# Readme

## APM32F4xx EVAL

**Release version：V1.0**

# 1 Introduction

The Geehy Semiconductor APM32F4xx EVAL board software development kit includes a series driver library, a group of example applications that demonstrate key peripheral functionality, and other development files.

Software development kit have a hierarchy as follows:

- SDK directory
  - Boards
  - Documents
  - Examples
  - Libraries
  - Middlewares

# Table of Contents


**1 Introduction ..... 1**

**2 About boards ..... 4**

**3 About documents ..... 5**

**4 About examples ..... 6**

4.1 CAN_LoopBack ..... 8

4.2 CAN_Normal ..... 8

4.3 DCI_OV2640 ..... 8

4.4 DMC_SDRAM ..... 9

4.5 ETH_Ping ..... 9

4.6 ETH_TCP_client ..... 9

4.7 ETH_TCP_server ..... 10

4.8 LCD_ShowFigure ..... 10

4.9 LCD_TOUCH ..... 11

4.10 RTC_Alarm ..... 11

4.11 RTC_Calendar ..... 11

4.12 SDIO_SDCard ..... 12

4.13 SPI_Flash ..... 12

4.14 SPI_FullDuplex ..... 13

4.15 USART_Interrupt ..... 13

4.16 USART_Polling ..... 13

4.17 USART_RS485 ..... 14

4.18 USBD_HID ..... 14

4.19 USBD_MSC ..... 14

4.20 USBD_VCP ..... 15

4.21 USBH_CDC ..... 15

4.22 USBH_HID ..... 16

4.23 USBH_MSC ..... 16

**5 About libraries ..... 17**

**6 About middlewares ..... 18**

**7 Revision History ..... 19**

# 2 About boards

The boards folder includes a board support package for APM32F4xx EVAL board. It can help drive the peripheral circuit or components on the board quickly. The BSP can be found in the ~/Boards directory.

The BSP provided are built for APM32F4xx EVAL board compatibility. For other user development board use, some minor modifications may be required.

Boards have a hierarchy as follows:

- Boards folder
  - Board folder
    - inc
    - src
  - Board.c
  - Board.h

Board APM32F407 EVAL include following board support package:

- Board_APM32F407_EVAL src folder
  - Board_APM32F407_EVAL
  - bsp_delay
  - bsp_i2c
  - bsp_lcd
  - bsp_lcdFont
  - bsp_touch
  - bsp_usart

# 3 About documents

The documents folder includes a link file that can be redirected to the technical support center of Geehy semiconductor. The BSP can be found in the ~/Documents directory.

# 4 About examples

The example applications can be found in the ~/Examples directory.

The examples provided are built for APM32F4xx EVAL board compatibility. For other user development board use, some minor modifications may be required.

Example projects have a hierarchy as follows:

- Example folder
  - Include
  - Project
    - IAR
    - MDK
  - Source

All example applications tested with: **APM32F4xx StdPeriphDriver v1.0.1**, include the following examples:

- Examples
  - CAN
    - CAN_LoopBack
    - CAN_Normal
  - DCI
    - DCI_OV2640
  - DMC
    - DMC_SDRAM
  - ETH
    - ETH_Ping
    - ETH_TCP_client
    - ETH_TCP_server

* LCD
  - LCD_ShowFigure
  - LCD_TOUCH
* RTC
  - RTC_Alarm
  - RTC_Calendar
* SDIO
  - SDIO_SDCard
* SPI
  - SPI_Flash
  - SPI_FullDuplex
* USART
  - USART_Interrupt
  - USART_Polling
  - USART_RS485
* USB
  - Device
    - USBD_HID
    - USBD_MSC
    - USBD_VCP
  - Host
    - USBH_CDC
    - USBH_HID
    - USBH_MSC

## 4.1 CAN_LoopBack

### 4.1.1 Example Description

This example describes how to config a communication the CAN in loopback mode. CAN transmit a message to self. Then compare the received message with transmitted message. The data of Polling transmit and Interrupt transmit will displayed on serial assistant through USART1.

### 4.1.2 Directory contents

This example can be found in the ~ /Examples/CAN/LoopBack directory.

## 4.2 CAN_Normal

### 4.2.1 Example Description

This example describes how to config a communication the CAN in normal mode. CAN1 transmit a message to CAN2. Then compare the received message with transmitted message. The result of Polling transmit and Interrupt transmit will displayed on serial assistant through USART1.

- Polling transmit success: The LED2 turns on, printf "CAN polling test passed!". Otherwise LED2 toggles, printf "CAN polling test failed!".

- Interrupt transmit success: The LED3 turns on, printf "CAN interrupt test passed!". Otherwise LED3 toggles, printf "CAN interrupt test failed".

### 4.2.2 Directory contents

This example can be found in the ~ /Examples/CAN/CAN_Normal directory.

## 4.3 DCI_OV2640

### 4.3.1 Example Description

The program aims to show how to use interrupt or DMA to get camera image data by using DCI, in this case, DCI continuously get the data of the OV2640 camera through DMA and sends it to the host by USART2(jpeg mode). In another regb 565 mode, DCI continuously get the data of the OV2640 camera through DMA and transfer it to LCD data register to display.And OV2640 can be set to jpeg and rgb565 data output format.

### 4.3.2 Directory contents

This example can be found in the ~ /Examples/DCI/DCI_OV2640 directory.

## 4.4 DMC_SDRAM

### 4.4.1 Example Description

The program aims to show how to use DMC to read and write data to external SDRAM, the test information will print by USART1.

### 4.4.2 Directory contents

This example can be found in the ~ /Examples/DMC/DMC_SDRAM directory.

## 4.5 ETH_Ping

### 4.5.1 Example Description

This example describes how to use ethernet PHY by using APM32F4xx_ETH_Driver library. After configured ethernet mainboard will be use USART1 to printf static IP address. if computer ping static IP address（192.168.73.22）， computer will be visit mainboard normality.

The phenomenon of data interaction process can be displayed using cmd.exe.

### 4.5.2 Directory contents

This example can be found in the ~ /Examples/ETH/ETH_Ping directory.

## 4.6 ETH_TCP_client

### 4.6.1 Example Description

This example shows how to configure TCP Client to connect server.

phenomenon :

- After initialization, You can see the system information on serial assisatant through USART1 or LCD screen.

- connect server(IP 192.168.73.51:6000) by KEY1. And disconnect server by KEY2.

- you can send data to the EVAL board by TCP server using tcp assisatant.

### 4.6.2 Directory contents

This example can be found in the ~ /Examples/ETH/ETH_TCP_client directory.

## 4.7 ETH_TCP_server

### 4.7.1 Example Description

This example shows how to create TCP server. received and send data to TCP Client by tcp assisatant.

phenomenon :

- After initialization, You can see the system information on serial assisatant through USART1 or LCD screen.

- create server(IP 192.168.73.22:5000) by KEY1. And close server by KEY2.

- you can connect and send data to the EVAL board by tcp assisatant.

### 4.7.2 Directory contents

This example can be found in the ~ /Examples/ETH/ETH_TCP_server directory.

## 4.8 LCD_ShowFigure

### 4.8.1 Example Description

The program aims to show how to use LCD Driver to show Figure on LCD Screen. You can push KEY1 to switch the figures, and the LED will show different status. (LCD Screen is 4.3 inch and 800 x 480 pixe).

### 4.8.2 Directory contents

This example can be found in the ~ /Examples/LCD/LCD_ShowFigure directory.

## 4.9 LCD_TOUCH

### 4.9.1 Example Description

The program aims to show how to use interrupt to get LCD touch chip data by using I2C, in this case, I2C continuously get the data of the touch chip through interrupt and convert it to axis lable data in the window of LCD(4.3 inch and 800 x 480 pixe).

You can touch some button to get control led and get some information in the LCD. Each time a complete touch data is got , LED1 will be toggle.

### 4.9.2 Directory contents

This example can be found in the ~ /Examples/LCD/LCD_TOUCH directory.

## 4.10 RTC_Alarm

### 4.10.1 Example Description

This example shows how to configure RTC and ALARM .

phenomenon :

- After initialization, Alarm begin to count down with LED2 is on. Five second later, Alarm is waking up and LED2 is off.

- You can monitor the system state on serial assisatant through USART1.

### 4.10.2 Directory contents

This example can be found in the ~ /Examples/RTC/RTC_Alarm directory.

## 4.11 RTC_Calendar

### 4.11.1 Example Description

This example shows how to configure RTC time and date to display Calendar.

phenomenon :

- After initialization, LED2 is on. And You can see the Calendar state

- on serial assisatant through USART1 or LCD Screen.

### 4.11.2 Directory contents

This example can be found in the ~ /Examples/RTC/RTC_Calendar directory.

## 4.12 SDIO_SDCard

### 4.12.1 Example Description

The program aims to show how to DMA or polling mode to write and read SD card data by SDIO, in this case, The data can write to SD card or read data from SD card by SDIO. Verification will occur after transmission,

The SD card sector data can be displayed using serial assistant. After power on, can switch between single block test or multi block test by KEY1 and KEY2.

KEY1 ----> single block test

KEY2 ----> multi block test

### 4.12.2 Directory contents

This example can be found in the ~ /Examples/SDIO/SDIO_SDCard directory.

## 4.13 SPI_Flash

### 4.13.1 Example Description

This example shows how to use SPI Peripheral to read and write Flash. Press KEY1 to write data to Flash. And read data form Flash. if read and write success,LED1 will turn on. The data will be shown on serial assistant trough usart1.if error, LED2 will turn on. if initialize the flash error, LED3 will turn on.

### 4.13.2 Directory contents

This example can be found in the ~ /Examples/SPI/SPI_Flash directory.

## 4.14 SPI_FullDuplex

### 4.14.1 Example Description

This example shows how to use SPI Peripheral to transfer data. Press KEY1 to send data from SPI1 to SPI2, if communication success, LED2 will turn on. The data will be shown on serial assistant trough usart1. Press KEY2 to use SPI1 to SPI2 to Realize full-duplex transmission.If communication success, LED3 will turn on. The data will be shown on serial assistant trough usart1.

### 4.14.2 Directory contents

This example can be found in the ~ /Examples/SPI/SPI_FullDuplex directory.

## 4.15 USART_Interrupt

### 4.15.1 Example Description

The program aims to show how to use interrupt to send or received data by using USART, in this case, USART1 and USART2 send or received data to each other. Verification will occur after transmission, if Data transmission pass from USART1 to USART2 , LED2 will be on. If data transmission pass from USART2 to USART1 , LED3 will be on.

The phenomenon of data interaction process can be displayed using serial assistant.

### 4.15.2 Directory contents

This example can be found in the ~ /Examples/USART/USART_Interrupt directory.

## 4.16 USART_Polling

### 4.16.1 Example Description

The program aims to show how to how to use polling to send or received data by using USART, in this case, USART1 and USART2 send or received data to each other. Verification will occur after transmission, if Data transmission pass from USART1 to USART2 , LED2 will be on. If data transmission pass from USART2 to USART1 , LED3 will be on.

The phenomenon of data interaction process can be displayed using serial assistant.

### 4.16.2 Directory contents

This example can be found in the ~ /Examples/USART/USART_Polling directory.

## 4.17 USART_RS485

### 4.17.1 Example Description

This example describes how to use MAX485 module to transfer data with serial COM. When you send some data from PC to USART3(PB10,PB11), then MCU will send the same data to PC.

### 4.17.2 Directory contents

This example can be found in the ~ /Examples/USART/USART_RS485 directory.

## 4.18 USBD_HID

### 4.18.1 Example Description

This example describes how to use the USB OTG device module on APM32F407 to enumerated as a HID Mouse. This example use PC as host, and KEY1 and KEY2 is used to control the direction of the mouse. When KEY1 is pressed, cursor will move left, otherwise KEY2 to right.

### 4.18.2 Directory contents

This example can be found in the ~ /Examples/USB/Device/USBD_HID directory.

## 4.19 USBD_MSC

### 4.19.1 Example Description

This example describes how to use the USB OTG device module on APM32F407 to enumerated as a MSC USB disk. This example use PC as host, and APM32F407 use internal flash to simulate usb flash drives. PC will recognizes the motherboard as a usb flash drives, and formatting the U disk, can read and write files to the U disk.

If define TEST_USB_SPEED = 1,This example will be Used to test usb speed. MSC USB disk will switch storage media. APM32F407 will use sram array to simulate a fake U disk, unable to store files, only Used to test usb speed.

### 4.19.2 Directory contents

This example can be found in the ~ /Examples/USB/Device/USBD_MSC directory.

## 4.20 USBD_VCP

### 4.20.1 Example Description

This example describes how to use the USB OTG device module on APM32F407 to enumerate as a Virtual Com Port. This example use PC as host, you can use serial assistant to transfer USB data. Once serial assistant send data to device through the Virtual Com Port that USB enumerated, then device will send the same data back to PC.

### 4.20.2 Directory contents

This example can be found in the ~ /Examples/USB/Device/USBD_VCP directory.

## 4.21 USBH_CDC

### 4.21.1 Example Description

This example describes how to use the usb host to enum a CDC class device. When press key1(PA1), host will keep receiving data from device when USART1 show "Enable CDC Get data.", press key1 again will toggle the state and host will not receive data from device. When press key2(PA0), host will send data to device.

### 4.21.2 Directory contents

This example can be found in the ~ /Examples/USB/Host/USBH_CDC directory.

## 4.22 USBH_HID

### 4.22.1 Example Description

This example describes how to use the usb host to enum mouse. When mouse move right or left, user can use USART1(PA9/PA10) to catch the track of mouse.

### 4.22.2 Directory contents

This example can be found in the ~ /Examples/USB/Host/USBH_HID directory.

## 4.23 USBH_MSC

### 4.23.1 Example Description

This example describes how to use the usb host to enum a U disk. When press key1(PA1), Host will Scan files on the USB disk. And When press key2(PA0), Host will Write a file to USB disk then readme the file back.

### 4.23.2 Directory contents

This example can be found in the ~ /Examples/USB/Host/USBH_MSC directory.

# 5 About libraries

The libraries folder includes a series library. It can provide supports for APM32F4xx MCU such as device support and standard peripheral and USB OTG etc. The libraries can be found in the ~/Libraries directory.

APM32F4xx MCU include following library:

- Libraries folder
  - APM32F4xx_ETH_Driver
  - APM32F4xx_OTG_Driver
    - Core_Device
    - Core_Host
  - APM32F4xx_StdPeriphDriver
  - CMSIS
  - Device

# 6 About middlewares

The middlewares folder includes a series third-party middleware. The middlewares can be found in the ~/middlewares directory.

The middlewares used by APM32F4xx EVAL include following:

- Middlewares folder
  - fat_fs
  - lwip-1.4.1

# 7 Revision History

Table 1 File Revision History

| Date | Rev | Description |
|---|---|---|
| 2022.05.31 | 1.0 | First Release version of APM32F4xx EVAL SDK |