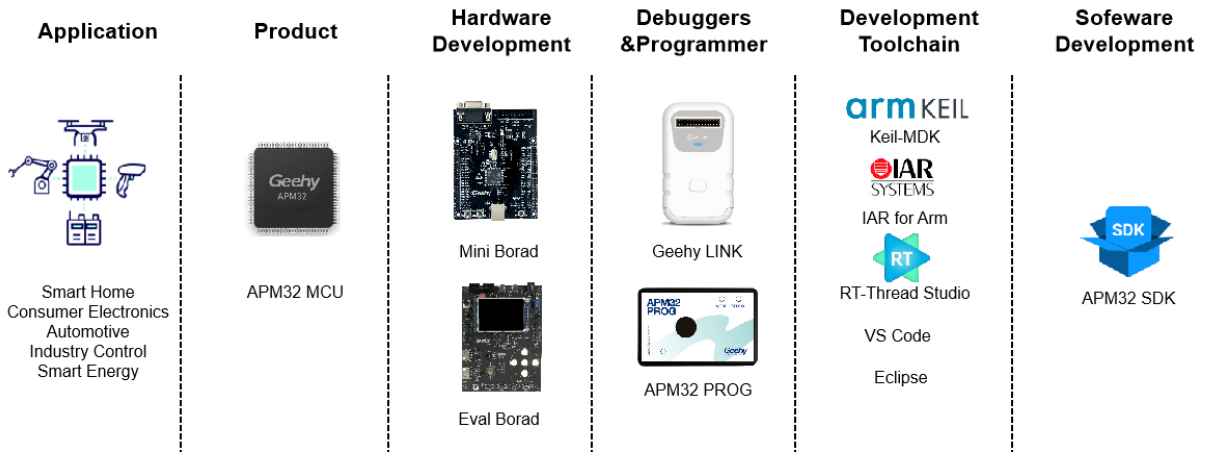
1 Introduction

This User Manual describes the functions, on-board resources and supporting SDK of APM32F407IG EVAL Board. The SDK and related data mentioned in the document can be obtained from the official website of Geehy (www.geehy. com).

1.1 APM32 Ecosystem

The APM32 ecosystem includes product application solution, hardware development board, download simulation tool, development tool chain and SDK. Moreover, the development tool chain is suitable for many development tools at home and abroad, such as Keil-MDK, IAR for Arm, Eclipse, etc., and all of them are equipped with relevant enginering in the SDK to meet the needs of different users in different platforms.

Figure 1 APM32 Ecosystem



1.2 Evaluation board

APM32F407 EVAL Board is a complete demonstration and development platform for high-performance APM32F4xx series MCU, including a base board and two core boards. The two core boards use APM32F407ZGT6 (LQFP144) and APM32F407IG T6 (LQFP176) as the core master, and the two MCU chips are based on Arm® Cortex®-M4 core, working frequency 168MHz. It is equipped with the corresponding SDK, which can help developers evaluate chip performance of APM32F4xx series MCU or related development applications.

Figure 2 APM32F407 EVAL Base Board

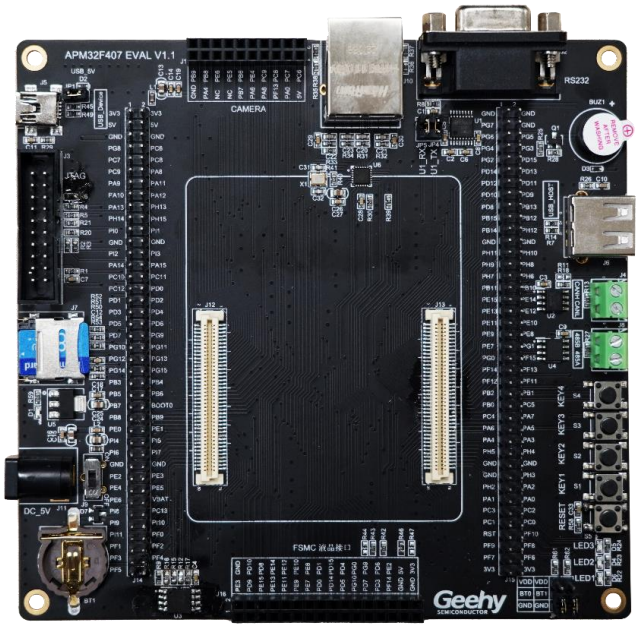
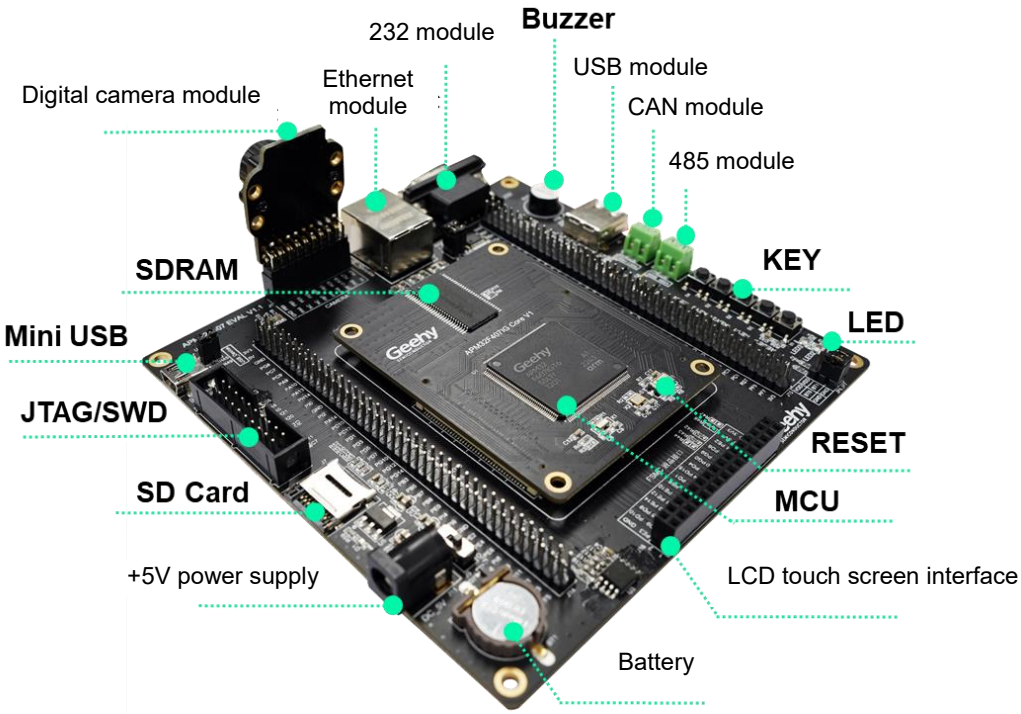


Figure 3 APM32F407ZG Core V1 Board

Figure 4 APM32F407IG Core V1 board

Figure 5 APM32F407 EVAL Board (taking APM32F407IG CoreV1 board as an example)



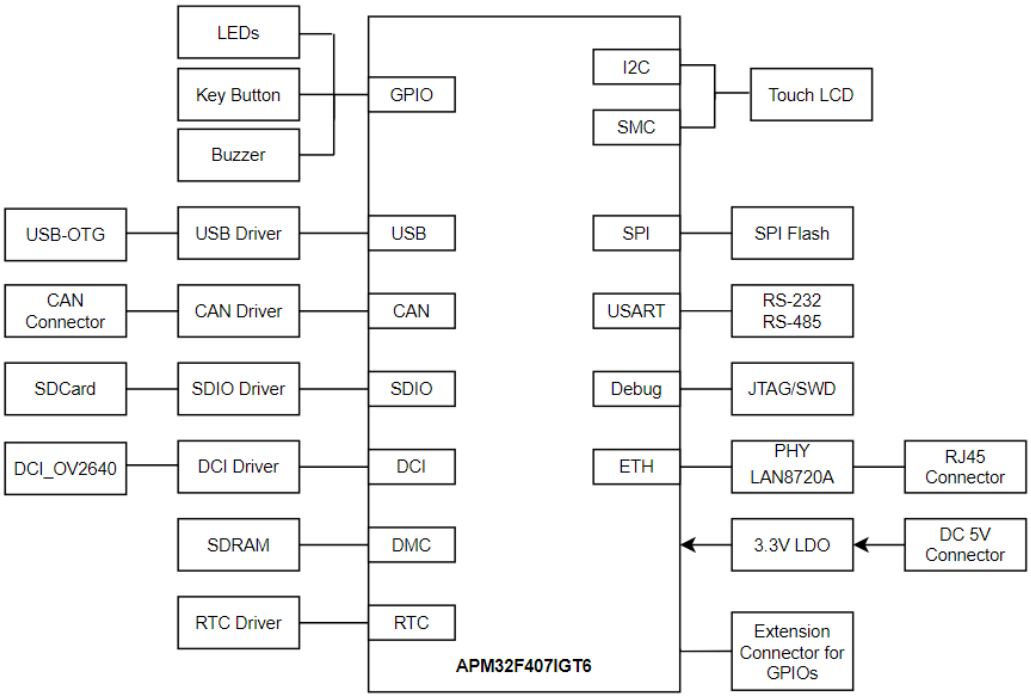
Contents

2 Function overview

The APM32F407 EVAL Board mainly includes the following functional peripherals:

1. LDE：4
2. KEY：5
3. Buzzer: 1
4. JATA/SWD：1
5. Ethernet：1
6. USB-OTG：HS/FS
7. Support CAN-2.0/RS-232/RS-485
8. Support RTC with backup battery
9. Support MicroSD™ memory card
10. Support OV2640 camera module and expansion connector
11. Support 4.3" 800x480 TFT color LCD touch screen

Figure 6 APM32F407 EVAL Board Overview

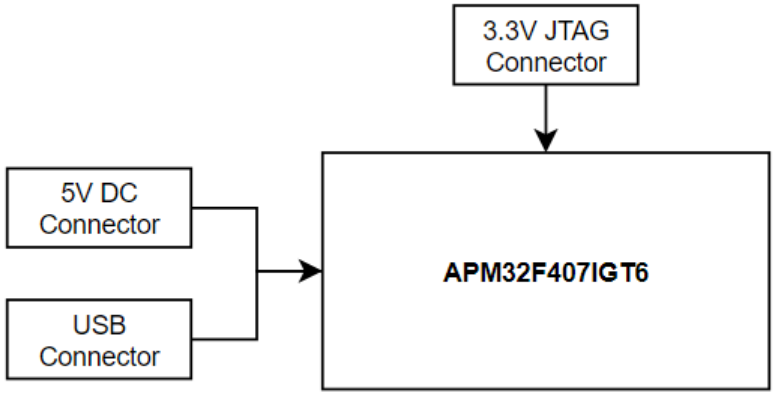


2.1 Power supply

The APM32F407 EVAL Board can be powered by an external 5V DC power supply, or by selecting USB-to-serial port Mini USB interface through the jumper cap. Other required voltages are provided by the on-board voltage regulator. In addition, the JTAG interface can also supply power to the evaluation board, but it can only supply 3.3V voltage to the evaluation board.

Besides, rich 5V/3.3V/GND pin headers are reserved on the board for users to conduct extended experiment.

Figure 7 Power Supply of APM32F407 EVAL Board



2.2 Battery interface

On-board CR1220 standard battery interface, used as backup power supply of RTC.

2.3 Clock

APM32F407 EVAL Board carries two external clocks:

- X2, 32.768KHz clock is used for RTC or other required peripherals.
- X2, 8MHz clock is used for APM32F407xx MCU. If the internal RC clock of the chip is used, the clock can be removed or the HSE can be disabled

2.4 Reset control

Provide two reset controls, which can be generated by hardware or software:

- RESET key; press it to send a reset signal.
- JTAG reset signal.

2.5 Simulation and download interface

- Standard 20-pin IDC JTAG connecting interface.
- Four-wire SWD connecting interface

2.6 LCD screen

It supports 4.3-inch TFT color LCD touch screen with 800 x 480 pixels, and drives the LCD screen through the SMC and I2C interfaces of MCU.

2.7 LED lights

APM32F407 EVAL Board carries 3 user LED lights, which can be used for display by users or for indication purpose during the experiment.

2.8 Keys

APM32F407 EVAL Board carries 3 user keys, which can be used for LCD menu switching or other input purposes.

2.9 SPI Flash

APM32F407 EVAL Board carries a SPI Flash chip W25Q16, provides 16M Bit of external storage space, and is connected to this chip through the SPI1 interface of MCU.

2.10 RS485 interface

APM32F407 EVAL Board is connected to the RS485 chip through the USART3 peripheral and provides a RS485 interface. It should be noted that the RS485 function requires 5V power supply to work normally.

2.11 CAN interface

APM32F407 EVAL Board has an on-board CAN interface, which can be used by users for CAN communication. It should be noted that the CAN function module requires 5V power supply to work normally.

2.12 Ethernet interface

APM32F407 MCU is equipped with Ethernet peripheral module, which can communicate with external physical layer chip through RMII interface. The physical layer chip on the APM32F407 EVAL Board is LAN8720A.

2.13 DCI interface

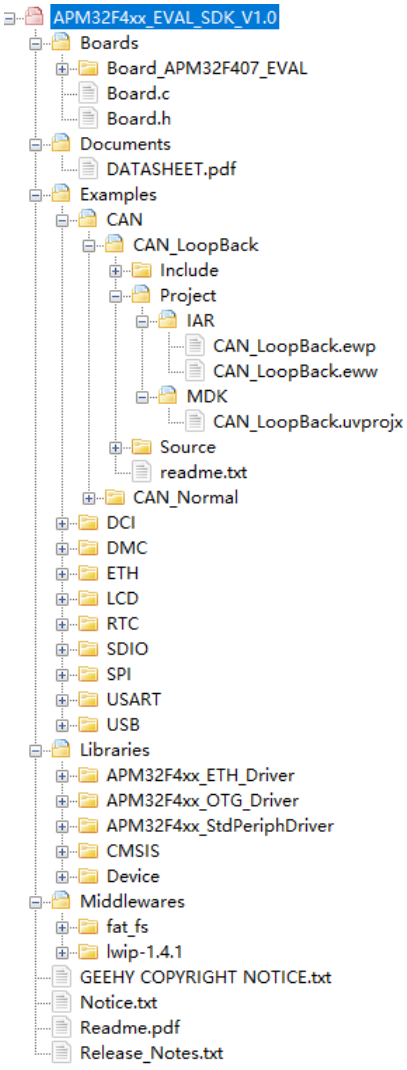
APM32F407 EVAL Board on-board DCI interface can transmit data of OV2640 camera through DMA and display it on LCD. It supports processing data in the formats of JPEG and RGB565.

3 SDK Overview

The SDK[1] is provided in the form of compressed packages, including on-board driver packages, such as basic LED, Button, Buzzer and COM driver, DCI OV2640 driver package, LCD screen driver package, SPI W25Q16 Flash driver package, SDIO SD Card driver package and so on.

It also includes several necessary libraries, such as APM32F4xx standard library, Ethernet peripheral driver library and USB OTG peripheral driver library. It involves many routines that are easy to reuse, such as RS485 communication, CAN communication and Ethernet communication.

Figure 8 SDK Directory Architecture

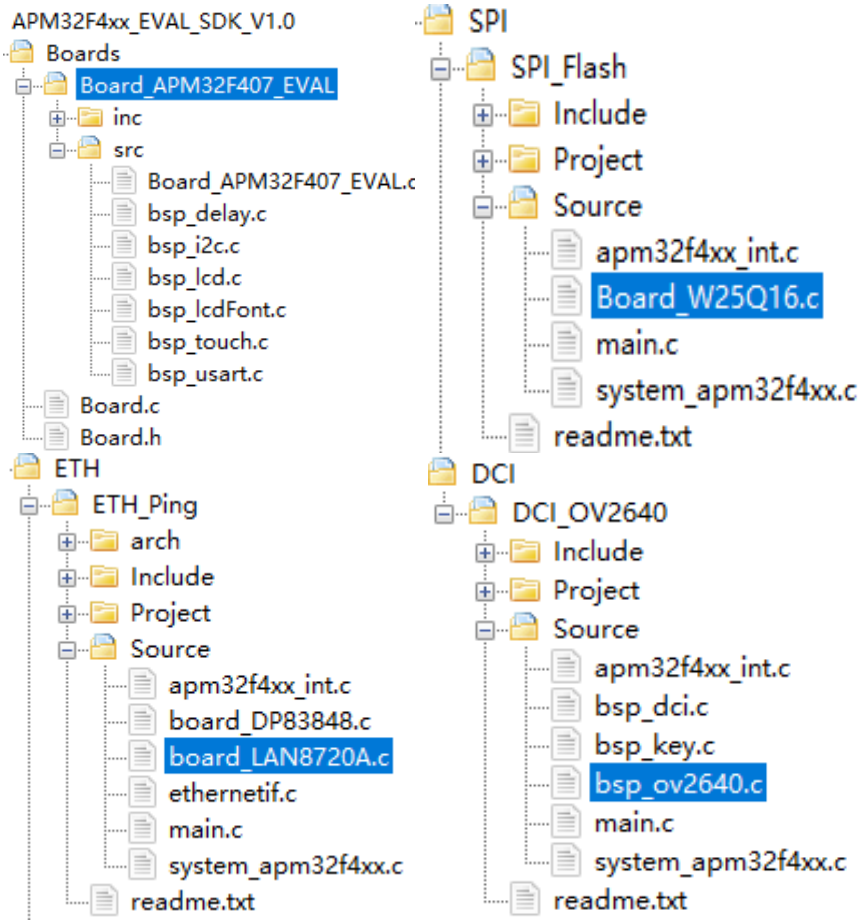


[1]. There is no package supported by IDE in the SDK of APM324xx_EVAL Board. You need to obtain it on the official website of Geehy www.geehy.com.

3.1 On-board driver

The on-board drivers include basic LED, Button, Buzzer and COM drivers, DCI OV2640 driver package, LCD screen driver package, and W25Q16 Flash driver package.

Figure 9 On-board Driver



3.2 Library file

The library file contains APM32F4xx standard peripheral driver library, Ethernet peripheral driver library and USB OTG peripheral driver library.

3.3 Middleware

The middleware used in the SDK package includes fat_fs and lwip-1.4.1.

3.4 IDE support

This SDK routine is equipped with the engineering of two development tools of Keil-MDK and IAR for Arm.

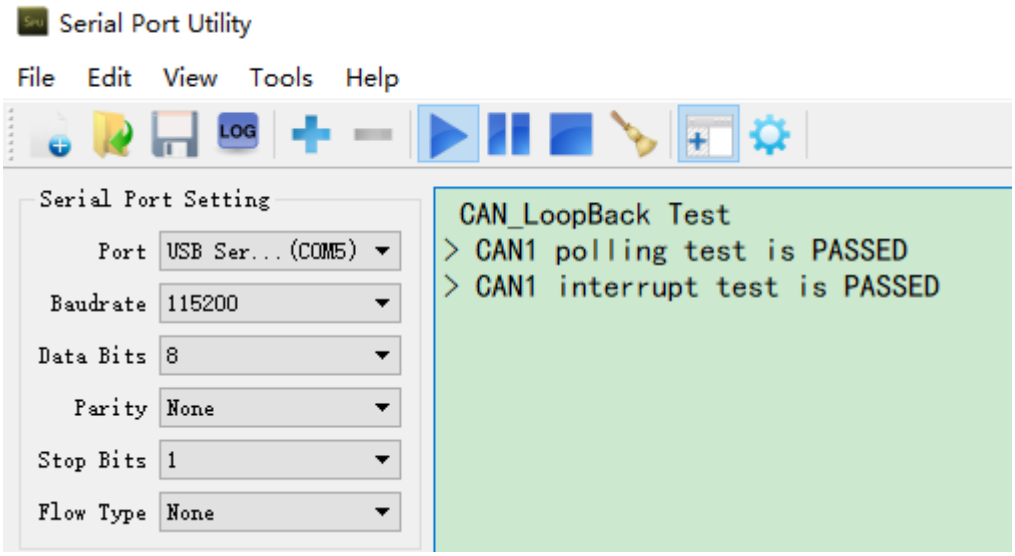
3.5 Routine

The SDK package contains many applications that are easy to reuse, such as DCI OV2640 camera, LCD touch screen, RS485 communication, CAN communication and Ethernet RS485.

3.5.1 CAN LoopBack

The CAN LoopBack routine describes how to configure communication in loopback mode, and compares the received message with the transmitted message. The data transmitted by polling and interrupt will be displayed on the serial port assistant through USART1.

Figure 10CAN LoopBack



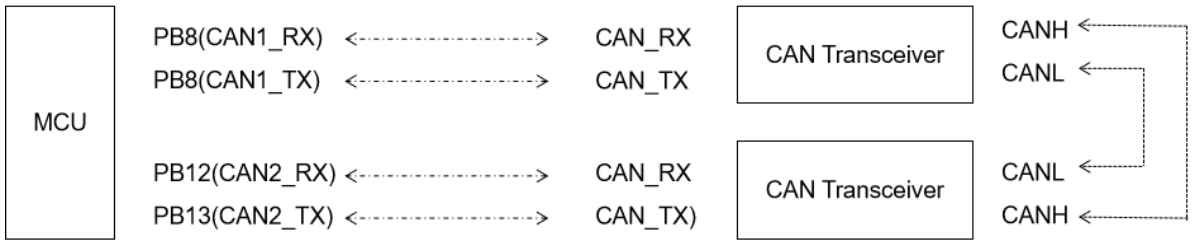
3.5.2 CAN Noemal

The CAN Noemal routine is communication conducted by two-way CAN peripherals of APM32F107 and is used to show how to use the CAN module. During the experiment, the H/L wires of CAN1 and CAN2 need to be short-circuited, and it shall be ensured that 5V power supply is provided to the board. You can select to transmit data by polling or interrupt.

In polling mode: after successfully receiving the data, "CAN polling test passed!" will be displayed on the serial port assistant through USART1 and LED2 will be on.

In interrupt mode: after successfully receiving the data, "CAN interrupt test passed!" will be displayed on the serial port assistant through USART1 and LED3 will be lit.

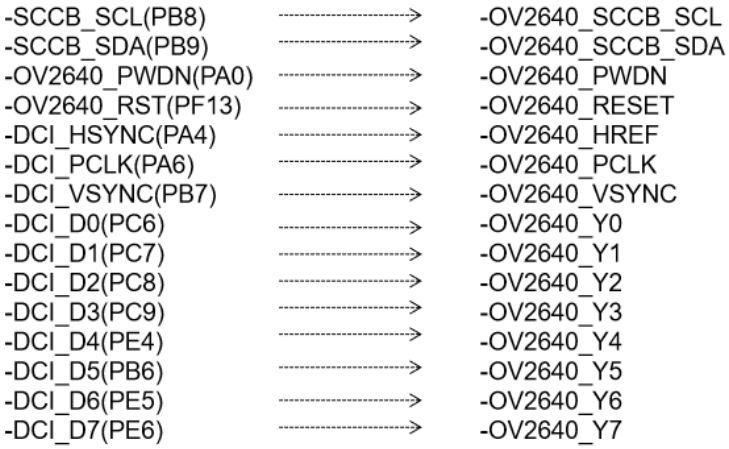
Figure 11 CAN Dual Menu


MCU
PB8(CAN1_RX)
PB8(CAN1_TX)
PB12(CAN2_RX)
PB13(CAN2_TX)
CAN_RX
CAN_TX
CAN_RX
CAN_TX)
CAN Transceiver
CAN Transceiver
CANH
CANL
CANL
CANH

3.5.3 DCI OV2640

DCI OV2640 routine shows how to obtain image data of OV2640 camera by using DCI. DCI continuously obtains the data of OV2640 camera through DMA and sends it to LCD through USART2. When KEY1 is pressed, select RGB565 data format; when KEY2 is pressed, select JPEG data format.

Figure 12DCI OV2640



3.5.4 DMC SDRAM

This routine provides how to use DMC to read and write data to external SDRAM, and prints the information to the serial port assistant through USART2.

Figure 13DMC SDRAM

```
**************************************
* APM32F40x DMC SDRAM Example
* AHB Clock       : 168 MHz
* SDRAM Clock Psc : 4 div
* SDRAM Clock     : 42 MHz
* Row Address Width    :11 bit
* Column Address Width :8 bit
* Bank Address Width   :1 bit band addr
* capacity             :2 MByte
**************************************
    Test 8 bits read and write operation on the 2MB bytes space of external SDRAM
    Check passed
**************************************
    Test 16 bits read and write operation on the 2MB bytes space of external SDRAM
    Check passed
**************************************
    Test 32 bits read and write operation on the 2MB bytes space of external SDRAM
    Check passed
**************************************
```

3.5.5 ETH Ping

This routine describes how to use APM32F4xx_ ETH_ Driver library to use Ethernet functions. After the Ethernet motherboard is configured, the static IP address from USART1 to printf will be used. If the computer pings the static IP address (192.168.73.22), the computer will access the motherboard normally.

Figure 14ETH Ping

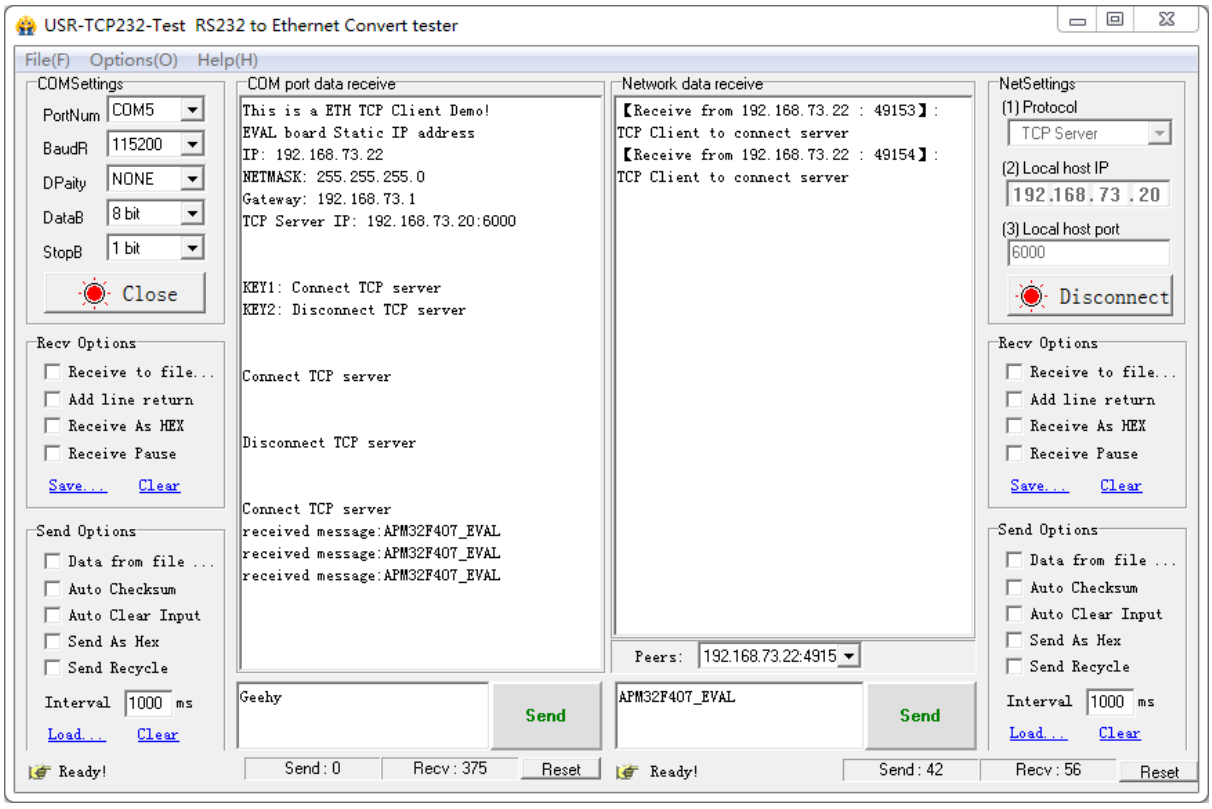
```
This is a Eth Demo !
    Static IP address
IP：192，168，73，22
NETMASK：255，255，255，0
Gateway：192，168，73，1
```

```
C:\Windows\system32>ping 192.168.73.22

Pinging 192.168.73.22 with 32 bytes of data:
Reply from 192.168.73.22: bytes=32 time<1ms TTL=128
Reply from 192.168.73.22: bytes=32 time<1ms TTL=128
Reply from 192.168.73.22: bytes=32 time<1ms TTL=128
Reply from 192.168.73.22: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.73.22:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

3.5.6 ETH TCP Client

This routine describes how to use APM32F4xx_ ETH_ Driver library to use Ethernet function. After initialization, you can see the system information of the serial port assistant through the USART1 or LCD screen. Connect to the server (IP 192.168.73.51:6000). Disconnect the server through KEY2. You can use the TCP auxiliary program to send data to the EVALBoard through the TCP server.

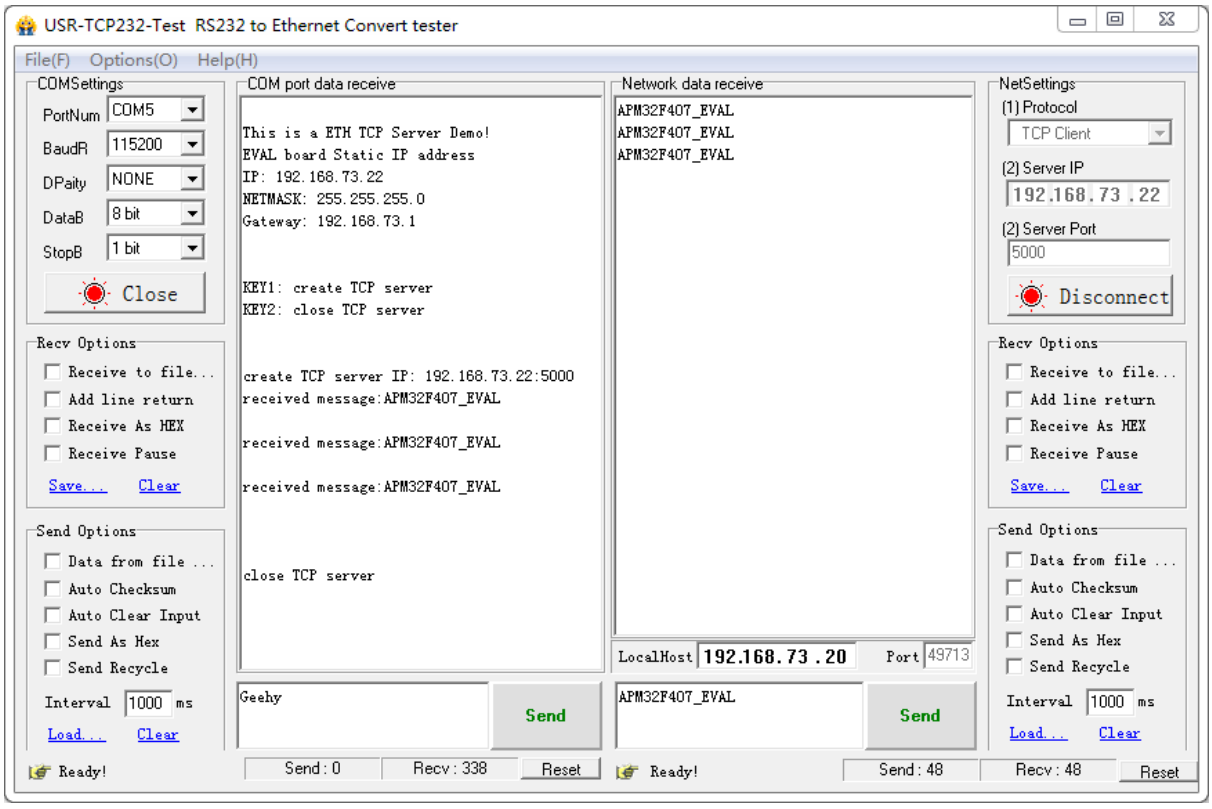
Figure 15ETH TCP Client



3.5.7 ETH TCP Server

This routine describes how to use APM32F4xx_ ETH_ Driver library to use Ethernet functions to create TCP server, receive and send data to TCP client through tcp assistant. After initialization, you can see the system information of the serial port assistant through the USART1 or LCD screen. Create the server (IP 192.168.73.22 - 5000). Shut down the server through KEY2. You can connect data to the EVAL Board through tcp assisatant.

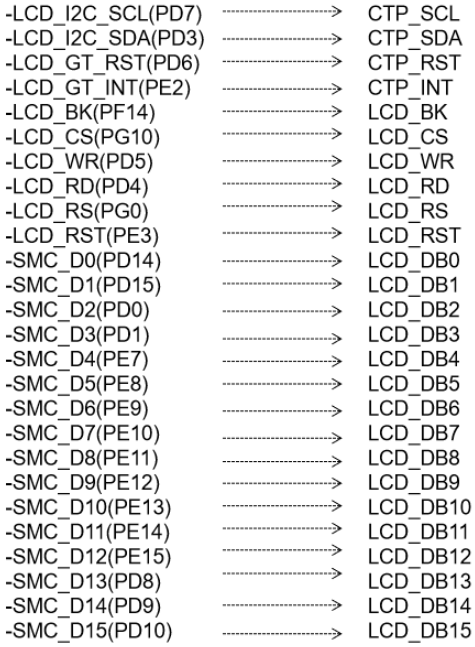
Figure 16ETH TCP Server



3.5.8 LCD ShowFigure

LCD_ShowFigure routine describes how to use the LCD driver to display graphics on the LCD screen. Press KEY1 and the LED will display different states. (LCD screen is 4.3 inches, 800 x 480 pixels).

Figure 17LCD ShowFigure



3.5.9 LCD Touch

LCD_Touch routine describes how to use interrupts to obtain LCD touch screen data. I2C continuously obtains contact data through interrupts and converts it into axis calculation data in the LCD window (4.3 inches and 800 x 480 pixels). You can touch some buttons to control the LED and get feedback on the LCD. Each time complete touch data is obtained, the LED will flip.

Figure 18LCD Touch

3.5.10 RTC Alarm

This routine describes how to use the RTC Alarm function. After initialization, LED2 will light up; count down for five seconds, then wake up the RTC alarm, LED2 will go out, and LED3 will flash. It can be monitored through USART1.

3.5.11 RTC Calendar

This routine describes how to use the RTC calendar function. After initialization, LED2 will light up and LED1 will flash every second. It can be monitored through USART1 and LCD.

Figure 19LCD Calendar

```
 RTC initialization completed

This is a RTC Calendar Demo!
Date : 2022/05/20 weekday:5
Time : 02:02:02

This is a RTC Calendar Demo!
Date : 2022/05/20 weekday:5
Time : 02:02:04

This is a RTC Calendar Demo!
Date : 2022/05/20 weekday:5
Time : 02:02:05

This is a RTC Calendar Demo!
Date : 2022/05/20 weekday:5
Time : 02:02:06
```

3.5.12 SDIO SD Card

This program shows how to use the SDIO module to access the SD card data through DMA and verify it after transmission. The data of the SD card can be displayed using the serial port assistant. After the power supply, single-block test or multiple-block test can be selected by KEY1 and KEY2.

Figure 20SDIO SDCard

```
SD Init Success
Card Type:SDHC V2.0
Card ManufacturerID:254
Card s_rca:1
Card Capacity:7600 MB
Card BlockSize:512

Single Block Test
write success
Sector 0 Data:
1 2 3 4 5 6 7 8 9 A B C D E F 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 0
1 2 3 4 5 6 7 8 9 A B C D E F 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 0
Sector 0 Data Ended
Single Block Test OK
```

3.5.13 SPI Flash

This routine uses SPI interface to access the external flash chip W25Q16, with a size of 16M Bit. The test process is to press KEY1, write and read data to Flash and compare them. If the compared read and written data are equal, LED1 will light up and it can be monitored through USART1.

Figure 21SPI Flash

```
 This is a 16M flash(W25Q16) example.

FlashID is 0xEF4015,  Manufacturer Device ID is 0x14

SPI FLASH W25Q16 detected!

Push KEY1 to Read and Write FLASH W25Q16 by SPI.

SPI write Data to flash

Write data:
0x1011  0x1012  0x1013  0x1014  0x1015  0x1016  0x1017  0x1011
0x1012  0x1013  0x1014  0x1015  0x1016  0x1017  0x1011  0x1012
0x1013  0x1014  0x1015  0x1016  0x1017  0x1011  0x1012  0x1013
0x1014  0x1015  0x1016  0x1017  0x1011  0x1012  0x1013  0x1014
0x1015  0x1016  0x1017  0x1011  0x1012  0x1013  0x1014  0x1015
0x1016  0x1017  0x1011  0x1012  0x1013  0x1014  0x1015  0x1016
0x1017  0x1011  0x1012  0x1013  0x1014  0x1015  0x1016  0x1017

Read data:
0x1011  0x1012  0x1013  0x1014  0x1015  0x1016  0x1017  0x1011
0x1012  0x1013  0x1014  0x1015  0x1016  0x1017  0x1011  0x1012
0x1013  0x1014  0x1015  0x1016  0x1017  0x1011  0x1012  0x1013
0x1014  0x1015  0x1016  0x1017  0x1011  0x1012  0x1013  0x1014
0x1015  0x1016  0x1017  0x1011  0x1012  0x1013  0x1014  0x1015
0x1016  0x1017  0x1011  0x1012  0x1013  0x1014  0x1015  0x1016
0x1017  0x1011  0x1012  0x1013  0x1014  0x1015  0x1016  0x1017

16M flash(W25Q16) test OK! LED1 on
```

3.5.14 SPI FullDuplex

This example shows how to transmit data through SPI peripherals. Press KEY1 to send data from SPI1 to SPI2. If the communication is successful, LED2 will light up and the data will be displayed on the serial port assistant. Press KEY2 to realize full-duplex transmission from SPI1 to SPI2. If the transmission is successful, LED3 will light up and the data will be displayed on the serial port assistant.

Figure 22SPI FullDuplex

```
Connect SPI1 and SPI2 pins as follows:
SPI1 NSS    (PA4)   to   SPI2  NSS    (PB12)
SPI1 SCK    (PA5)   to   SPI2  SCK    (PB13)
SPI1 MISO   (PA6)   to   SPI2  MISO   (PB14)
SPI1 MOSI   (PA7)   to   SPI2  MOSI   (PB15)
```

3.5.15 USART Polling

This program is designed to show how to send or receive data by polling. In this case, USART1 and USART2 send or receive data from each other. Verify after transmission. If the data transmission from USART1 to USART2 is successful, LED2 will light up. If the data transmission from USART2 to USART1 is successful, ED3 will light up.

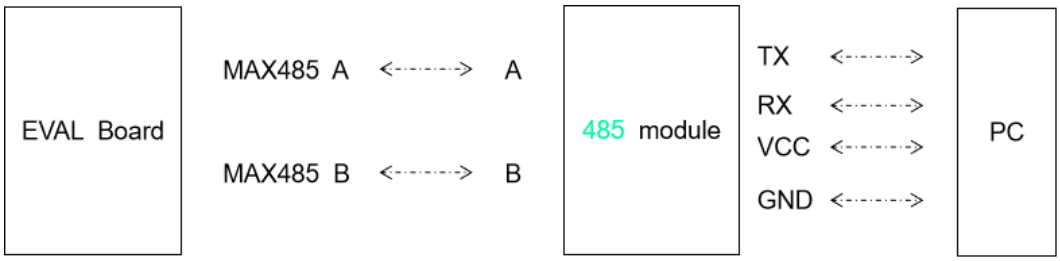
3.5.16 USART Interrupt

This program is designed to show how to send or receive data by interrupt. In this case, USART1 and USART2 send or receive data from each other. Verify after transmission. If the data transmission from USART1 to USART2 is successful, LED2 will light up. If the data transmission from USART2 to USART1 is successful, ED3 will light up.

3.5.17 USART RS485

This routine shows the RS485 communication process. The test process is to send data to the upper computer through the on-board RS485 module; or the upper computer can actively send data to the development board, and the development board will return the same data to the upper computer after receiving the data. RS485 module is a USART3 peripheral connected to MCU.

Figure 23USART RS485



3.5.18 USBD_VCP

This routine implements virtual serial port device through the USB OTG peripheral of APM32F407, and the evaluation board is configured as the Device. When the USB port of the evaluation board is connected to the PC terminal, you can see an additional COM port in the device manager. You can open this COM port through the serial port debugging tool, and then send data to the evaluation board through this COM port. After receiving the data, the evaluation board will return the same data to the serial port debugging tool.

3.5.19 USB HID Mouse

This routine simulates the implementation of a USB mouse through the USB OTG peripheral of APM32F407, and the evaluation board is configured as the Device. When the USB interface of the evaluation board is connected to the PC terminal, you can see an additional mouse device in the device manager. Users can operate the KEY1 - KEY4 keys on the board to simulate movement function of the mouse.

3.5.20 USB MSC Disk

This routine simulates the implementation of a USB device through the USB OTG peripheral of APM32F407, and the evaluation board is configured as the Device. When the USB interface of the evaluation board is connected to the PC terminal, you can see an additional disk device in the device manager. This routine uses internal RAM of APM32F407 to simulate a U disk device.

4 References

For chip specifications and peripheral details, see *APM32F4xxx User Manual*, *APM32F405xG 407xExG Data Manual*, *APM32F407ZG Core Schematic Diagram*, and *APM32F407IG Core Schematic Diagram*. For more technical support, please visit the official website of Geehy: www.geehy.com.

5 Revision History

Table 1 Document Revision History

Date	Revision	Changes
2023.1.18	1.0	New

Statement

This manual is formulated and published by Zhuhai Geehy Semiconductor Co., Ltd. (hereinafter referred to as "Geehy"). The contents in this manual are protected by laws and regulations of trademark, copyright and software copyright. Geehy reserves the right to correct and modify this manual at any time. Please read this manual carefully before using the product. Once you use the product, it means that you (hereinafter referred to as the "users") have known and accepted all the contents of this manual. Users shall use the product in accordance with relevant laws and regulations and the requirements of this manual.

1. Ownership of rights

This manual can only be used in combination with chip products and software products of corresponding models provided by Geehy. Without the prior permission of Geehy, no unit or individual may copy, transcribe, modify, edit or disseminate all or part of the contents of this manual for any reason or in any form.

The "Geehy" or "Geehy" words or graphics with "®" or "TM" in this manual are trademarks of Geehy. Other product or service names displayed on Geehy products are the property of their respective owners.

2. No intellectual property license

Geehy owns all rights, ownership and intellectual property rights involved in this manual.

Geehy shall not be deemed to grant the license or right of any intellectual property to users explicitly or implicitly due to the sale and distribution of Geehy products and this manual.

If any third party's products, services or intellectual property are involved in this manual, it shall not be deemed that Geehy authorizes users to use the aforesaid third party's products, services or intellectual property, unless otherwise agreed in sales order or sales contract of Geehy.

3. Version update

Users can obtain the latest manual of the corresponding products when ordering Geehy products.

If the contents in this manual are inconsistent with Geehy products, the agreement in Geehy sales order or sales contract shall prevail.

4. Information reliability

The relevant data in this manual are obtained from batch test by Geehy Laboratory or cooperative third-party testing organization. However, clerical errors in correction or errors caused by differences in testing environment may occur inevitably. Therefore, users should understand that Geehy does not bear any responsibility for such errors that may occur in this manual. The relevant data in this manual are only used to guide users as performance parameter reference and do not constitute Geehy's guarantee for any product performance.

Users shall select appropriate Geehy products according to their own needs, and effectively verify and test the applicability of Geehy products to confirm that Geehy products meet their own

needs, corresponding standards, safety or other reliability requirements. If loses are caused to users due to the user's failure to fully verify and test Geehy products, Geehy will not bear any responsibility.

5. Compliance requirements

Users shall abide by all applicable local laws and regulations when using this manual and the matching Geehy products. Users shall understand that the products may be restricted by the export, re-export or other laws of the countries of the product suppliers, Geehy, Geehy distributors and users. Users (on behalf of itself, subsidiaries and affiliated enterprises) shall agree and promise to abide by all applicable laws and regulations on the export and re-export of Geehy products and/or technologies and direct products.

6. Disclaimer

This manual is provided by Geehy "as is". To the extent permitted by applicable laws, Geehy does not provide any form of express or implied warranty, including without limitation the warranty of product merchantability and applicability of specific purposes.

Geehy will bear no responsibility for any disputes arising from the subsequent design and use of Geehy products by users.

7. Limitation of liability

In any case, unless required by applicable laws or agreed in writing, Geehy and/or any third party providing this manual "as is" shall not be liable for damages, including any general damages, special direct, indirect or collateral damages arising from the use or no use of the information in this manual (including without limitation data loss or inaccuracy, or losses suffered by users or third parties).

8. Scope of application

The information in this manual replaces the information provided in all previous versions of the manual.