User Manual

APM32F072 EVAL Board

Version: V1.0

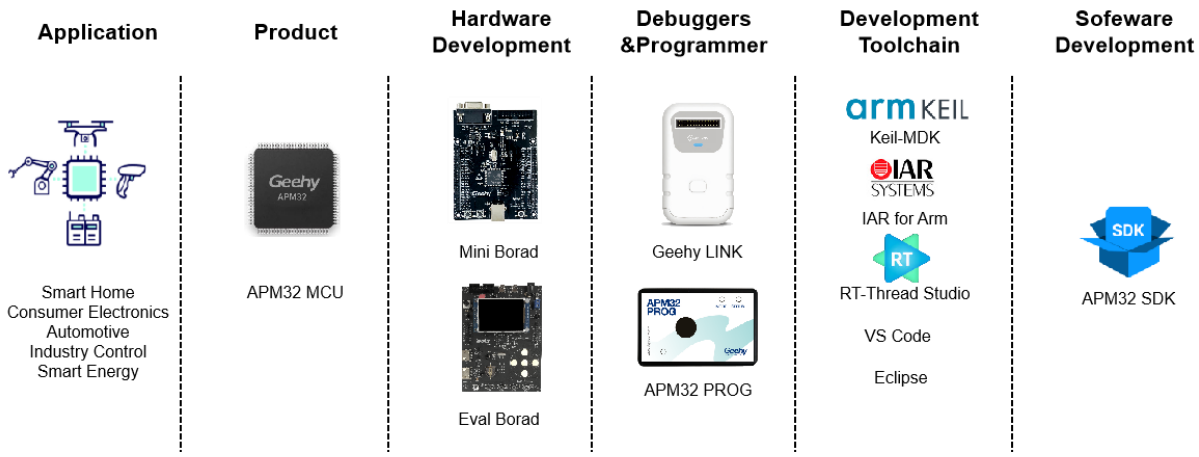
1 Introduction

This User Manual mainly describes the functions, on-board resources and supporting SDK of APM32F072VB 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.

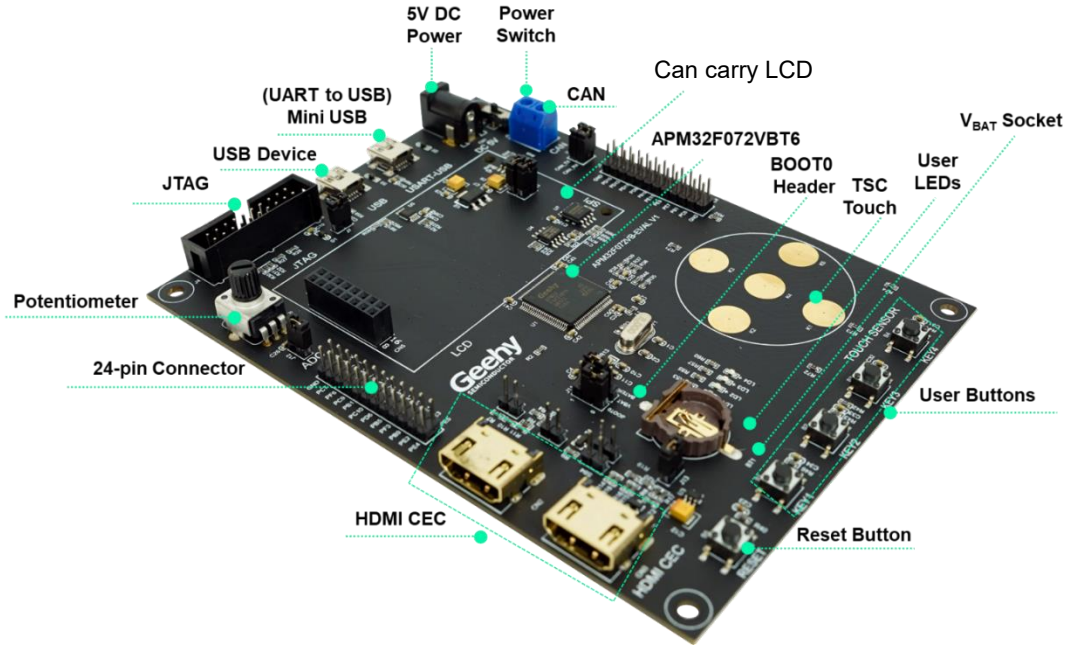
Figure 1 APM32 Ecosystem



1.2 Evaluation board

The APM32F072VB EVAL Board is a complete demonstration and development platform for the basic APM32F0xx series MCU, which is used to demonstrate RS485, RTC, TSC and other peripheral functions. It carries an APM32F072VBT6 MCU chip, which is based on ARM Cortex-M0+core, with operating frequency 48MHz, Flash 128KB, and has a standard JTAG interface and supports users to simulate and download programs.

Figure 2 APM32F072VB EVAL Board



Contents

2 Function overview

The APM32F072VB EVAL Board includes the following peripheral functions, and is equipped with the EVAL SDK, which can help developers evaluate chip performance or develop related applications

1.TFT LCD

2.EEPROM

3.SPI FLASH

4.Touch Sensor Keyboard

5.Potentiometer

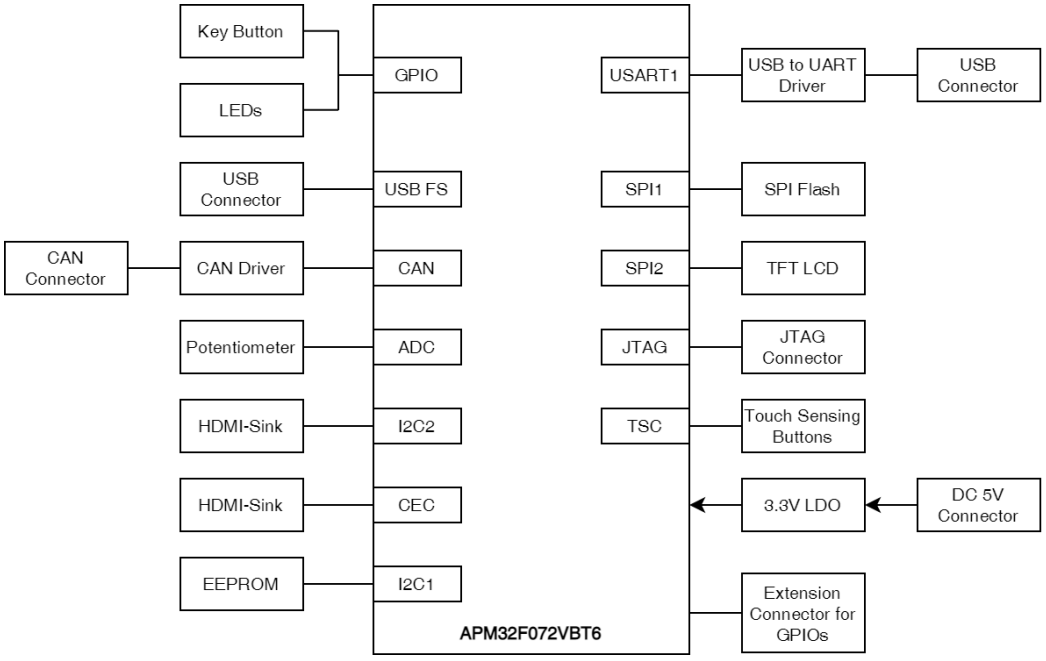
6.USB Device

7.HDMI-CEC

8.Button x 4

9.LED x 4

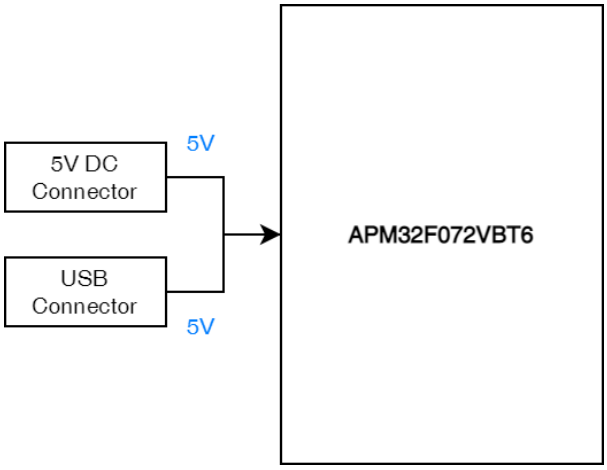
Figure 3 APM32F072VB EVAL Board Overview



2.1 Power supply

APM32F072VB EVAL Board can be powered by external 5V DC power supply or USB. JTAG interface can supply power to MCU, and other required voltages are provided by on-board voltage regulator.

Figure 4 Power Supply of Evaluation Board



2.2 Battery interface

On-board CR1220 standard battery interface.

2.3 Clock

APM32F072VB EVAL Board carries two external clocks, including:

- X1, 32.768KHz clock for RTC
- X2, 8MHz clock is used for APM32F072VBT6. 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

Two reset controls are provided:

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

2.5 Simulation and download interface

A standard 20-pin IDC JTAG connector on board.

2.6 LCD screen

A SPI LCD control interface is provided. The supporting routine is 2.4-inch TFT LCD screen with 240 x 320 pixels, and the control chip is ILI9341. The LCD screen is driven through MCU SPI interface.

2.7 LED lights

4 general red LED lights on board, which can be used for display.

2.8 Keys

4 keys provided, which can be used for LCD menu switching or other input purposes.

2.9 EEPROM

Onboard B EEPROM chip, which can be driven by I2C peripheral.

2.10 Flash

A Flash chip on board, which provides 2MB external storage space and is driven by SPI.

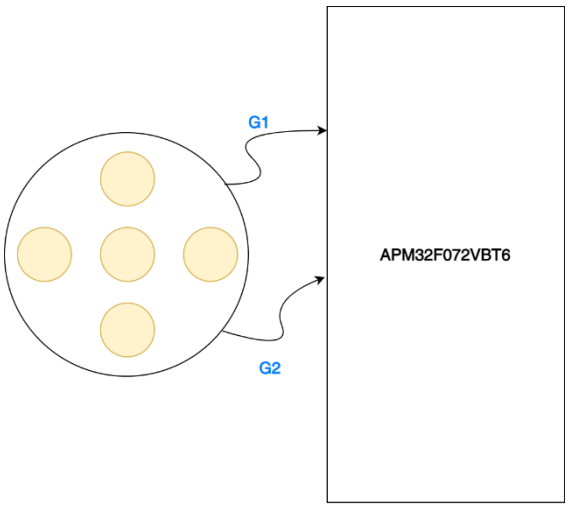
2.11 HDMI CEC interface

Provide CAN2 and CAN3 two HDMI-CEC interfaces.

2.12 Touch keys

5 touch keys on board, which are connected to two capacitive sensing channels in Group 1 and three capacitive sensing channels in Group 2.

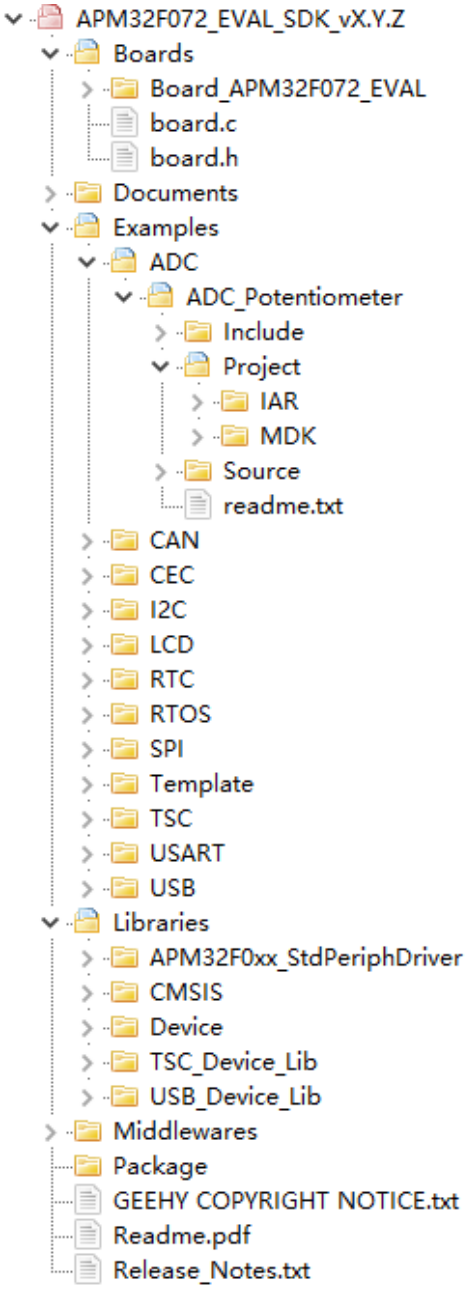
Figure 5 Touch Key Signal Connection



3 SDK Overview

The SDK [1] is provided in the form of compressed packages, including on-board driver packages, such as basic LED, Button and COM drivers, I2C for EEPROM driver packages, LCD screen driver packages and W25Q16 Flash driver packages, as well as multiple necessary libraries, such as APM32F0xx standard library, TSC device library, USB device library and many applications that are easy to reuse, such as CAN double-computer communication, EEPROM reading and writing, LCD multi-level menu, RTC calendar, TSC, and HDMI-CEC double-computer communication.

Figure 6 SDK Directory Structure

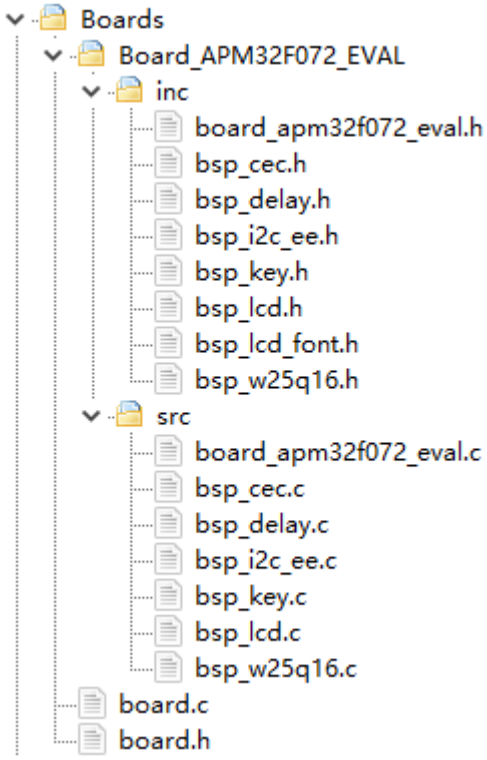


[1]. There is no package supported by IDE in the SDK of APM32xx_EVAL Board. You need to obtain it on the official website of Geehy www.geehy.com.

3.1 On-board driver

The on-board drivers contain basic LED, Button and COM drivers, I2C for EEPROM driver package, LCD screen driver package, W25Q16 Flash and HDMI CEC driver package.

Figure 7 On-board Driver



3.2 Library file

The library files contain the APM32F0xx standard library, USB device library and TSC device library.

3.3 IDE support

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

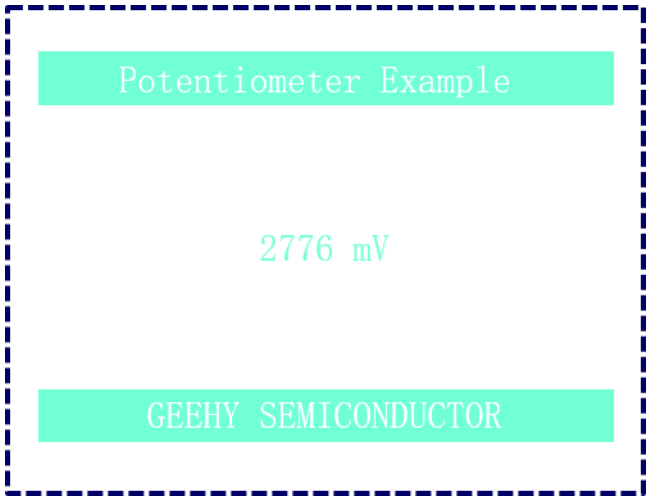
3.4 Routine

The routine folder contains many applications that are easy to reuse, such as CAN double-computer communication, EEPROM reading and writing, LCD multi-level menu, RTC calendar, TSC, and HDMI-CEC double-computer communication.

3.4.1 ADC

The ADC uses a 10K potentiometer as the sampling source, which can be adjusted by the knob. The AD sampling channel is channel 9. After downloading and running of the routine, the change of the voltage value sampled by ADC will be displayed in real time on the on-board LCD.

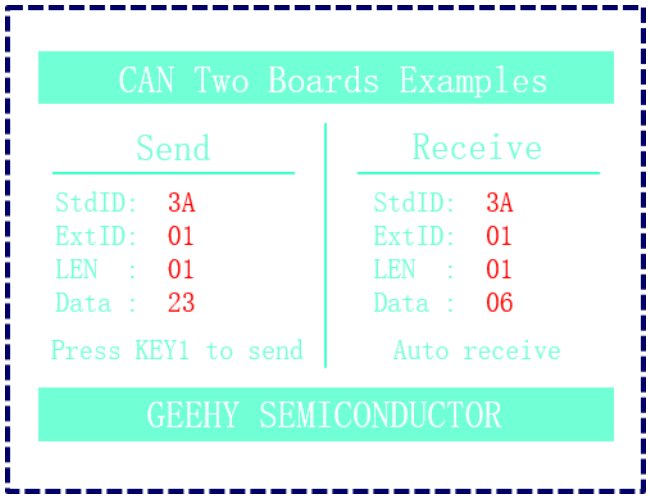
Figure 8 ADC Menu



3.4.2 CAN TwoBoards

The CEC TwoBoardsExchange routine is used to show how to use CAN to communicate between two APM32F072VB EVAL Boards. These two boards can download the same firmware.

Figure 9 CAN Two Boards Menu



3.4.3 I2C EEPROM

EEPROM adopts AT24C32, the device address is 0xA0, and the addressing address is 16bit. After downloading and running of the routine, the routine test process is to write and read 255 data to EEPROM and compare them.

Figure 10 I2C EEPROM Menu

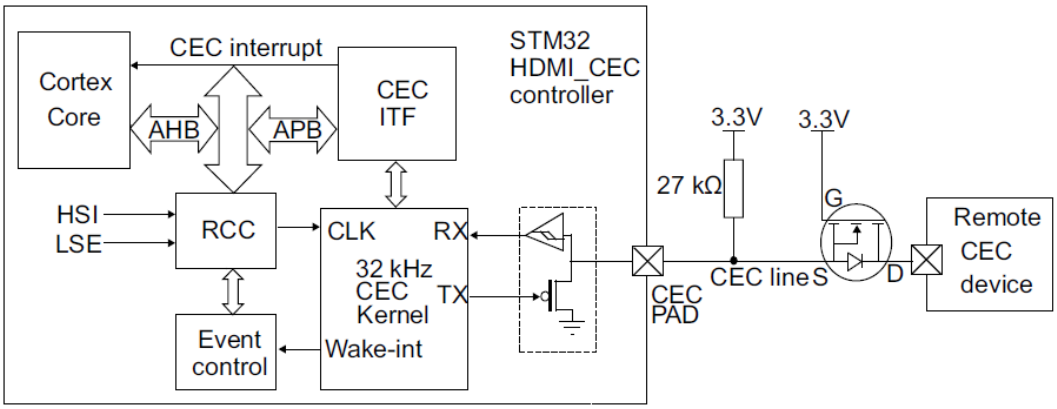
3.4.4 CEC TwoBoardsExchange

3.4.4.1 CEC introduction

CEC is a single-bus protocol and part of HDMI standard. AV products can discover devices and communication between systems. CEC enables each device connected through HDMI to operate with each other, thus reducing the number of IR remote controllers and keys required for basic operation of the system.

The CEC bus consists of a two-way line, which is used to transfer data to and from the device. It is connected to +3.3V power supply voltage through 27kΩ pull-up resistor.

Figure 11CEC Hardware Structure

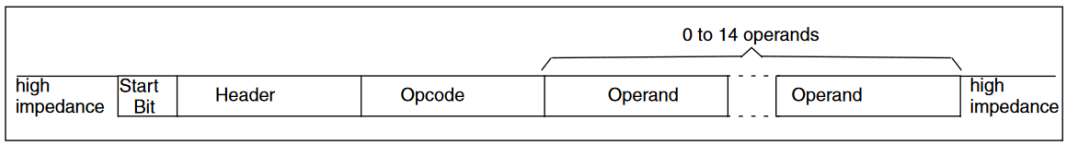


3.4.4.2 Figure 11 CEC Hardware Structure

All transmissions on the CEC line are composed of an initiator and one or more followers. The initiator is responsible for transmitting the message structure and data. The follower is the receiver of any data and is responsible for setting the response bit.

Transmission of a single message frame includes a start bit, a header block, an opcode (optional), and a variable number of operand blocks (optional).

Figure 12CEC Message Structure



3.4.4.3 Message block

The message blocks include Header Block and Opcode/Operand Block. Each block consists of EOM (End Of Message) and ACK. Where:

EOM：

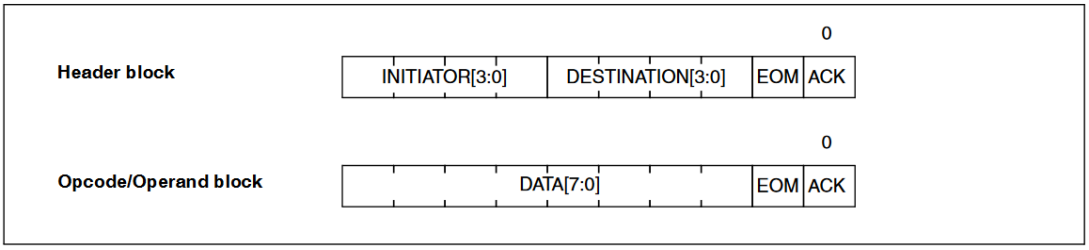
- 0: Followed by one or more data blocks
- 1: Message transmission is completed

ACK：

- 0: The receiver requests messages
- 1：NACK

The INITIATOR address and DESTINATION address in the header block occupy 4 bits respectively. The data block contains 8-bits valid data.

Figure 13 Message Block



3.4.4.4 Physical address

Each device has a physical address in order to enable CEC to address specific physical devices. When HPD finds that the device connection status changes, DDC will display the data channel/extended display identification data.

3.4.4.5 Logical address

In addition to the physical address, each device appearing on the control signal line has a unique logical address. This address defines the device type and unique identifier. Logical addresses are assigned only when the physical address of the device is valid. The logical address table is as follows. Each logical address is fixed with the device identifier.

Figure 14CEC Logical Address

Address	Device
0	TV
1	Recording device 1
2	Recording device 2
3	Tuner 1
4	Playback device 1
5	Audio system
6	Tuner 2
7	Tuner 3
8	Playback device 2
9	Recording device 3
10	Tuner 4
11	Playback device 3
12	Reserved
13	Reserved
14	Free use
15	Unregistered (as initiator address) Broadcast (as destination address)

3.4.4.6 Opcode

Each different Opcode code represents different functions and has its action scope. Please see *CEC_HDMI_Specification v1.3.pdf* for details. The following table shows the Opcode of User Control Pressed, and the code is 0x44.

Figure 15CEC OPCODE

Opcode	value	Description	Parameters	Parameter description	Response	Directly addressed	Broadcast	Mandatory for Initiator	Mandatory for Follower
<User Control Pressed> [4]	0x44	Used to indicate that the user pressed a remote control button or switched from one remote control button to another.	[UI Command]	UI command issued by user.	Update display or perform an action, as required.	•			

3.4.4.7 Operand

Each Opcode code is attached with an Operand. The following table shows the Operand of User Control Pressed. Different Operand ID represents different functions. For example, the Operand of volume increase is 0x41, and the Operand of volume decrease is 0x42.

Figure 16CEC Operand

Operation id	User Operation
0x00	**Select**
0x01	**Up**
0x02	**Down**
0x03	**Left**
0x04	**Right**
0x05	**Right-Up**
0x06	**Right-Down**
0x07	**Left-Up**
0x08	**Left-Down**
0x09	Root Menu
0x0A	Setup Menu
0x0B	Contents Menu
0x0C	Favorite Menu
0x0D	**Exit**
0x0E - 0x1F	Reserved
0x20 - 0x29	**Numbers 0-9**
0x2A	Dot
0x2B	Enter
0x2C	Clear
0x2D - 0x2F	Reserved
0x30	Channel Up
0x31	Channel Down

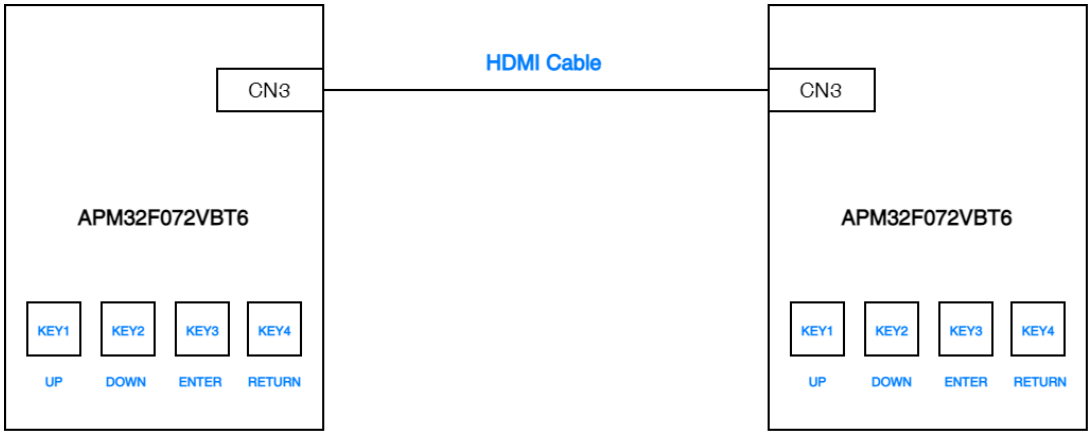
Operation id	User Operation
0x32	Previous Channel
0x33	Sound Select
0x34	Input Select
0x35	Display Information
0x36	Help
0x37	Page Up
0x38	Page Down
0x39 - 0x3F	Reserved
0x40	Power
0x41	Volume Up
0x42	Volume Down
0x43	Mute
0x44	Play
0x45	Stop
0x46	Pause
0x47	Record
0x48	Rewind
0x49	Fast forward
0x4A	Eject
0x4B	Forward
0x4C	Backward
0x4D - 0x4F	Reserved

Operation id	User Operation
0x50	Angle
0x51	Sub picture
0x52 - 0x5F	Reserved
0x60	Play Function
0x61	Pause-Play Function
0x62	Record Function
0x63	Pause-Record Function
0x64	Stop Function
0x65	Mute Function
0x66	Restore Volume Function
0x67	Tune Function
0x68	Select Disk Function
0x69	Select A/V Input Function
0x6A	Select Audio Input Function
0x6B - 0x70	Reserved
0x71	**F1 (Blue)**
0x72	**F2 (Red)**
0x73	**F3 (Green)**
0x74	**F4 (Yellow)**
0x75	**F5**
0x76 - 0x7F	Reserved

3.4.4.8 Hardware connection

The CEC TwoBoardsExchange routine is used to show how to use CEC to communicate between two APM32F072VB EVAL Boards (through CN3). These two boards can download the same firmware, and the address can be selected from the menu.

Figure 17Hardware Connection



3.4.4.9 Figure 17 Hardware Connection

Select the address of the device in the "Select device address" menu. Note that the two development boards need to select different device addresses.

Figure 18Select Device Address

Select the address to enter the initialization menu.

Figure 19Initialization Menu

3.4.4.10 CEC operation menu

In the Send area of the CEC operation menu, you can select to send the preset Volume Up/Volume Down/Mute/Play/Stop command to the destination address by pressing the UP/DOWN key; at the same time, the received command information will be displayed in the Receive area of the menu.

Figure 20Send Command Menu

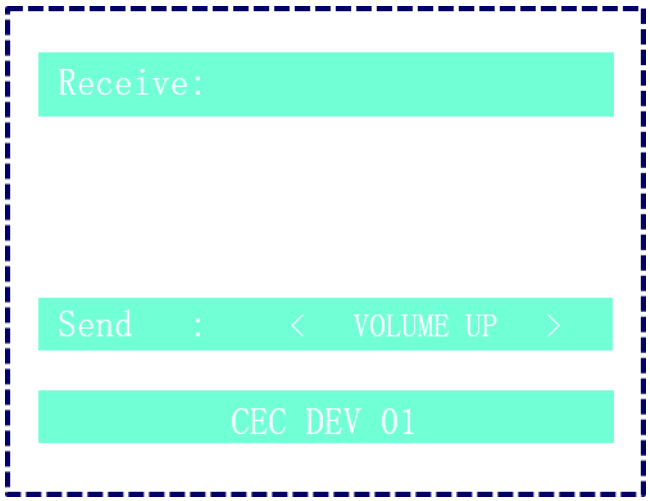
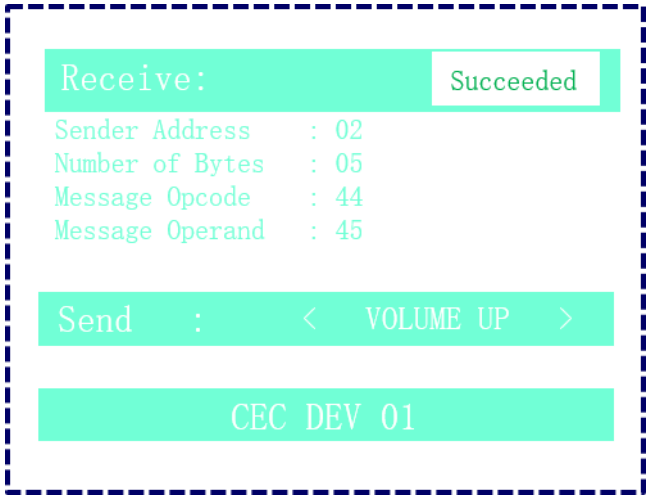


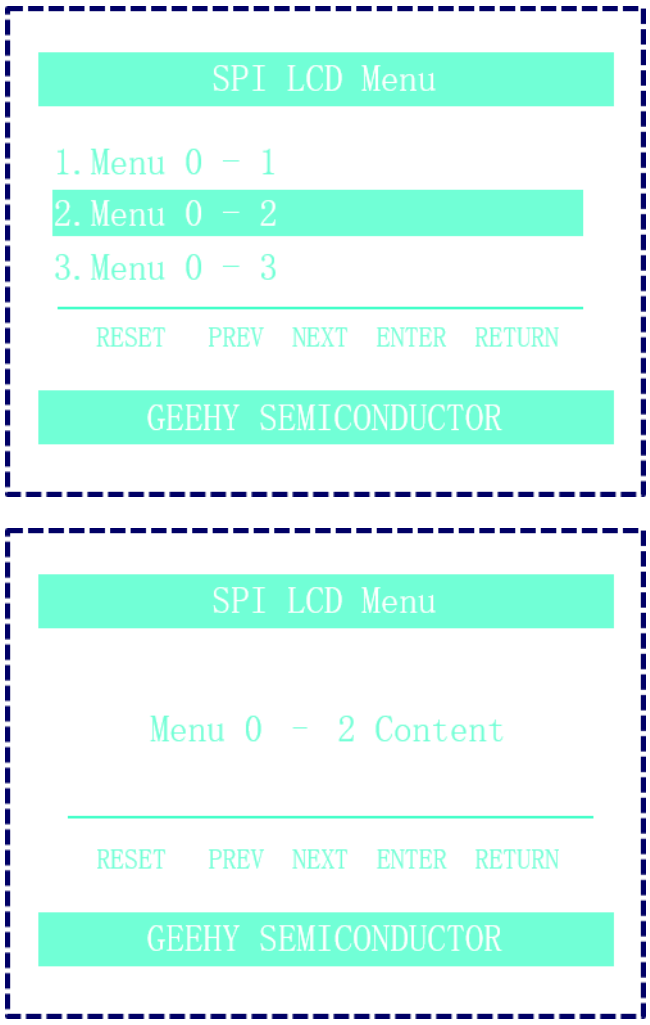
Figure 21Receive Command Menu



3.4.5 LCD Menu

This routine uses SPI to drive a 2.4- inch LCD screen to display a sample of multi-level menu.

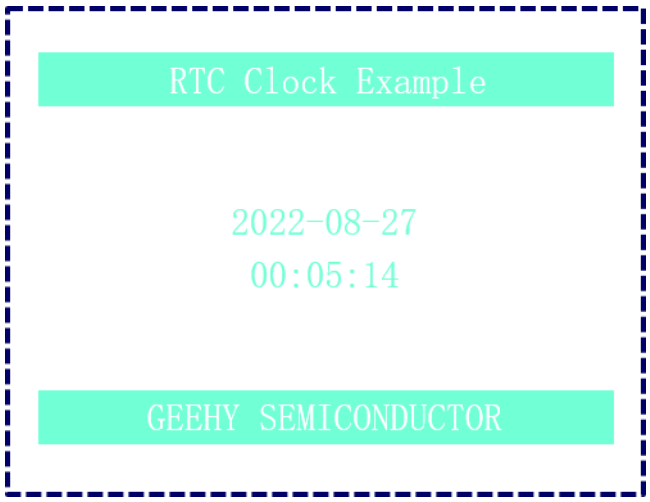
Figure 22 LCD Menu



3.4.6 RTC Clock

This routine uses RTC and displays an example of RTC calendar in combination with LCD screen.

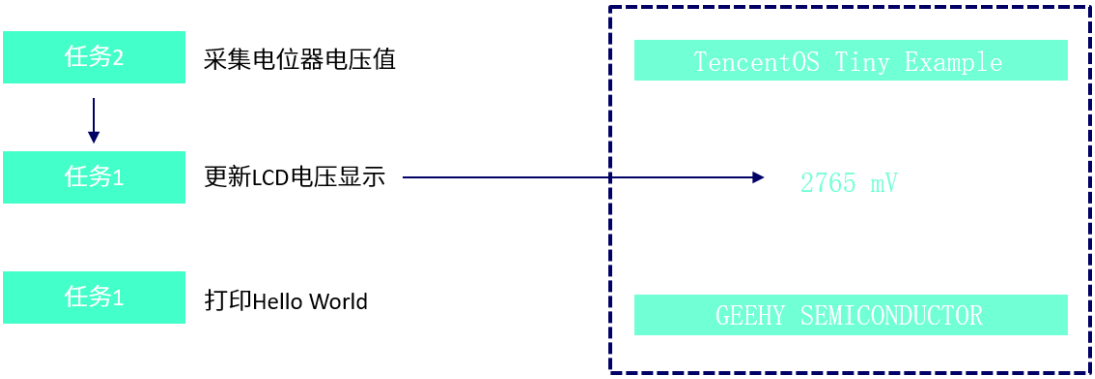
Figure 23 RTC Menu



3.4.7 TencentOS Tiny

TencentOS tiny is a real-time operating system developed by Tencent for the field of the Internet of Things. It has the characteristics of low power consumption, low resource occupancy, modularization, safety and reliability, and can effectively improve the development efficiency of terminal products of the Internet of Things. The routine in the SDK shows a simple multi-task management example.

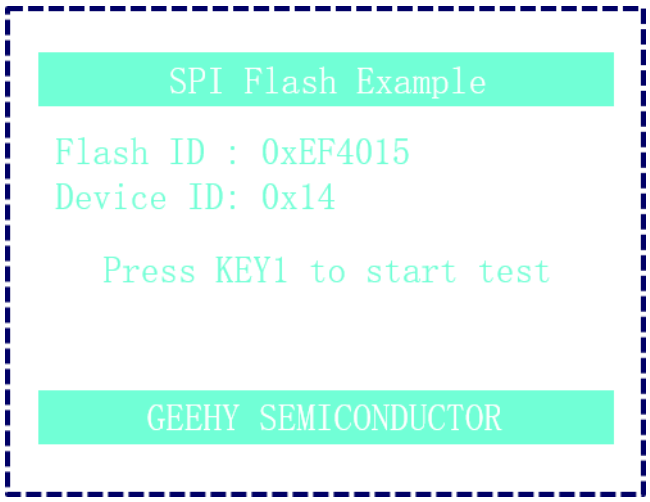
Figure 24 TencentOS Tiny Menu



3.4.8 SPI Flash

This routine uses SPI to access the external flash chip W25Q16, with a size of 2MB. The test process is to write and read data to Flash and compare them.

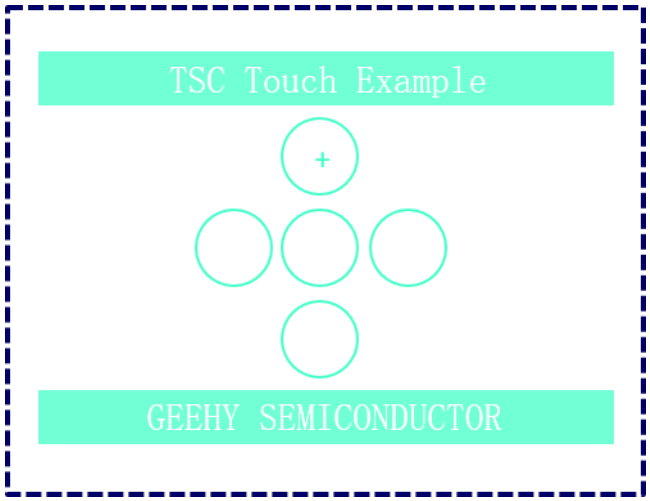
Figure 25 SPI Flash Menu



3.4.9 TSC

The TSC Key Liner Rotate routine mainly displays the response results of touch keys.

Figure 26 TSC Key Liner Rotate Menu



The touch sensing function of APM32 is implemented based on the principle of surface charge transfer and acquisition. The principle is to charge the sensor capacitor Cx and transfer the accumulated charge to the sampling capacitor Cs. This process will be repeated until the voltage at both ends of the sampling capacitor Cs reaches the threshold. The number of charge transfer times required to reach the threshold is used to indicate the size of the capacitance Cx to be measured.

When sensor KEYx is touched, the capacitance of the sensor to the ground increases, the number of charge transfer times required for the voltage at both ends of the sampling capacitor Cs to reach the threshold decreases, and the measured value is reduced. When the measured value is lower than the threshold value, TSC Device Lib will generate a detection event.

3.4.10 USART Polling

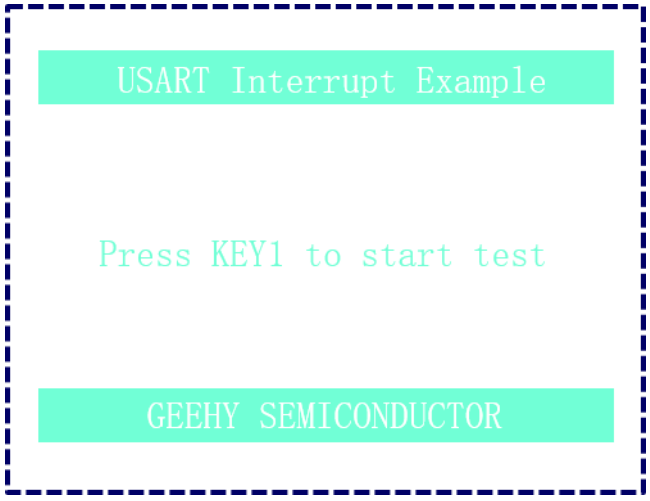
This routine shows how to transmit and receive serial port data by polling. The test process is to send the string '123456' to the upper computer through USART1, the upper computer returns the same string '123456', and then the lower computer checks and matches, and finally displays the matching results.

Figure 27 USART Polling Menu

3.4.11 USART Interrupt

This routine shows how to send serial port data by interrupt. The test process is to send a string to the upper computer through USART1.

Figure 28 USART Interrupt Menu



3.4.12 USB CDC Virtual COM Port

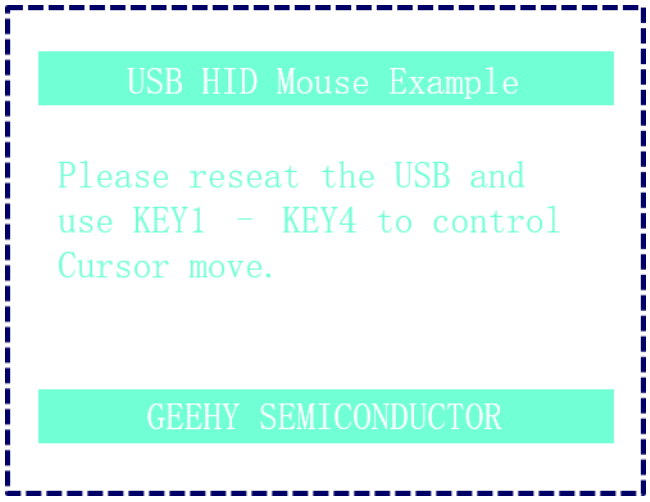
This routine shows the implementation of USB CDC Virtual COM. After the device is connected to the PC, and the serial port assistant sends data to the device through the Virtual COM port, the device will return the same data to the PC.

Figure 29USB CDC Virtual COM Port Menu

3.4.13 USB HID Mouse

This routine shows the implementation of HID Mouse. Use KEY1 – KEY4 to simulate up/down/left/right sliding operation of the mouse.

Figure 30USB HID Mouse



3.4.14 USB MSC Disk

This routine shows the implementation of MSC Disk. Use APM32F072VB internal RAM to simulate a U disk.

Figure 31 USB MSC Disk Menu

4 References

For chip specifications and peripheral details, see *APM32F072x8xB Data Manual*, *APM32F072x8xB User Manual* and *APM32F072VB EVAL Board 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.