

GigaDevice Semiconductor Inc.

GD32450Z-AliOS Things Demo

用户手册

V1.1

目录

目录.....	1
表	2
1 简介	3
2 功能引脚分配	3
3 入门指南	6
4 硬件设计概述	6
4.1 供电电源	6
4.2 启动方式选择	7
4.3 LED 指示灯	7
4.4 按键	7
4.5 串口	8
4.6 模数转换器	8
4.7 数模转换器	8
4.8 I2S	9
4.9 I2C	9
4.10 SPI	9
4.11 CAN	10
4.12 以太网	10
4.13 SDIO	11
4.14 SDRAM	12
4.15 LCD	12
4.16 USBFS	13
4.17 USBHS	14
4.18 扩展电路	14
4.19 GD-Link	15
4.20 WIFI 模组	15
5 例程使用指南	15
5.1 开始前准备	15
5.2 开发板配套 DEMO	17
6 版本更新历史	23

表

Table 1 引脚分配表	3
Table 2 版本更新历史.....	23

1 简介

GD32450Z-EVAL 评估板使用 GD32F450ZKT6 作为主控制器。评估板使用 Mini USB 接口或者 DC-005 连接器提供 5V 电源。提供包括扩展引脚在内的及 SWD, Reset, Boot, User button key, LED, CAN, I2C, I2S, USART, RTC, LCD, SPI, ADC, DAC, EXMC, CTC, SDIO, ENET, USBFS, USBHS, GD-Link 等外设资源。更多关于开发板的资料可以查看 GD32450Z-EVAL-V1.1 原理图。

2 功能引脚分配

Table 1 引脚分配表

Function	Pin	Description
LED	PD4	LED1
	PD5	LED2
	PG3	LED3
RESET		K1-Reset
KEY	PA0	K2-Warkup
	PC13	K3-Tamper
	PB14	K4-User key
USART0	PA9	USART0_TX
	PA10	USART0_RX
ADC	PF6	ADC012_IN4
DAC	PA4	DAC_OUT0
I2C	PB6	I2C0_SCL
	PB7	I2C0_SDA
SPI	PG10	SPI5_IO2
	PG11	SPI5_IO3
	PG13	SPI5_SCK
	PG14	SPI5_MOSI
	PG12	SPI5_MISO
	PG9	SPI5_CS
I2S	PC6	I2S1_MCK
	PC7	I2S1_CK
	PB9	I2S1_WS
	PC1	I2S1_SD
CAN	PD0	CAN0_RX
	PD1	CAN0_TX
SDRAM	PC0	EXMC_SDNWE
	PC2	EXMC_SDNE0
	PC3	EXMC_SDCKE0
	PD0	EXMC_D2

	PD1	EXMC_D3
	PD8	EXMC_D13
	PD9	EXMC_D14
	PD10	EXMC_D15
	PD14	EXMC_D0
	PD15	EXMC_D1
	PE0	EXMC_NBL0
	PE1	EXMC_NBL1
	PE7	EXMC_D4
	PE8	EXMC_D5
	PE9	EXMC_D6
	PE10	EXMC_D7
	PE11	EXMC_D8
	PE12	EXMC_D9
	PE13	EXMC_D10
	PE14	EXMC_D11
	PE15	EXMC_D12
	PF0	EXMC_A0
	PF1	EXMC_A1
	PF2	EXMC_A2
	PF3	EXMC_A3
	PF4	EXMC_A4
	PF5	EXMC_A5
	PF11	EXMC_NRAS
	PF12	EXMC_A6
	PF13	EXMC_A7
	PF14	EXMC_A8
	PF15	EXMC_A9
	PG0	EXMC_A10
	PG1	EXMC_A11
	PG2	EXMC_A12
	PG4	EXMC_A14
	PG5	EXMC_A15
	PG8	EXMC_SDCLK
	PG15	EXMC_NCAS
SDIO	PD2	SDIO_CMD
	PC12	SDIO_CK
	PC8	SDIO_D0
	PC9	SDIO_D1
	PC10	SDIO_D2
	PC11	SDIO_D3
LCD	PC6	TLI_HSYNC

		PA4	TLI_VSYNC
		PG7	TLI_PIXCLK
		PF10	TLI_DE
		PG6	TLI_R7
		PA8	TLI_R6
		PA12	TLI_R5
		PA11	TLI_R4
		PB0	TLI_R3
		PD3	TLI_G7
		PC7	TLI_G6
		PB11	TLI_G5
		PB10	TLI_G4
		PG10	TLI_G3
		PA6	TLI_G2
		PB9	TLI_B7
		PB8	TLI_B6
		PA3	TLI_B5
		PG12	TLI_B4
		PG11	TLI_B3
Ethernet		PA1	ETH_RMII_REF_CLK
		PA2	ETH_MDIO
		PA7	ETH_RMII_CRS_DV
		PB11	ETH_RMII_TX_EN
		PB12	ETH_RMII_TXD0
		PB13	ETH_RMII_TXD1
		PC1	ETH_MDC
		PC4	ETH_RMII_RXD0
		PC5	ETH_RMII_RXD1
USB_FS		PA9	USB_VBUS
		PA11	USB_DM
		PA12	USB_DP
		PD13	USB_VBUS_CTRL
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

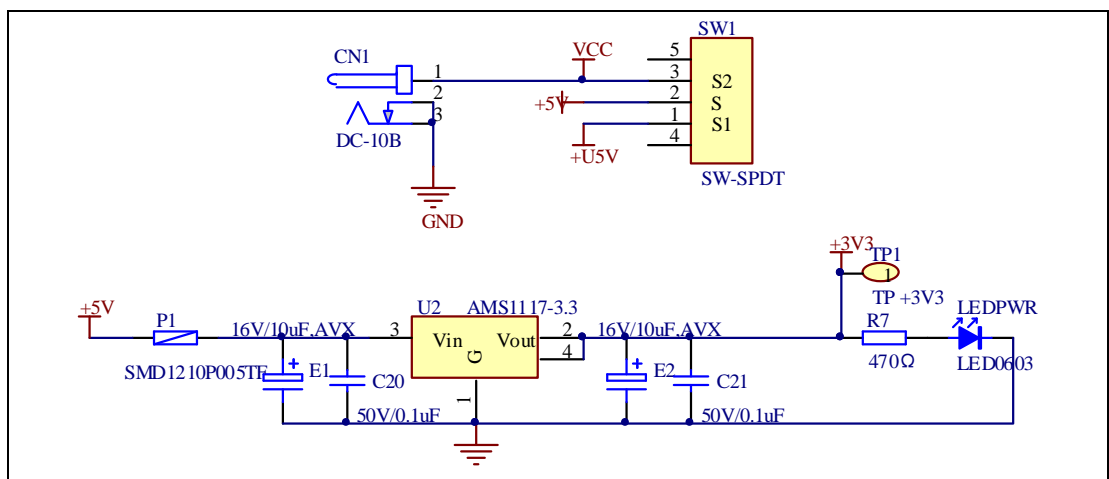
3 入门指南

评估板使用 Mini USB 或者 DC-005 连接器提供 5V 电源。下载程序到评估板需要一套 J-Link 或者使用 GD-Link 工具，在选择了正确的启动方式并且上电后，LEDPWR 将被点亮，表明评估板供电正常。

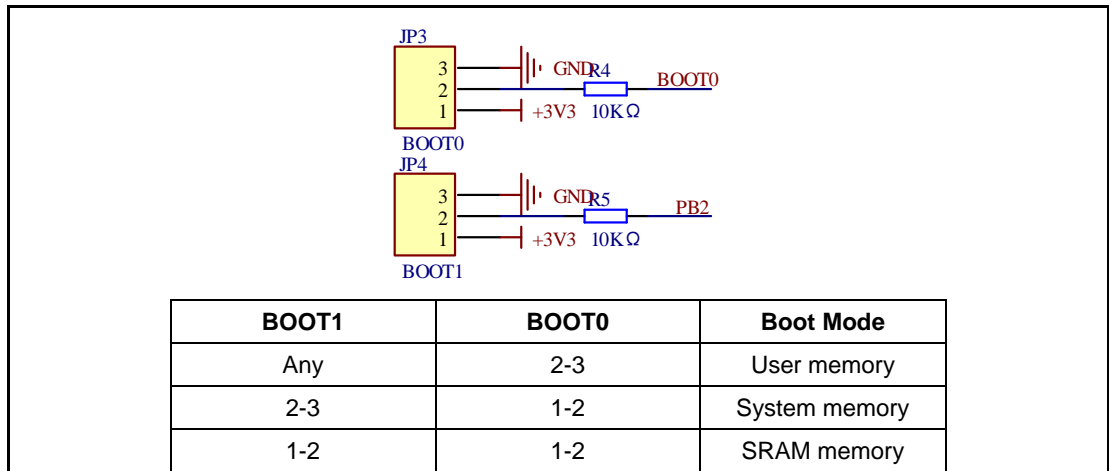
- 1、如果使用 Keil uVision4 打开工程，安装\Library\Firmware\GD32F4xx_Addon.1.0.0.exe，以加载相关文件；
- 2、如果使用 Keil uVision5 打开工程，有两种方法解决“Missing Device(s)”问题。第一种是方法先安装\Library\Firmware\GigaDevice.GD32F4xx_DFP.1.0.0.pack，在 Project 菜单中选择 Manage 子菜单，点击 Migrate to Version 5 Format...菜单，将 Keil uVision4 工程转为 Keil uVision5 工程，同时在 Option for Target 的 C/C++ 中添加路径 C:\Keil_v5\ARM\Pack\ARM\CMSIS\4.2.0\CMSIS\Include；第二种方法是直接安装 Addon，在 Folder Selection 中的 Destination Folder 那一栏选择 Keil uVision5 软件的安装目录，如 C:\Keil_v5，然后在 Option for Target 的 Device 选择对应的器件，同时在 Option for Target 的 C/C++ 中添加路径 C:\Keil_v5\ARM\Pack\ARM\CMSIS\4.2.0\CMSIS\Include。
- 3、如果使用 IAR 打开工程，安装\Library\Firmware\IAR_ GD32F4xx_Addon.exe，以加载相关文件。

4 硬件设计概述

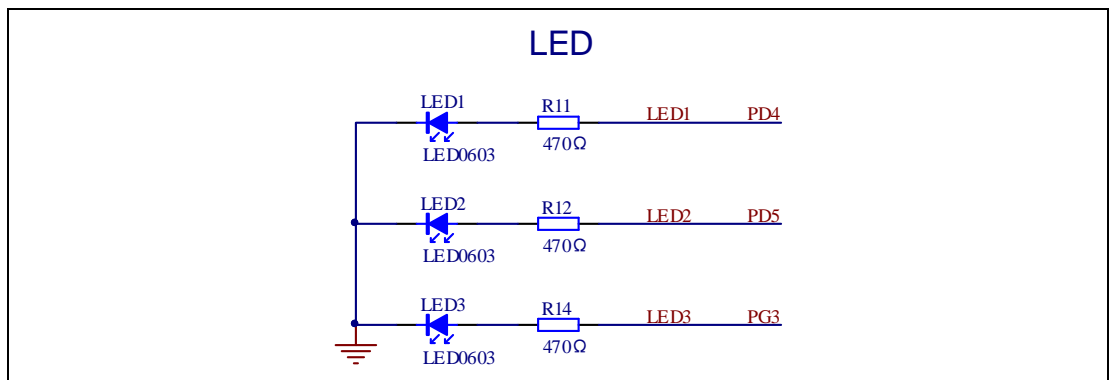
4.1 供电电源



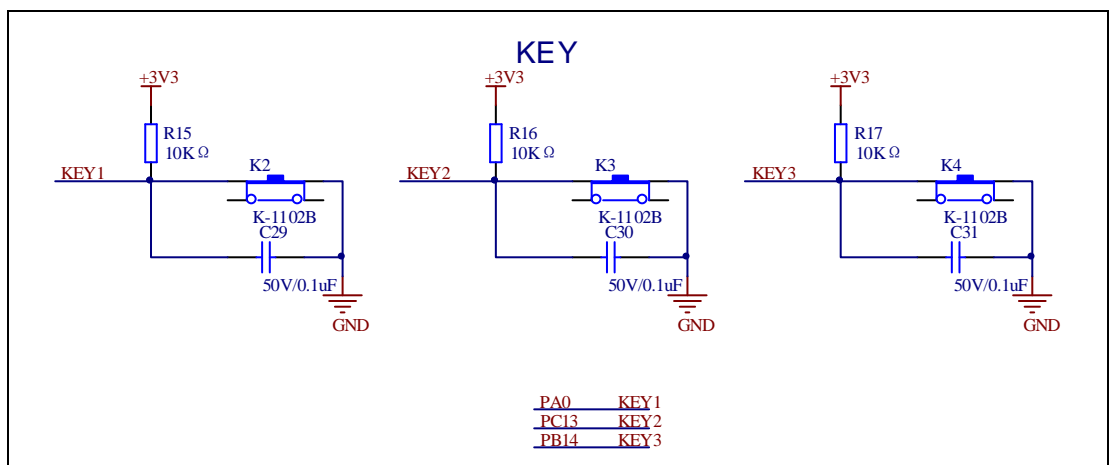
4.2 启动方式选择



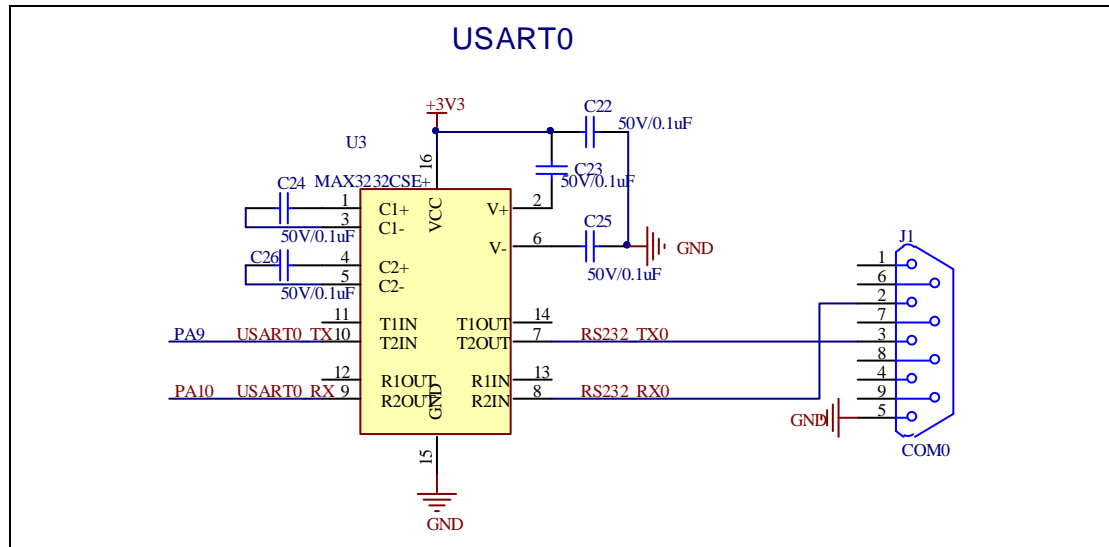
4.3 LED 指示灯



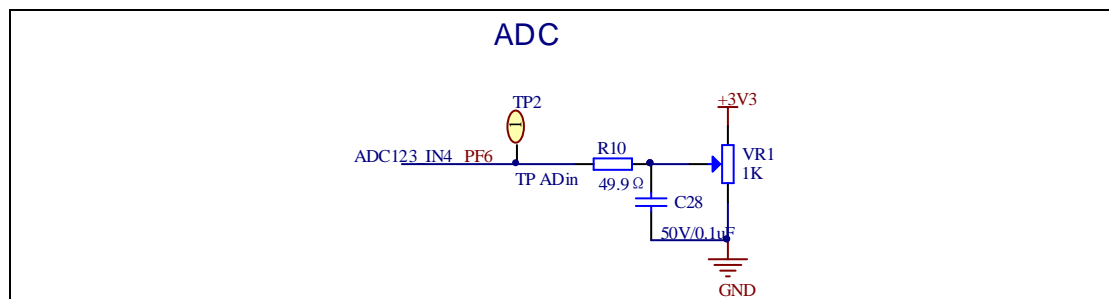
4.4 按键



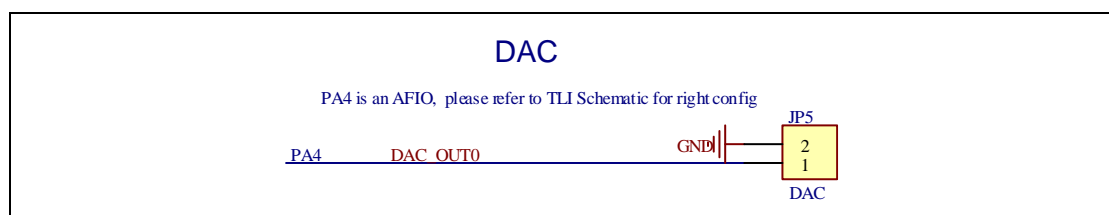
4.5 串口



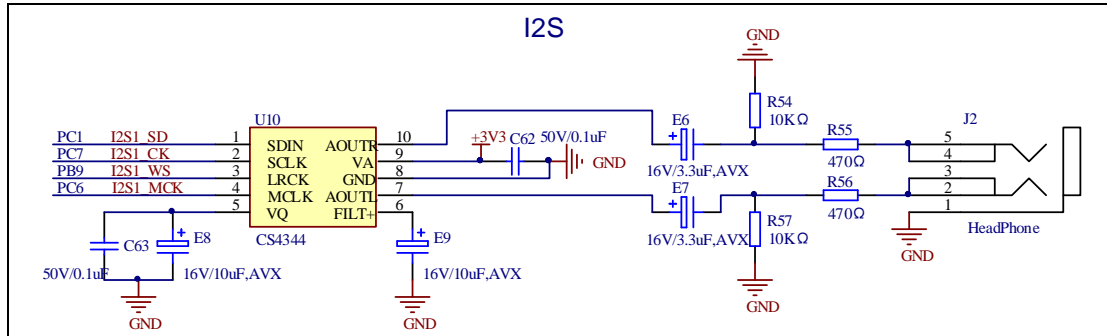
4.6 模数转换器



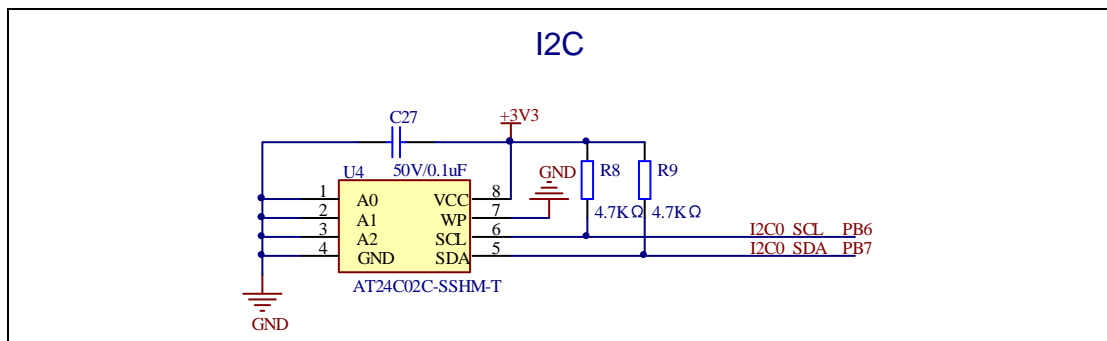
4.7 数模转换器



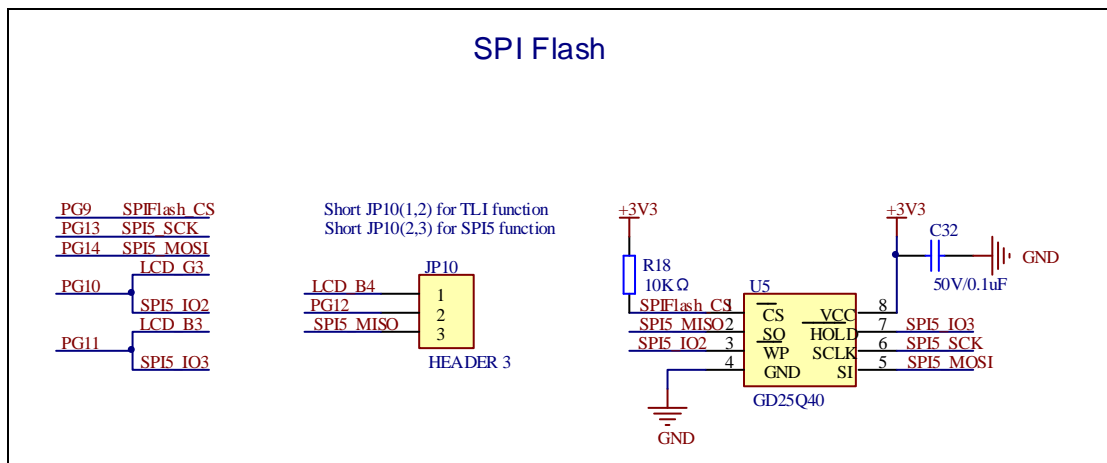
4.8 I2S



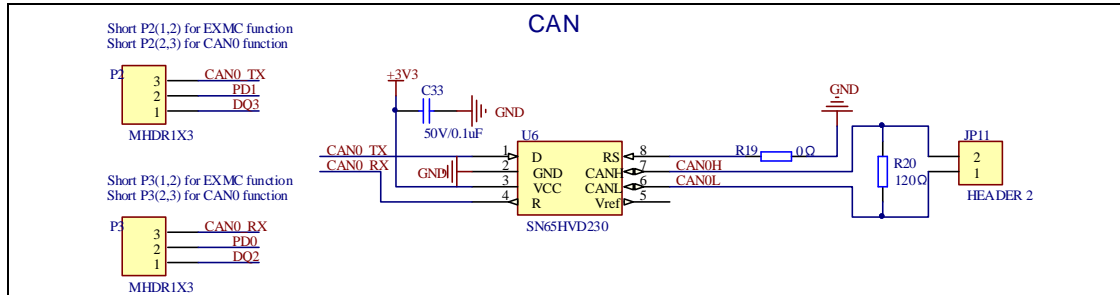
4.9 I2C



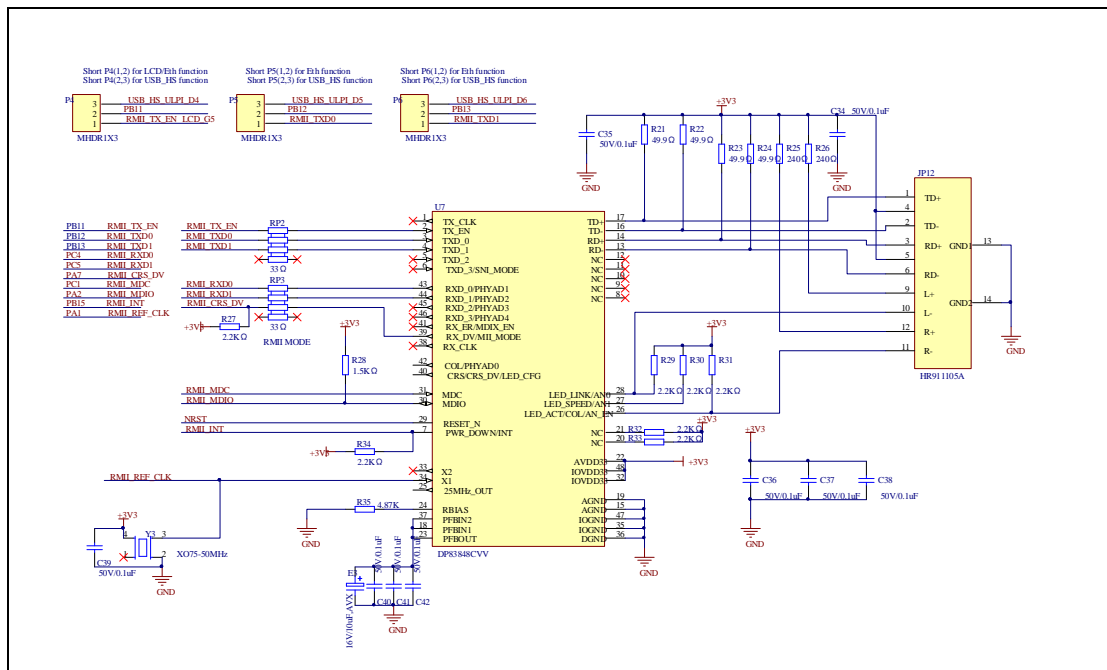
4.10 SPI



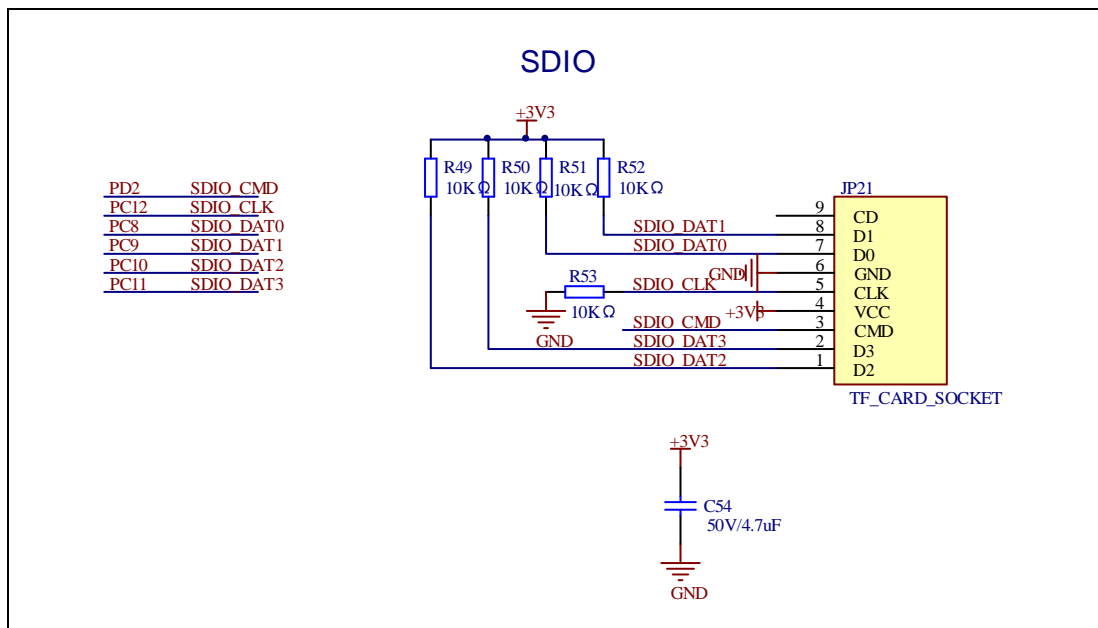
4.11 CAN



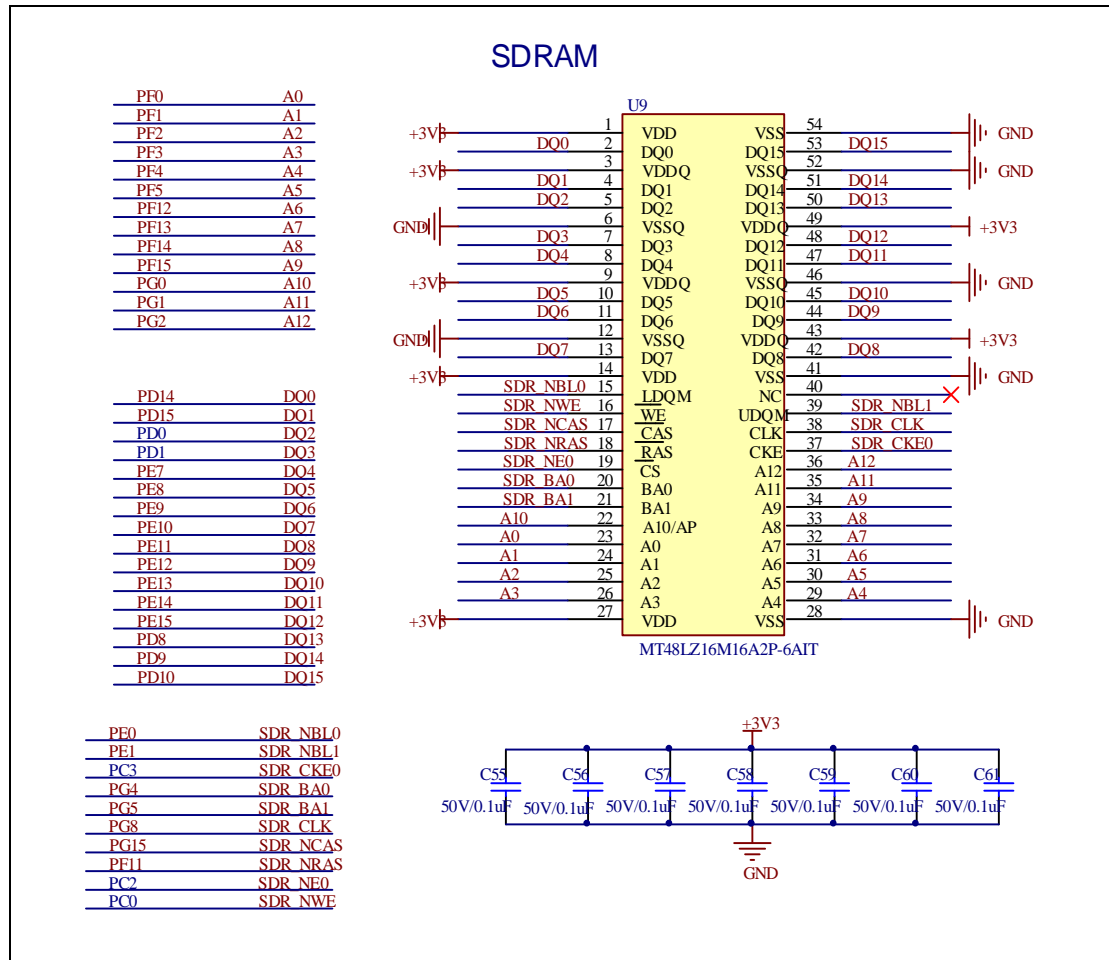
4.12 以太网



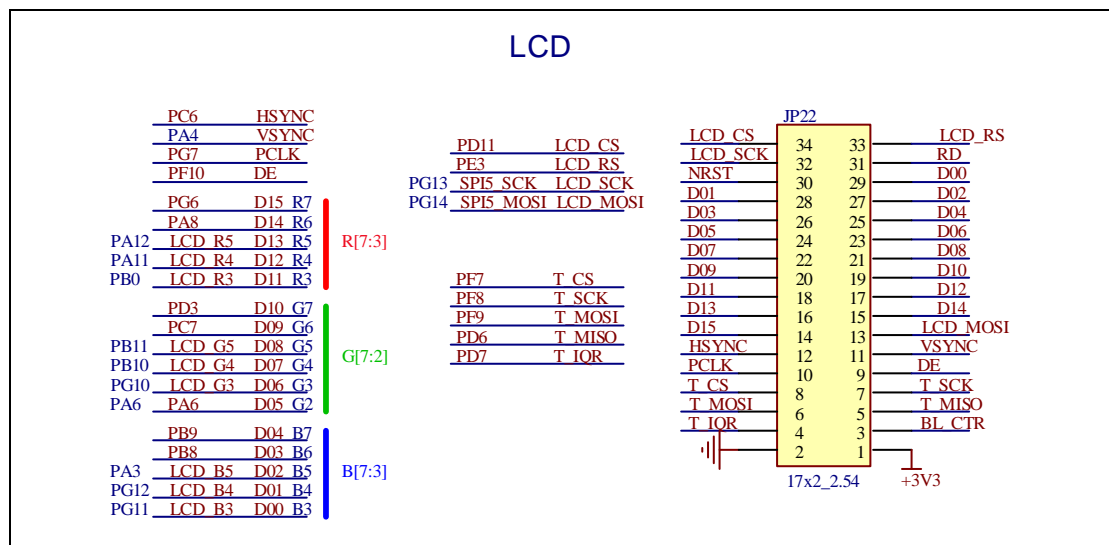
4.13 SDIO



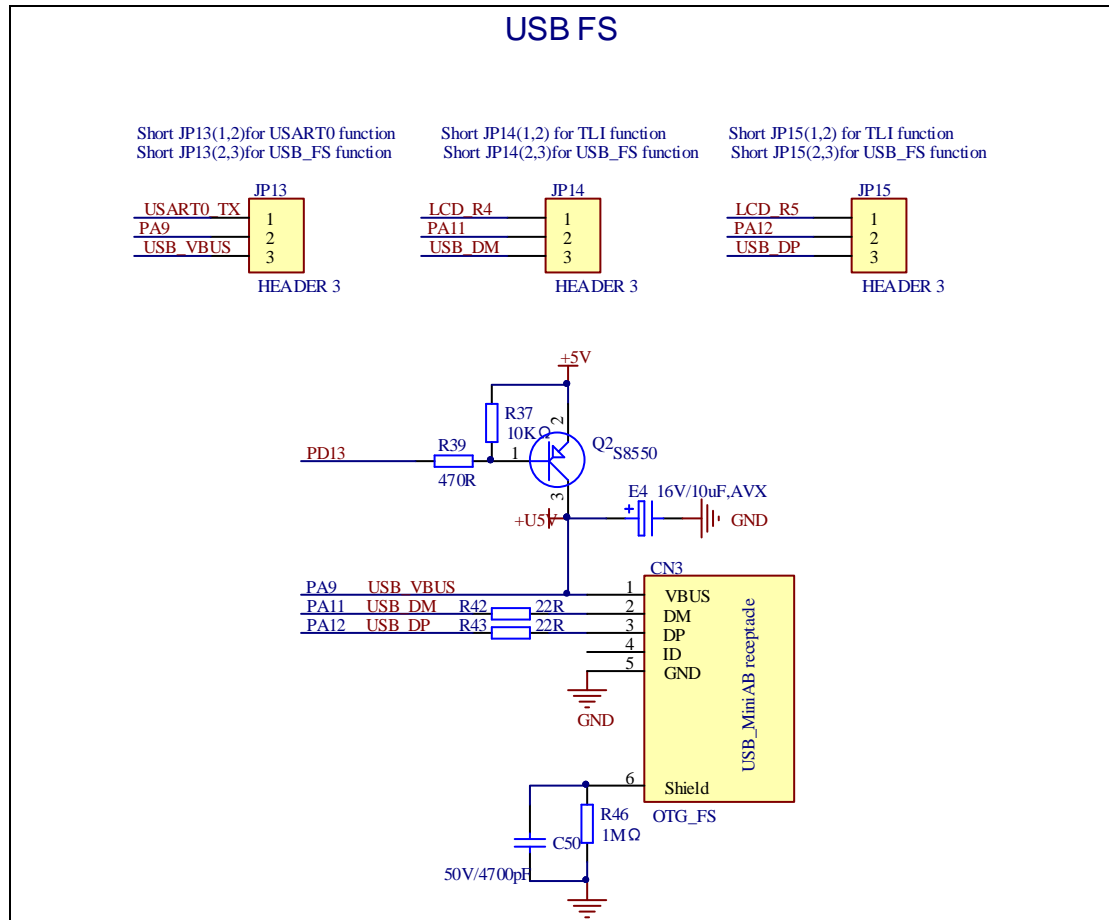
4.14 SDRAM



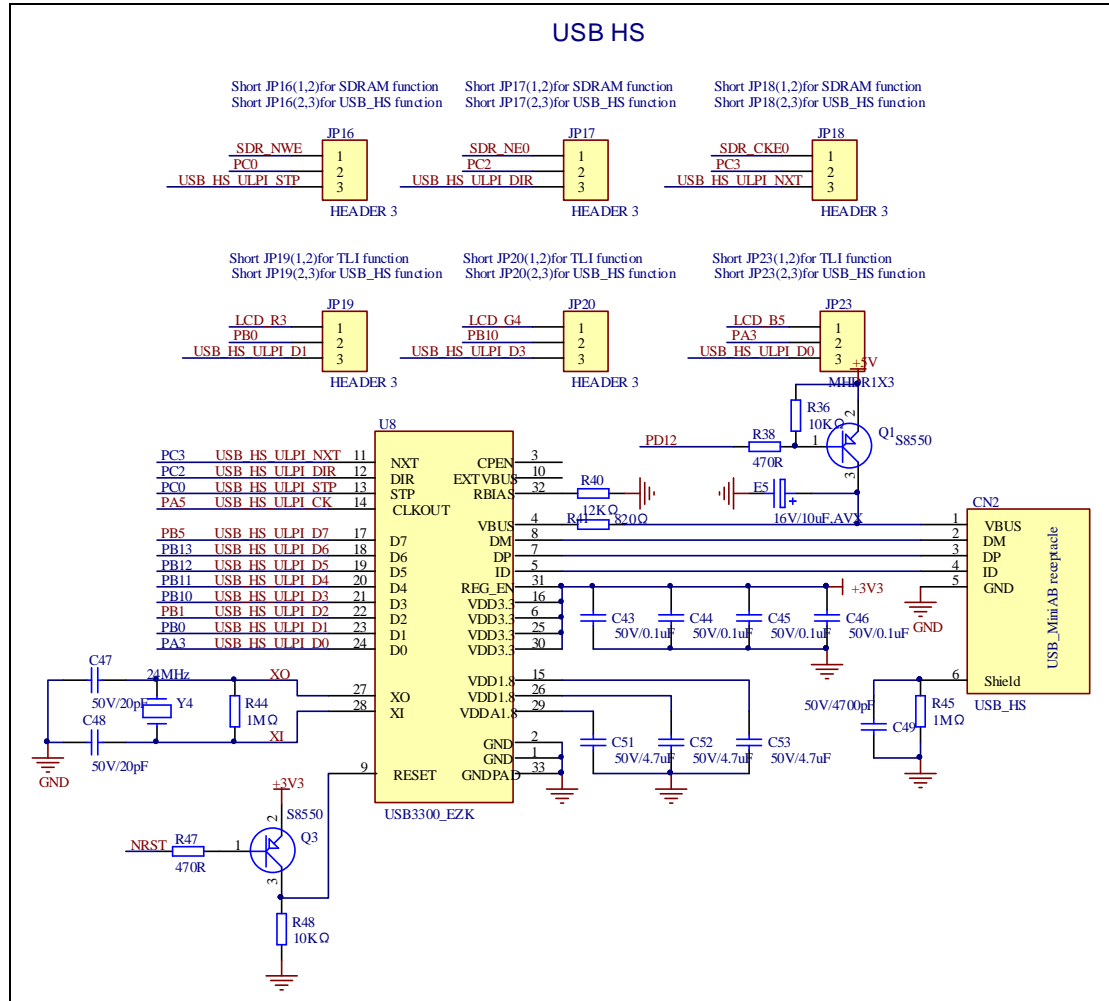
4.15 LCD



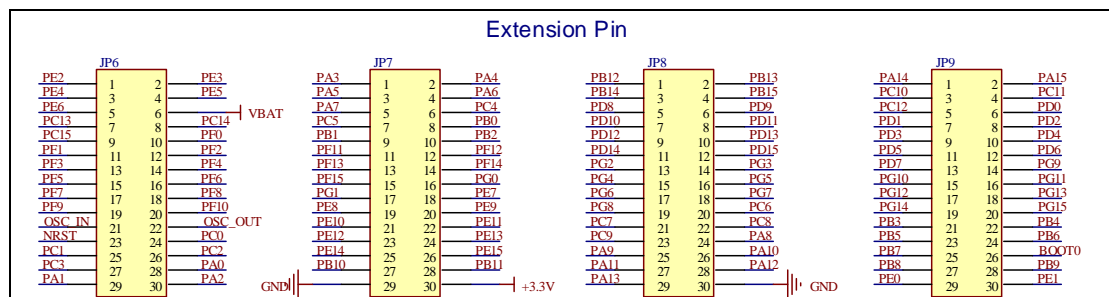
4.16 USBFS



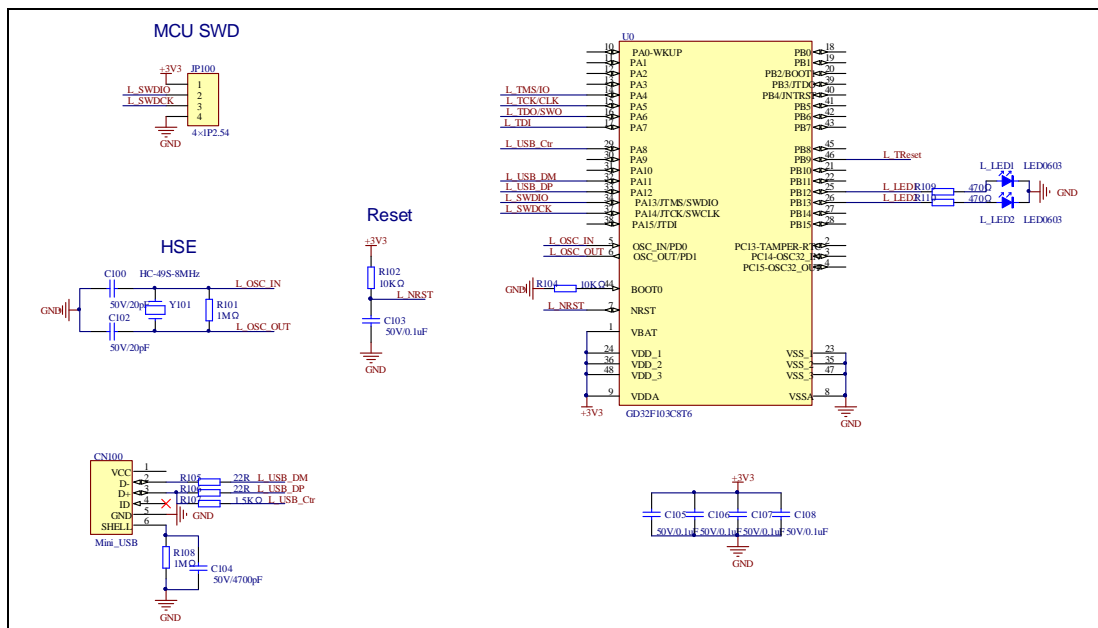
4.17 USBHS



4.18 扩展电路



4.19 GD-Link



4.20 WIFI 模组

WIFI 模组采用庆科 MK3060 模组，详细资料请参考《EMW3060_datasheet》及《RM0087CN_MiCOKit-3060_硬件手册_V0.1》

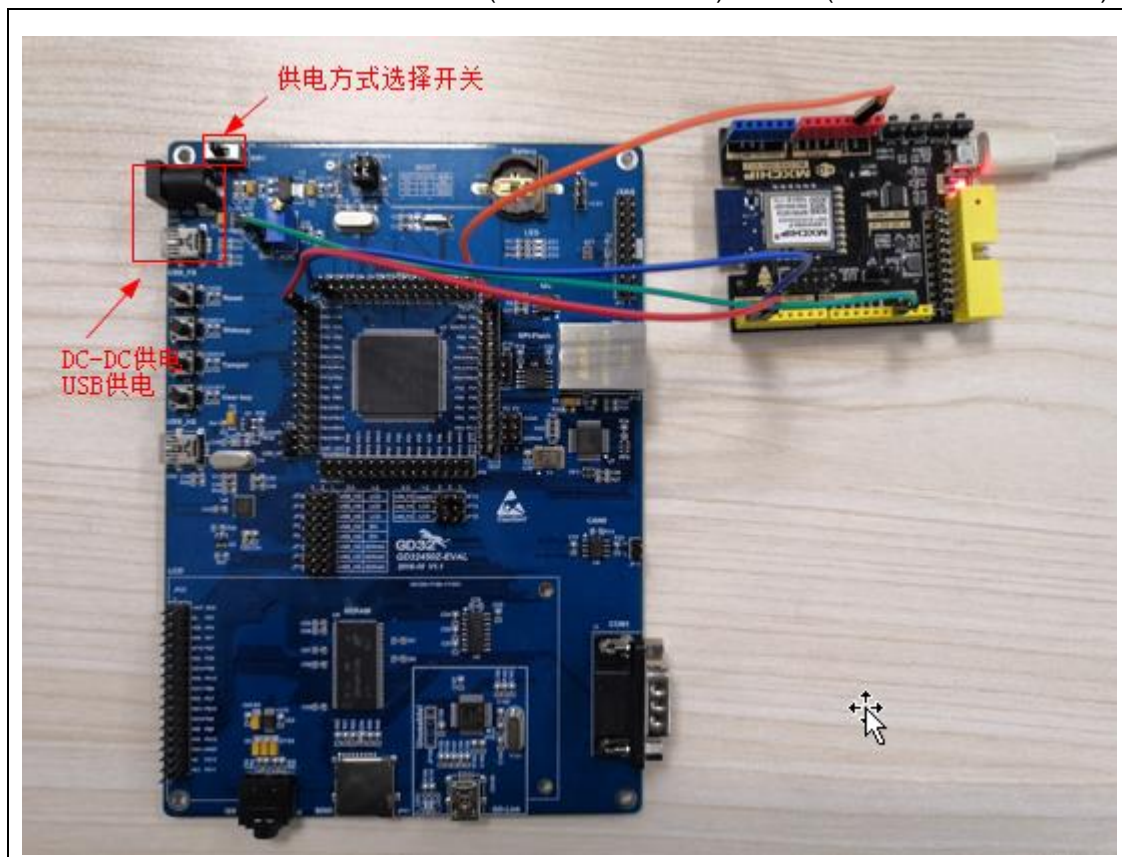
5 例程使用指南

5.1 开始前准备

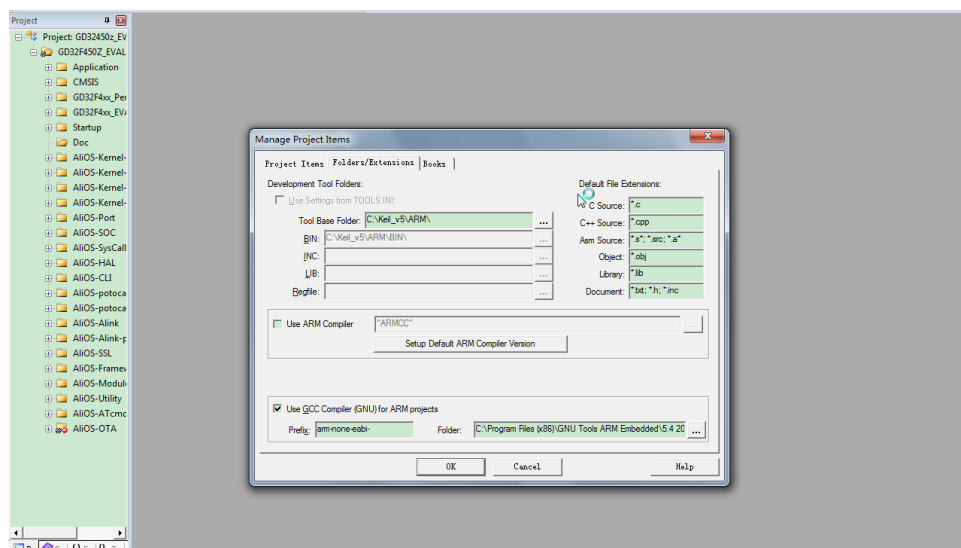
在使用开发板之前，请先按照以下步骤确认硬件配置：

1. 通过跳线帽 JP13 选择串口输出。并将 COM1 口通过串口与 PC 相连，用于 DEMO 的交互操作及 LOG 打印显示。
2. 开发板供电可以选择 5V DC-DC/或 USB 供电。
3. 开发板与 WIFI 模组之间通过串口通讯，WIFI 模组运行在 RAW ENET 模式，该模式下 WIFI 模组只负责数据的转发，因此网络协议栈及阿里云相关协议均在 MCU 中运行。
4. DEMO 默认使用 USART1 与 WIFI 模组进行通讯，使用引脚：PA2(MCU TX -> WIFI RX)，PA3(MCU RX <- WIFI TX)，PE0(MCU PE0 -> WIFI RESET)，GND(MCU GND <-> WIFI GNE)。波特率：115200（最高可手动配置到 921600）；如果需要 WIFI 模组将 LOG 打印到 PC，则需要单独使用 USB 线连接 WIFI 模组供电。

5. 开发板支持的烧写工具为 GD-Link(USB 接口: CN100)及 J-link(JTAG/SWD 接口: JP2)



6. 本 DEMO 使用的 AliOS 源码尚不支持 ARMCC 编译器，如需使用 ARMCC 编译器进行开发，可从 github 上下载最新版本并在 projects 文件夹下找到工程文件，地址为 <https://github.com/alibaba/AliOS-Things>
7. 本 DEMO 需要使用 GCC 编译器进行编译。需要对 KEIL 使用的编译器进行配置，配置截图如下：

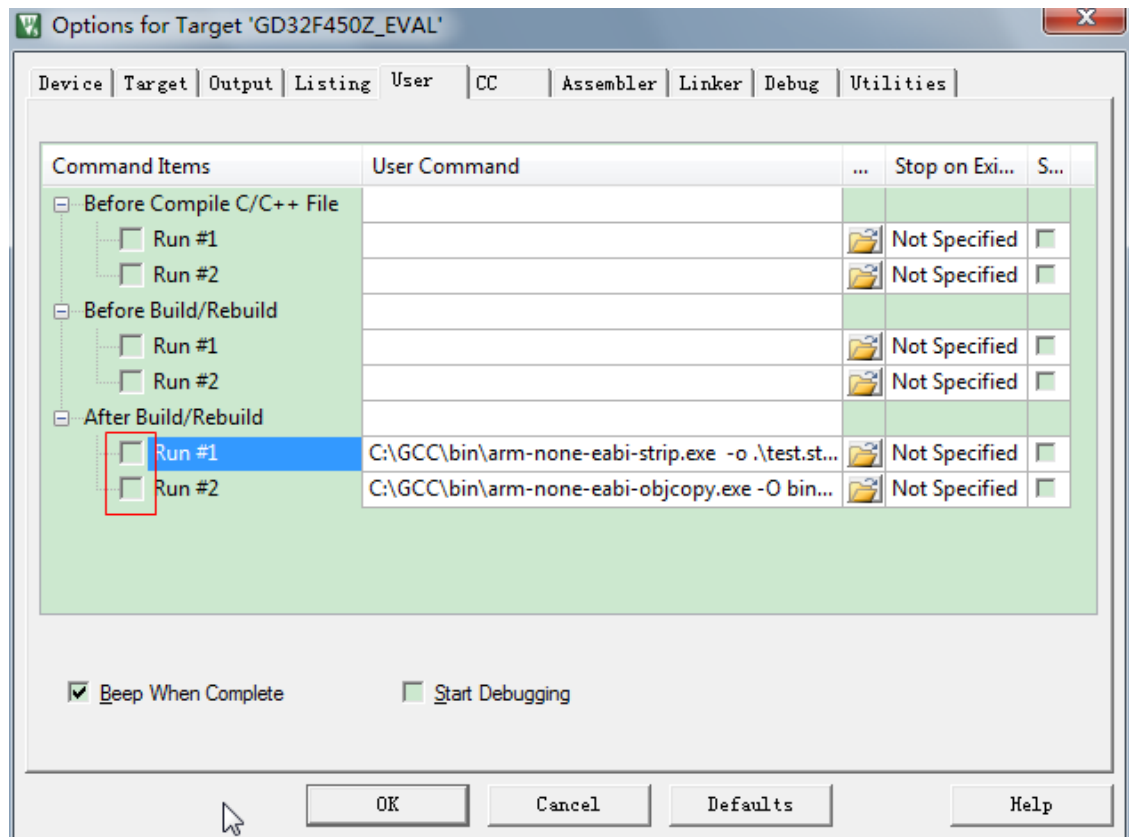


8. GCC 编译器可在如下地址下载：

<https://developer.arm.com/open-source/gnu-toolchain/gnu-rm/downloads>

<https://launchpad.net/gcc-arm-embedded/+download>

9. 若需要生成对应 BIN 文件，请勾选下图红色方框中的选项，并将路径修改为实际电脑上编译器工具的对应路径。



5.2 开发板配套 DEMO

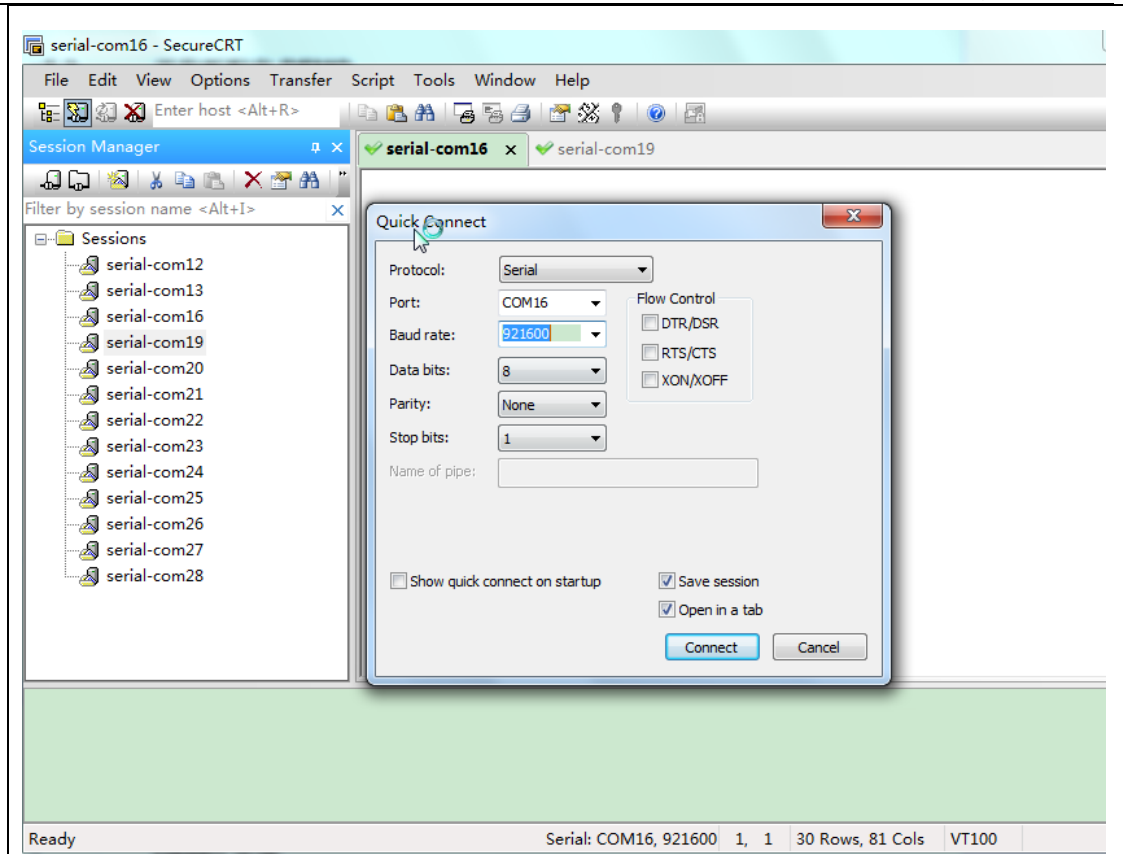
5.2.1 DEMO: AliOS_for_GD32F450Z

DEMO 介绍

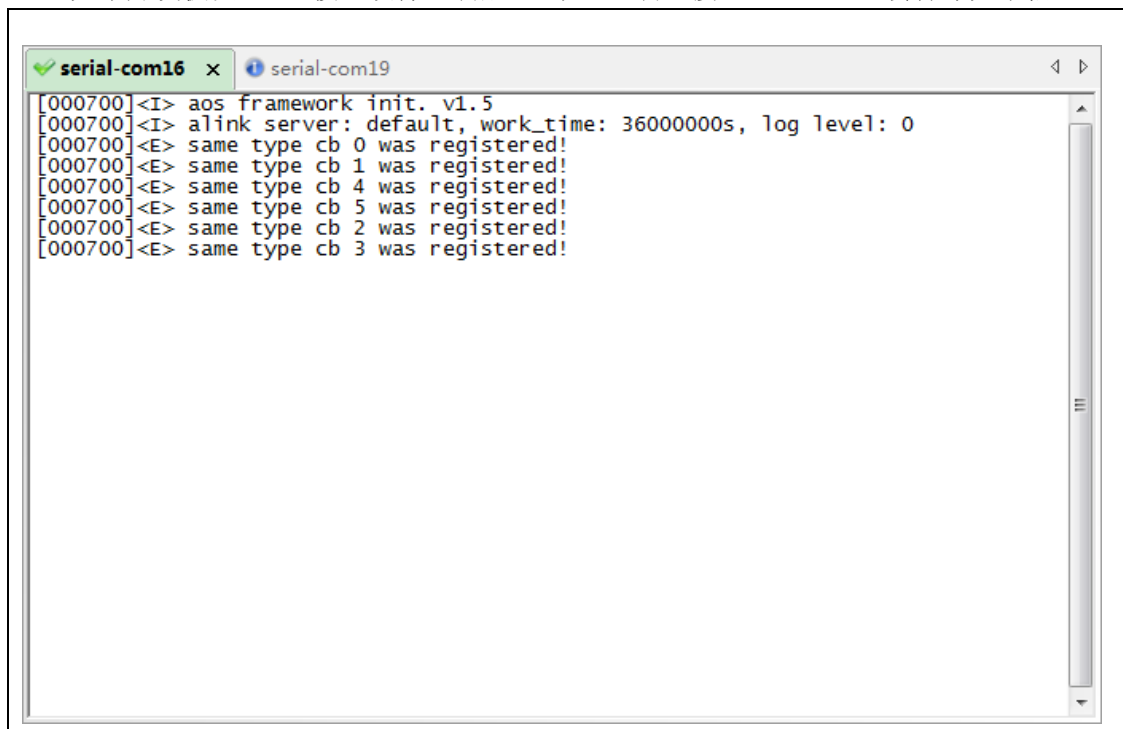
本 DEMO 通过串口实现了 PC 与开发板之间的交互指令，可以通过串口输入指令控制 WIFI 模组连接路由器并自动与阿里云完成对接，实现与云端数据的交互。

本 DEMO 演示流程如下：

1. 打开 SecureCRT 软件，根据 PC 串口实际使用的 COM 口进行配置，波特率为 921600



2. 如果开发板及 WIFI 模组硬件正确配置, 串口正确连接, SecureCRT 会打印如下信息:



3. 在输入框输入“netmgr connect 路由 SSID 路由 PASSWORD”, 如下图所示, 可以命令开发板进行连接路由的操作。

```

serial-com16 x serial-com19
[000700]<I> aos framework init. v1.5
[000700]<I> alink server: default, work_time: 36000000s, log level: 0
[000700]<E> same type cb 0 was registered!
[000700]<E> same type cb 1 was registered!
[000700]<E> same type cb 4 was registered!
[000700]<E> same type cb 5 was registered!
[000700]<E> same type cb 2 was registered!
[000700]<E> same type cb 3 was registered!
netmgr connect gigawifi gigawifi123

```

4. 一旦连接路由成功后，DEMO 会自动连接阿里云，并在连接成功后每三秒发送一包心跳包，保持连接。

```

[000700]<E> same type cb 5 was registered!
[000700]<E> same type cb 2 was registered!
[000700]<E> same type cb 3 was registered!
netmgr connect gigawifi gigawifi123

# [066080]<I> Got ip : 192.168.1.100, gw : 192.168.1.1, mask : 255.255.255.0
[066100]<I> Build Time: Thu Feb 9 01:44:23 Local time zone must be set--see zic
manual page 2017
[066100]<I> Git version: smarthome-bd18d6d
[066100]<I> accs initial
[066100]<I> wsf initial
[066100]<W> config init fail, reset...
[066100]<I> we will update local config
[066120]<I> ~~~~~dump device config~~~~~
[066120]<I> magic: Alink+
[066120]<I> version: 1
[066120]<I> log: 7
[066120]<I> uuid:
[066120]<I> token:
[066120]<I> alinkserver: online/alink.tcp.aliyun.com:443
[066120]<I> heartbeat: 3
[066130]<I> reqtimeout: 2
[066130]<I> dev_version:
[066130]<I> ota_version:
[066130]<I> crc: 35176
[066130]<I> ~~~~~
[066140]<I> ~~~~~dump device info~~~~~
[066140]<I> sn: FC0EE0E60050710E5
[066140]<I> cid: emw3006
[066140]<I> mac: FC:EE:E6:05:71:E5
[066140]<I> model: ALINKTEST_LIVING_LIGHT_ALINK_TEST
[066140]<I> os_version: aos
[066140]<I> alink_version: 1.0
[066140]<I> firmware_version: APP-1.0.0-20171214.1512
[066140]<I> key: 5gPF18G4GyFZ1fPwk20m
[066140]<I> secret: ngthgT1z65bX5LpviKIWNsDPhof2As9ChnoL9gqb
[066140]<I> debug_key: dpZZEpm9eBfqzK7yveLq
[066140]<I> debug_secret: THnfRRsU5vu6g6m9X6uFyAjuwflgz0iyGjdEnekm
[066150]<I> A[1.0|smarthome-bd18d6d|0000]OS[aos]T[.xxx]
[066150]<I> ~~~~~
[066150]<I> wsf_connect
[068150]<I> wsf initializing...

[068150]<I> wsf: reset connection
[068590]<I> add new tcp socket fd to poll list.

[068630]<I> save stats data to flash
[068640]<I> we will update local config
[069530]<V> we will start handshake work.

[069530]<V> handshake work started.

[069590]<V> alink cloud connected!
[069630]<I> we will update local config
[070200]<I> accs_handshake success
[074310]<I> send heartbeat
[077350]<I> send heartbeat

```

5. 之后输入 uuid，可以查询设备连入阿里云的 uuid 信息，在阿里云后台将该 uuid 与账

号信息绑定后，可以使用阿里提供的 APP-“ONEAPK”，在手机端与设备进行通讯。

```
[014090]<I> send heartbeat
[017130]<I> send heartbeat
[020180]<I> send heartbeat
[023220]<I> send heartbeat
[026260]<I> send heartbeat
[029300]<I> send heartbeat
[032340]<I> send heartbeat
[035380]<I> send heartbeat
[038420]<I> send heartbeat
[041460]<I> send heartbeat
[044490]<I> send heartbeat
[047530]<I> send heartbeat
[050560]<I> send heartbeat
[053600]<I> send heartbeat
[056640]<I> send heartbeat
[059680]<I> send heartbeat
[062720]<I> send heartbeat
[065760]<I> send heartbeat
[068800]<I> send heartbeat
[071830]<I> send heartbeat
[074870]<I> send heartbeat
[077910]<I> send heartbeat
[080950]<I> send heartbeat
[083990]<I> send heartbeat
[087030]<I> send heartbeat
[090070]<I> send heartbeat
uuid
uuid: C1D0A1F604637371E17EBA3A83B7048D
# [093110]<I> send heartbeat
```

DEMO 实现

1. DEMO 概述:

DEMO 使用阿里 AliOS 操作系统及其组件。关于该操作系统及组件的详细信息，请参考 <https://github.com/alibaba/AliOS-Things/wiki>

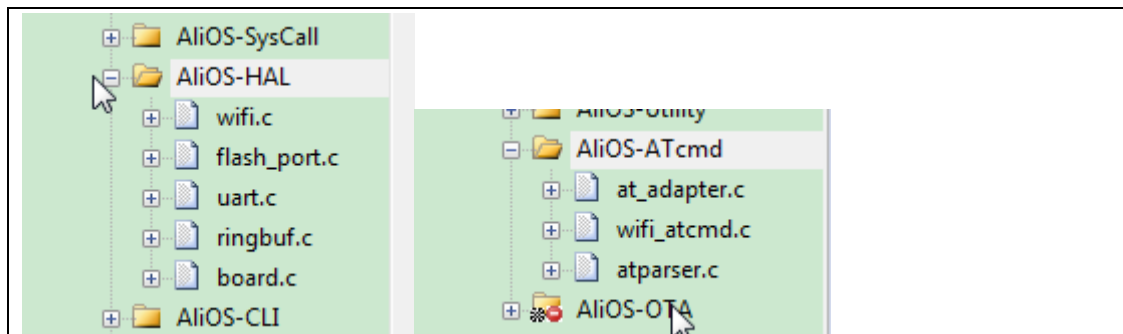
该操作系统高度组件化并且框架化，APP 代码入口为：

```
int application_start(int argc, char *argv[])
```

用户可以将自定义的代码放入该函数中实现。本 DEMO 使用 AliOS 源码 example 目录下的 alink_sample.c 实现阿里云的接入，其中定义的该函数如下图所示

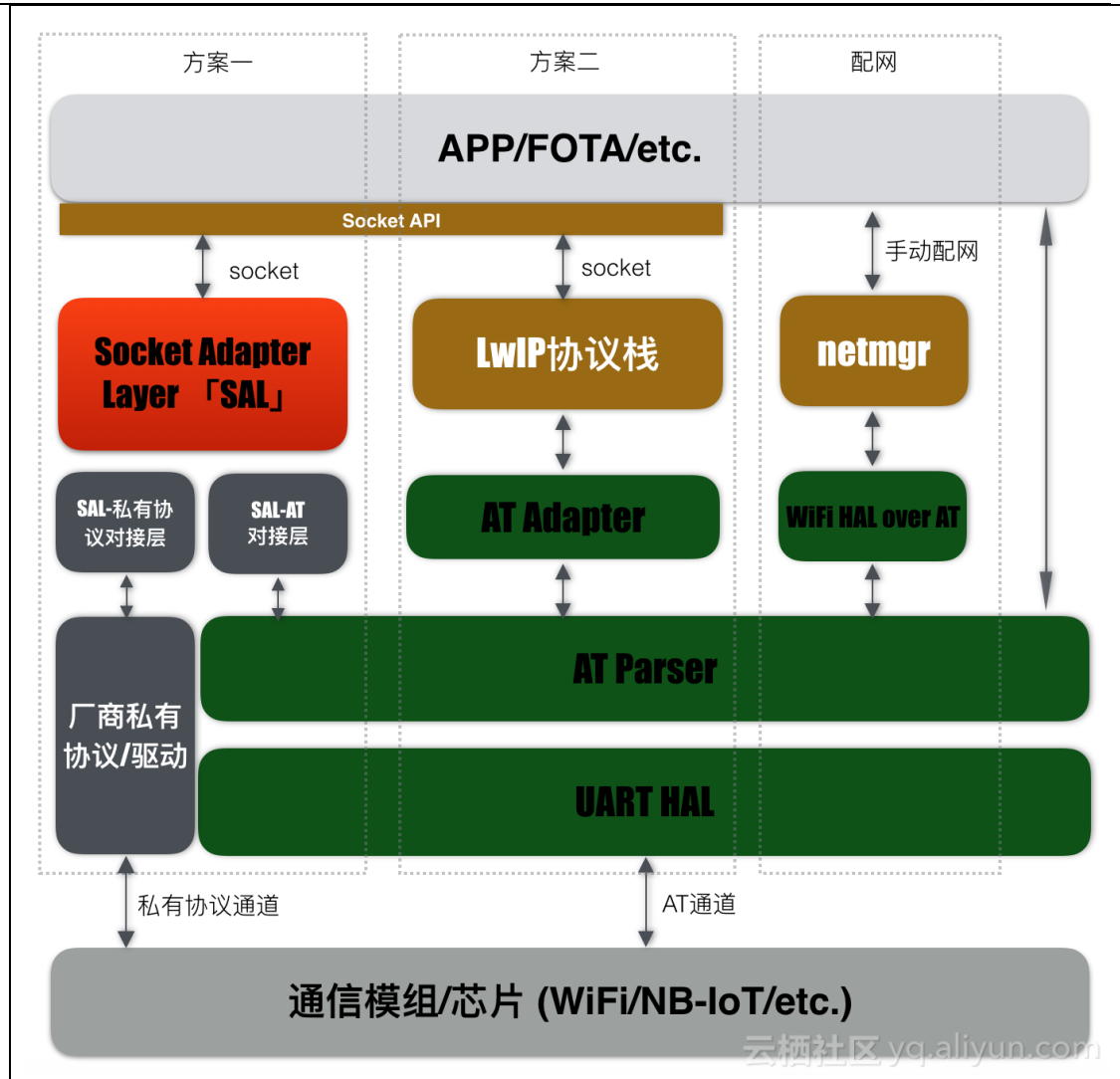
```
818 #endif
819
820 int application_start(int argc, char *argv[])
821 {
822     #ifdef AOS_AT_ADAPTER
823         uart_dev_t at_uart;
824         at_uart_configure(&at_uart);
825         at_init(&at_uart, AT_RECV_DELIMITER, AT_SEND_DELIMITER, 1000);
826         at.set_mode(ASYN);
827     #endif
828     parse_opt(argc, argv);
829
830     #ifndef AOS_AT_ADAPTER
831         if (mesh_mode == MESH_MASTER) {
832             #ifdef CONFIG_AOS_DDM
833                 ddm_run(argc, argv);
834             #endif
835             return 0;
836         }
837
838         #ifdef CONFIG_AOS_DDA
839             dda_enable(atoi(mesh_num));
840             dda_service_init();
841         #endif
842     #endif // #ifndef AOS_AT_ADAPTER
843
844     aos_cli_register_command(&suuidcmd);
845     alink_cloud_init();
846
847     if (mesh_mode == MESH_GATEWAY) {
848         aos_cli_register_command(&ncmd);
849         aos_cli_register_command(&modelcmd);
850         aos_cli_register_command(&resetcmd);
851     }
852     if (argv == SANDBOX)
```

MCU 与 WIFI 模组使用 AT 指令通过串口通讯，底层驱动代码位于 AliOS-HAL 目录下，AT 指令解析与处理位于 AliOS-ATcmd 目录下：



目前 AliOS 针对外扩 wifi 模组实现无线连接的场景提供了三种解决方案，如下图所示，本 DEMO 采用方案二实现。关于三种方案的介绍，请参考

<https://yq.aliyun.com/articles/292612?spm=5176.100239.blogcont313832.14.322f077dc7m5DM>



6 版本更新历史

Table 2 版本更新历史

Revision No.	Description	Date
1.0	初始发布版本	Jan. 19, 2018
1.1	增加对 gcc 编译器的介绍	Feb. 28, 2018