GD32 and STM32 Compatibility

1. Preparatory work

1.1. Data source

GD32MCU official website: www.gd32mcu.com, login is essential

GD32MCU unofficial cloud: https://pan.baidu.com/s/1mhQsNpu, no password extraction.

If client call for some other scripts and programs, please contract GigaDevice FAE.

GD32 developing resources is list as below:

1. Datasheet，also named specification, only English version
2. User Manual，English/Chinese version
3. Firmware and Example with each modules
4. Development board supporting resources, include routines and schematic circuit diagram
5. GD addon for MDK&IAR，
6. GD MCU hardware PCB resources

1.2. Integrated development environment

GD32MCU support KEIL MDK and IAR, above 4.74 for KEIL4, above 5.14 for KEIL5 and above 7.0 for IAR. It is convenient to for us to support client who using KEIL.

1.3. Emulator

Emulators which suit for GD32MCU are Jlink, ST-link (only for F1 series),Ulink and GD-link. Except GD-link, other emulators need to install driver.

1.4. Selecting GD device in IDE

It is necessary to install addon in KEIL and IAR for support GD device. There are three purposes in addon: 1, Select GD MCU device type; 2, Select GD flash download algorithm; 3, View registers in debug. All the addons could be downloaded from official website. Please install specific addon in given IDE.

If the client is not willing to install addon, please select ST device type for downloading and debugging

2. Differences and similarities between GD32 and STM32

2.1. Similarities

1. Peripheral Pin Definitions: for 10x and 4xx, the same device, the same pins; for 20x and 30x, same device is not exactly same.
2. Cortex M3 Core Version: STM32F103 is R1P1, STM32F205 is R2P1. GD32 core is R2P1, modify some core bugs. (Fig 1.1)
3. Core Internal Register and Peripheral Register: logical address is same, while several default values is different. Some peripheral timing design is different from STM32.
4. Library File: there is compatibility between GD32 and STM32.
5. Compile Tools: it is consistent with STM32 in IDE
6. Device Type Name Style: same

2.2. Hardware Difference

1. Voltage Range：GD32F10x： 2.6-3.6V STM32F10x：2.0-3.6V（external voltage）
2. GD32F： 1.2V （core voltage）STM32F10x\20x:1.8V（core voltage） STM32F4xx： 1.2V（core voltage）
3. BOOT0 Pin：when program start up from flash, BOOT0 pin of STM32 is in floating, while，for GD32, it is must be pull down.
4. ESD Parameter：Chip level ESD degree of GD32 is higher than STM32. For example, 103 series, STM32: The human body mode is 2KV and air mode is 500V, GD32: The human body mode is 4KV and air mode is 10KV.
5. JTAG/SWD Pin: according to the difference of download port Electrical characteristics and impedance matching between STM32 and GD32, if connect line between emulator and board is too long, or download speed is configured too high, it would bring download error. It is recommend to connect 10K pull up resistance to SWDIO and 10K pull down resistance to SWCLK, so as to avoid download error.

2.3. Difference of internal structure

1. Start-up Time: according to chip structure design difference, start-up time of

GD32 is longer than STM32.Such as GD32F10x, start-up time equals to hundreds ms. In application, it is necessary to overtime the timeout of star-up, and take care that longer start-up timer affect wakeup in low power consumption.

2. Frequency Clock: GD32F10x is up to 108MHz, STM32F10x is up to 72MHz; GD32F1x0 is up to 72MHz, STM32F0xx is up to 48MHz; GD32F4xx is up to 200MHz, STM32F4xx is up to 180MHz.
3. Flash Erase Time: if erase 1KB per page, GD32 is 60ms, while, STM32 is 30ms
4. Flash Size: GD32 is up to 3MB, while, STM32 is up to 2MB.
5. SRAM Size: the SRAM size which GD32F2 support maximally is bigger than STM32F2, so is F4 series.
6. EXMC: if the total number of pins come up to 100, there is EXMC in GD32, while, for STM32, flash must come up to 256KB at same time, so EXMC exist.

2.4. Power consumption difference

1. Take GD32F103 with flash below 256KB as an example
2. Sleep： GD32F： 12.4mA STM32F： 7.5mA
3. Deep Sleep： GD32F： 1.4mA STM32F： 24uA
4. Standby: GD32F: 10.5uA STM32F: 3.4uA
5. Running power consumption：GD32F: 32.4mA/72M STM32F: 52mA/72M

2.5. Difference of Internal Flash

1. ISP: erase timer is different
2. IAP: GD32 program in word or half word, STM32 program only in word
3. storage life： up to 100 thousand times for programming and erasing, up to 20 years for data storage。
4. Encryption technology characteristic: for GD32, except for read production and ID encryption, there is hardware encryption that storage data is continuous in logical, while, discontinuous in physical.

3. Compatibility Detailed Analysis

3.1. System

1. Issue：crystal oscillator start up timer modify

reason	Start-up time, there is difference between GD32 and STM32, to reset MCU accurately, the start-up time should be optimized.
solution	change #define HSE_STARTUP_TIMEOUT ((uint16_t)0x0500) to #define HSE_STARTUP_TIMEOUT ((uint16_t)0xFFFF) note：start-up time macro location： 1, lib 3.0 later, the macro is located in stm32f10x.h 2, lib 3.0 previous, the macro is located in stm32f10x_rcc.
type	GD32F10x、GD32F1x0
tips	If client configure clock without HSE, it is no need to modify it.

2. Issue：distinguish GD32 and STM32

reason	Actually, some clients hope the code suit for both GD32 and STM32, so it is necessary to distinguish them.

solution	Run the function below
	uint16_t Get_JTAG_ID( )
	{
	    if(*(uint8_t *)(0xE00FFFE8) & 0x08)
	    {
	        return ((*(uint8_t *)(0xE00FFFD0) & 0x0F) << 8)
	|
	        ((*(uint8_t *)(0xE00FFFE4) & 0xFF) >> 3) |
	        ((*(uint8_t *)(0xE00FFFE8) & 0x07) << 5) + 1 ;
	    }
	    return 0;
	}
	If return 0x041, the chip is ST, while, if return 0x7A3 ,the chip is GD 。

type	GD32F10x
tips	

3. Issue: attention which is brought by GD32F10x flash Instruction fetch zero wait

reason	At same frequency, GD32 increase code execution speed
solution	If there is “for loops” or “while loops” in code for precise timing, the expectant time will be shorten due to higher code execution

	speed. If the code select timer to calculate time, the effect above will lack.
type	GD32F10x、GD32F1x0
tips	For F2 and F4 series, code execution speed is alike at same frequency. If main frequency do not exceed 24MHz, the performance difference between GD and ST is tiny.

4. Issue：Transplant code form STM32, GD32 power on failure in same hardware.

reason	If there are some high power modules on client board, such as WIFI module, GSM module and GPS module, and 3.3V power input output capacitance is improper, they will lead GD MCU start up failure. 3.3V supply both MCU and high power modules, there is larger current at the power on moment, so as to generate a voltage falling at LDO output capacitance. If 3.3V drop to range between 2.0V and 2.6V, ST will still work in normal, while GD MCU will be abnormal.
solution	If there is high power modules on client board, it is suggested that adjust LDO output capacitance parameter, prefer to use tantalum capacitance, increase capacitance value properly, these adjusting would decrease ESR value, so as to decrease voltage falling range.
type	GD32 all series
tips	Before power supply in normal, if there is residual voltage in VCC, and the voltage is bigger than power on reset threshold voltage, less than 2.6V which GD32 MCU could work in normal, it will leads a start-up failure.

5. Issue：BOOT0 pin pull down

reason	when program start up from flash, BOOT0 pin of STM32 is in floating, while，for GD32, it is must be pull down.
solution	The BOOT0 pin must be pulled down with 10K resistance
type	GD32 all series
tips	

6. Issue：appear“Flash Download Failed”, unable to download

reason	According to the difference of download port Electrical characteristics and impedance matching between STM32 and GD32, if connect line between emulator and board is too long (exceed 20cm), no pull up or pull down resistance connect to SW port or download speed is configured too high, it would bring download error.

solution	It is recommend to connect 10K pull up resistance to SWDIO and 10K pull down resistance to SWCLK, shorten the connect line between emulator and board and download speed is configured to 4MHz below, so as to increase anti-interference and avoid download error.
type	GD32 all series
tips	

7. Issue：frequently software reset leads code exception with probability

reason	Because there is cache ,which is treated as execution area, corresponds to flash, frequently software reset will leads code exception in execution area with probability
solution	1、 Avoid frequently software reset, please use hardware reset or power reset as much as possible; 2、 Advice client to locate the software reset in SRAM. On the reset moment, flash code area of MCU, which is executing code, is easiest to be interfered. If software reset is put in SRAM, it avoid to change flash content accidentally. 3、 Process CRC check at the beginning of application. If check value occurs, MCU will enter into standby mode, wait for next software reset, so as to reset wakeup. Refer details to AN006 Programming Notes for Reset
type	GD CortexM3 series
tips	Hardware reset without this question

8. Issue：in GD32F450, when system clock switch from high frequency clock to low frequency clock (HSI), it will leads HIS,HSE,PLL clock disappear, program will run out of control.

reason	When system clock is high frequency, load current is big, switch to low frequency clock fast, leads internal LDO is unable to follow fast change, so VCORE falling disable part circuits
solution	Before switch clock, divide HCLK to 2 or 4 multiple, then switch clock
type	GD32F450 series
tips	Other series without this question

9. Issue：Serial port rate exception for GD32F101 who use ST library

reason	Function RCC_GetClocksFreq() in ST library, which is judge if HSE is divide frequency from CFGR2 register of STM32F100, while GD32F101 lack this register, leads to unable to get correct system frequency.
solution	LSB of CFGR2 is as same as bit 17 of CFGR register, so checking bit 17 of CFGR in RCC_GetClocksFreq() is able to get correct frequency.
type	GD32F101 series
tips	

3.2. Internal Flash

1. Issue：chip cannot configured to be read protection

reason	There is difference between GD32 and STM32 in configuring read protection
solution	After wrote KEY register sequence, it is necessary to read this bit for checking unlock is successful. So, insert this statements While(! (FMC_CTL & 0x200)); Or insert two NOP __NOP(); __NOP(); In ST library FLASH_Status FLASH_EraseOptionBytes(void) FLASH_Status FLASH_ProgramOptionByteData(uint32_t Address, uint8_t Data) FLASH_Status FLASH_EnableWriteProtection(uint32_t FLASH_Pages) FLASH_Status FLASH_ReadOutProtection(FunctionalState NewState) Should modify
type	GD32F10x series
tips	Other series without this question

2. Issue：IAP program

reason	Erasing and programing time of GD32 is tiny longer than STM32, it is advised to modify the timeout for erasing and programing
solution	change #define EraseTimeout ((uint32_t)0x000B0000) #define ProgramTimeout ((uint32_t)0x00002000) To be #define EraseTimeout ((uint32_t)0x000FFFFF) #define ProgramTimeout ((uint32_t)0x0000FFFF)
type	GD32 all series
tips	

3. Issue：The same code execution speed of GD32 is lower than STM32

reason	GD32 flash consist of two areas: code area and data area. There is no difference in erasing and programing, but make difference in reading in two areas. Code area is zero wait for fetching instruction. Data area have a large delay in executing code. Thus,

	it is not recommend to run real time code
solution	Scatterloading, locate the real time code in flash code area; Upgrade to be MCU with bigger code area flash, such as upgrade 1x0 to 3x0
type	10x/20x/30x/4xx series with flash size exceed 256KB, 1x0 with flash exceed 64KB
tips	

3.3. ISP Programmer Software

1. Issue: Incompatible with other manufacturer ISP version before 2012 years

reason	GD32 chip core version is newer
solution	Advise to download newest version MCUISP from www.mcuisp.com； Besides, download GigaDevice ISP from website: http://bbs.21ic.com/gd32. If develop homemade ISP software or off-line programmer, to realize compatibility between GD and ST, pleas modify parameters below. 1、Page erasing timeout add up to 300ms, whole chip erasing timeout add up to 3s 2、Word programming timeout add up to 2ms, page programming timeout add up to 300ms
type	GD32F10x series
tips	

3.4. FWDGT

1. Issue: wakeup abnormal when enter in standby or stop mode on the moment of FWDGT reload

reason	In the course between executing reload instruction and clearing reload signal, if enter in standby or stop mode at the moment, it is unable to response subsequent reload instruction.
solution	It is necessary to make sure that there is three LSI clock between executing reload instruction and entering in standby or stop mode in software.
type	GD32F10x、GD32F130/150 series
tips	Other series without this question

3.5. DMA

1. Issue: check if retransmit when disable DMA in the course of DMA transmitting

reason	DMA design difference between GD32 and STM32

solution	Software avoiding based on DMA difference between GD32 and STM32
type	GD32F10x all series
tips	For GD32F10x with flash below 256KB, it leads data transmit leakage in this case. For other GD32F10x series, it will happen retransmitting.

3.6. CAN

1. Issue: After CAN entered bus-off, it will not recover transmit in probability with configure ABOM

reason	CAN design difference between GD32 and STM32
solution	Open bus-off interrupt, initialize CAN in interrupt handler.
type	GD32 all series
tips	

2. Issue: CAN receive abnormal

reason	Repeat call CAN_FIFORelease function, so as to lose packet.
solution	CAN FIFO could be released in receive function, please do not repeat call CAN_FIFORelease function.
type	GD32 all series
tips	

3.7. I/O Port

1. Issue: can't open IO external interrupt After use IMR register

reason	During the period of the interrupt is disabled in medium-density products, if the external pin has a level change, the interrupt service routine will be responded and executed immediately after the interrupt is enabled using the EXTI_INTEN Theoretically, the level change of the pin before opening the interrupt will not affect the interrupt response after opening the interrupt

solution	Open or close interrupts by disabling the rising edge or falling edge detection registers instead of using the EXTI_ INTEN register
	EXTI_FTEN &= ~EXTI_Line3; // close edge detection, meaning close interrupt. Falling edge corresponds to FTEN register, rising edge corresponds to RTEN register
	EXTI->PD = EXTI_Line3;
	EXTI->FTEN |= EXTI_Line3;

type	GD32F10x series with flash below 256KB
tips	

2. Issue: PAB is unable to use in low power consumption

reason	PA8 is internally connected to ground in low power consumption
solution	Configure PA8 as in floating in low power consumption
type	
tips	

3. Issue: when IO is configured as pull up input, GD32 IO voltage level is 2.6V, STM is 3.3V

reason	GPIO port pull up and pull down design difference between GD32 and STM32
solution	Just signal level difference, not affect application.
type	GD32F10x series
tips	Other series without this question

4. Issue: two modules remap the same IO port output

reason	IO port output is assigned on the basis of Module priority. Thus, even if IO ports, which have been assigned to higher priority module, are idle, other modules remain cannot use.
solution	TIMER1_CH2 and USART1_RTS share PA1, after enable USART1 clock, PA1 only could be used by USART1_RTS. Even though USART1 do not use Hardware flow control, TIMER1_CH2 still cannot use PA1 USART0_RTS and CAN0_TX share PA12, when USART and CAN enable its clock, the priority of USART is higher than CAN. If PA12 is configured as output in both USART and CAN, PA12 is assigned to USART, not for CAN at any time. If one configure in, the other one configure out, or both configure in, nothing will be abnormal.
type	GD32F103/101 series below 256KB
tips	Other series without this question

5. Issue: GPIO continuous flip without delay, appear output level error.

reason	GPIO continuous set OCTL, set BC, reset OCTL and set OCTL again without delay. The second setting OCTL will be covered by previous setting BC and resetting OCTL.
solution	Add one cycle delay between setting BC, resetting OCTL and setting OCTL again.
type	GD32F130\150
tips	Other series without this question

6. Issue: PB1 configuration affect PB2 function

reason	If PB1 is configured as IPU、IPD、AF_PP or AF_OD，PB2 is out of control

solution	For using PB2 in normal, PB1 must be work in Out_PP, Analog, Input_floating mode
type	GD32F103/101 series below 256KB
tips	Other series without this question

7. Issue: IO cannot turn to high impedance state immediately in standby mode

reason	When testing the Standby mode, IO maintains its original state after downloading the program for the first time, and cannot immediately enter the high impedance state.
solution	The chip must be powered on again before entering the high impedance state.
type	GD32F10x\GD32F1x0 series
tips	Other series without this question

3.8. TIMER

1. Issue: timer upward pulse counting mode is abnormal

reason	Timer usage difference
solution	In pulse counting mode, the load value must be set larger than the expected value, or the CNT do not count. If the reload value in the ST is not set (initially 0), CNT can normally count. On GD, if the reload value is not set, even coming one pulse will cause the reset of all the registers and CNT cannot properly count because the reload value is zero.
type	GD32F10x above 256KB
tips	Other series without this question

2. Issue: unable to clear CHxIF by read TIMERx_CHxCV register when channel is configured in input capture mode

reason	GD32 do not use this method at present
solution	Software clearing interrupt flag
type	GD32F10x series
tips	Other series without this question

3.9. USART

1. Issue: stop bit length of USART

reason	There is idle bit between byte and byte. (Fig 1.2)
solution	There is no effect on common communication. If communication need special demand, please simulate USART in software.
type	GD32F10x series
tips	Other series without this question

2. Issue: ISO7816 apply to GD MCU

reason	It is able to implement application in hardware without pass EMV identification
solution	GD32F10x can accomplish ISO7816 applications, but the hardware signal cannot pass the EMV certification, because 0.5/1.5 STOP bit cannot be sent and there are also some differences in NAK signal. If you want to use the GD32F10x series to be certified by the EMV standard, you need to use the IO port to simulate data sending and answering signals.
type	GD32F10x series
tips	Other series without this question

3.10. I2C

1. Issue: I2C unable to work in normal

reason	Flag bit difference between GD32 and STM32
solution	change macro address #define I2C_EVENT_SLAVE_TRANSMITTER_ADDRESS_MATCHED ((uint32_t)0x00060002) #define I2C_EVENT_MASTER_TRANSMITTER_MODE_SELECTED ((uint32_t)0x00070002)
type	GD32F10x
tips	

2. Issue: I2C unable to clear RBNE flag bit

reason	It is necessary to read STR register and then read DR register immediately for clearing RBNE flag. If there is a long delay between read STR and read DR, RBNE is able to clear. It is common in ST chip
solution	turn off global interrupt before read STR and DR
type	GD32 all series
tips	

3.11. ADC

1. Issue: ADC sample value is 0 always due to IO mode error

reason	It is necessary to configure ADC sample pin as analog input mode, not as in floating default. If the pin in default mode, ST could work in normal, while, GD cannot.
solution	Configure IO as analog input mode
type	GD32 all series

tips	

2. Issue: ADC sample value is 0 always due to sample frequency

reason	If ADC clock do not divide system frequency to maximum allowable frequency (14M for F1), GD cannot work in normal, while, ST can do it.
solution	Configure ADC clock to maximum allowable frequency
type	GD32 all series
tips	

3. Issue: ADC sample value is 0 always due to delay after ADC enable

reason	ADC design difference between GD32 and STM32
solution	Insert 20us delay after ADC enable
type	GD32F103/101 with below 256KB
tips	Other series without this question

4. Issue: ADC2 insert group value is 0 when configure ADC1 and ADC2 in synchronous mode

reason	When configure ADC1 and ADC2 in synchronous mode, ADC2 followed ADC1 synchronically trigger, trigger mode of insert group should be configured as software trigger. Otherwise, STM32 is able to work in normal, while, ADC2 of GD32 insert group sample value is 0 with probability
solution	Configure ADC2 trigger mode as software trigger in ADC2 initialization
type	GD32 all series
tips	

5. Issue: ADC, which is triggered by external pulse signal, is unable to convert with probability

reason	High level hold time of GD32 ADC trigger signal should above one ADC cycle, not suit for STM32
solution	Change to other trigger mode(such as PWM trigger), it is necessary to make sure that high level hold time of GD32 ADC trigger signal should above one ADC cycle
type	GD32F103/101 with below 256KB
tips	Other series without this question

6. Issue: ADC accuracy of GD32 is tiny lower than STM32

reason	ADC accuracy of GD32 is tiny lower than STM32
solution	No solution so far, if higher accuracy is needed, please use

	GD32F3 and GD32F4
type	GD32F10x、GD32F1x0、GD32F20x。
tips	Other series without this question

7. Issue: sample value abnormal caused by non-5V-tolerate pin load voltage above 3.6V

reason	If non-5V-tolerate pin load voltage above 3.6V, internal circuit of ADC channel would be abnormal, and sample value without accuracy.
solution	If ADC is necessary, avoid non-5V-tolerate pin load voltage above 3.6V
type	GD32F10x、GD32F130\150
tips	Other series without this question

8. Issue: sample through resistor divider, ADC sample value is tiny smaller.

reason	There is difference between GD32 and STM32 in ADC input impedance, at same sample cycle, input impedance of GD is smaller than ST. If sample through resistor divider, sample value is smaller than actual value. (Fig 1.3)
solution	Increase sample cycle, decrease sample voltage resistor in hardware.
type	GD32 all series
tips	

9. Issue: EOC flag bit cannot be clear automatically in hardware.

reason	When converting of ADC regular group or insert group is completed, automatically set in hardware and clear by write 0
solution	clear by write 0
type	GD32F103/101 with below 256KB
tips	Other series without this question

10. Issue: ADC data dislocation if reconfigure ADC and DMA when ADC is working.

reason	After disable ADC and DMA, DMA request of ADC cannot clear. Thus, when reconfigure ADC and DMA, there is possible remain DMA request which is not answered, after enable DMA, answer the remaining DMA request immediately, so as to leads ADC data dislocation.
solution	In DMA TC interrupt handler, disable DMA request bit of ADC or bit ON of ADC, so as to make sure the remaining DMA request will not send out. Reset ADC bit of RCC, clear DMA request.
type	GD32F10x series

tips	Other series without this question

11. Issue: The first channel convert value of insert group is 0x800 when ADC use regular group and insert group at the same time.

reason	When insert group cycle is set as 1.5 or 7.5, this issue appear with probability
solution	Configure insert group cycle as 13.5.
type	GD32F103/101 with above 128KB，GD32F130\150。
tips	Other series without this question

12. Issue: there is no sample time register corresponding VBAT channel.

reason	Sampling time of CH0 also apply to VBAT channel actually
solution	Configure sampling time of CH0
type	GD32F130\150。
tips	Other series without this question

13. Issue: continually turning on and off VBAT channel affect ADC data sample

reason	Turning off VBAT channel would trigger ADC convert, thereby, leads data sampling is inconsistent with channel
solution	Avoid disable channel without enable channel previously
type	GD32F130\150。
tips	Other series without this question

14. Issue: 14M crystal oscillation deadlock in reset wait calibration register.

reason	For GD MCU, enable 14M clock and then select 14M clock as ADC clock. For ST MCU, if select 14M clock as ADC clock, automatically enable internal 14M clock.
solution	enable 14M clock and then select 14M clock as ADC clock
type	GD32F130\150
tips	

3.12. ETH

1. Issue: if MCU is busy, process descriptor speed is low, at the same time, frequently receive frame, possibly occur Ethernet abnormal. Even if the network is idle, remain not recover.

reason	Process receive FIFO overflow difference between GD and ST
solution	Open Flush ETH_InitStructure.ETH_DMA_FlushReceivedFrame = ETH_FLUSHRECEIVEDFRAME_ENABLE; Open ROS（Receive Overflow）interrupt, in its interrupt handler, reset ETH module and initialize parameters. in receive interrupt handler, check valid data, if RDES resource include irregularity data, drop them and release RDES resource uint32_t ETH_GetRxPktSize(void)

	{
	uint32_t size = 0;
	
	if((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_BUSY) != (uint32_t)RESET)
	{
	return 0;
	}
	
	if(((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_ERRS) != (uint32_t)RESET) ||
	((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_LDES) == (uint32_t)RESET) ||
	((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_FDES) == (uint32_t)RESET))
	{
	ETH_DropRxPkt();
	
	return 0;
	}
	
	if(((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_BUSY) == (uint32_t)RESET) &&
	((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_ERRS) == (uint32_t)RESET) &&
	((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_LDES) != (uint32_t)RESET) &&
	((DMACurrentRxDesc->Status &
	ETH_DMARXDESC_FDES) != (uint32_t)RESET))
	{
	/* Get the size of the received data including CRC */
	size =
	ETH_GetDMARxDescFrameLength(DMACurrentRxDesc);
	}
	
	/* Return Packet size */
	return size;
	}

type	GD32F107、GD32F207
tips	

3.13. SPI

1. Issue: Data send and receive exception in SPI master mode

reason	In the SPI master mode, the internal logic is subject to interference of changes in the SCK pin level, which will cause errors in the internal logic and output waveforms. These interferences include configuring SPI by software during the reconfiguration of GPIO, ESD electrostatic etc.
solution	The solution of configuring SPI by software during the reconfiguration of GPIO is to modify the configuration sequence to avoid interferences. The solution of ESD interference is to turn off SPI after one byte being sent each time and then enable SPI again.
type	GD32F10x all series
tips	Other series without this question

2. Issue: Data send and receive exception in SPI slave mode

reason	In SPI slave mode, if the input CLK is not a full multiple of 8 or 16 (determined by the configuration of the bit width) and the chip select is closed (set the SWNSS to 1 in software control mode or pull the NSS pin high in hardware control mode), GD32F10x SPI will not clear the count. After the chip select is enabled, and then wait for the corresponding amount of CLK, SPI will send and receive new data.
solution	Resetting SPI module in software
type	GD32F10x all series
tips	Other series without this question

3. Issue: SPI data receive error

reason	SPI timing design difference between GD32 and STM32
solution	Check RBNE flag to judge transmit complete
type	GD32F10x all series、GD32F130\150
tips	Other series without this question

4. Issue: SPI cannot transmit data after switching the clock

reason	SPI design difference between GD32 and STM32
solution	Close SPI before switching the SPI clock and enable SPI after the switching is completed
type	GD32 all series
tips	

5. Issue: SPI can send data and cannot receive data

reason	Receive port is configure as in floating mode in normal, while, receiving port of ST , which is configure as push pull output, could work in normal.
solution	Check the mode of MISO pin is floating input or not.
type	GD32 all series
tips	

3.14. USBOTG

1. Issue: computer cannot detect MCU as device

reason	Library update
solution	Some clients have tested that in ST old OTG lib V3.2.1 below, it is necessary to put OTGD_FS_CoreReset() before configure GCCFG, or else corereset could clear GCCFG configure in OTGD_FS_CoreInit(). In new OTG lib V2.1.0, code without this question.
type	GD32F105\107
tips	

2. Issue: hardware design as device

reason	As device, DP pull up in connected to PA9
solution	In hardware design, PA9 is connected to VBUS or VCC, no any remapping function
type	GD32F105\107
tips	

4. Attention items

4.1. Configure 108MHz

Combining PLLMF [3:0] and PLLMF [4] in RCC_CFG0, five bits ensure PLL clock multiplier factor, PLL output frequency is exceed Highest frequency (108MHz).

4.2. USB

According to lowest operating frequency，APB1 clock is bigger than 24MHz，such as HCLK is 48MHz, max division factor is 2, 48/2 = 24MHz，to make sure USB work in normal.

4.3. Ethernet hardware design attention

1、Ethernet and USART1 share PA2

PA2	ETH_RMII_MDIO /ETH_MII_MDIO/USART1_TX

ETH_RMII_MDIO, ETH_MII_MDIO and USART1_TX share PA2, if USART1 clock enable, PA2 is preferentially assigned to USART1. If use USART1 and Ethernet at same time, must configure USART1 for other pin remapping, or else Ethernet is unable to work in normal.

2、Ethernet and I2C1 share PB10/PB11

PB10	I2C1_SCL/USART2_TX/ETH_MII_RX_ER
PB11	I2C1_SDA/USART2_RX/ETH_MII_TX_EN/ETH_RMII_TX_EN

Ethernet and I2C1 share PB10/PB11, if I2C1 clock enable, PB10/PB11 are preferentially assigned to I2C1. If use I2C1 and Ethernet at same time, must configure I2C1 for other pin remapping, or else Ethernet is unable to work in normal.

3、Ethernet and TIM1 share PB12/PB13

PB12	TIMER0_BKIN/ETH_MII_TXD0/ETH_RMII_TXD0
PB13	TIMER0_CH1N/ETH_MII_TXD1/ETH_RMII_TXD1

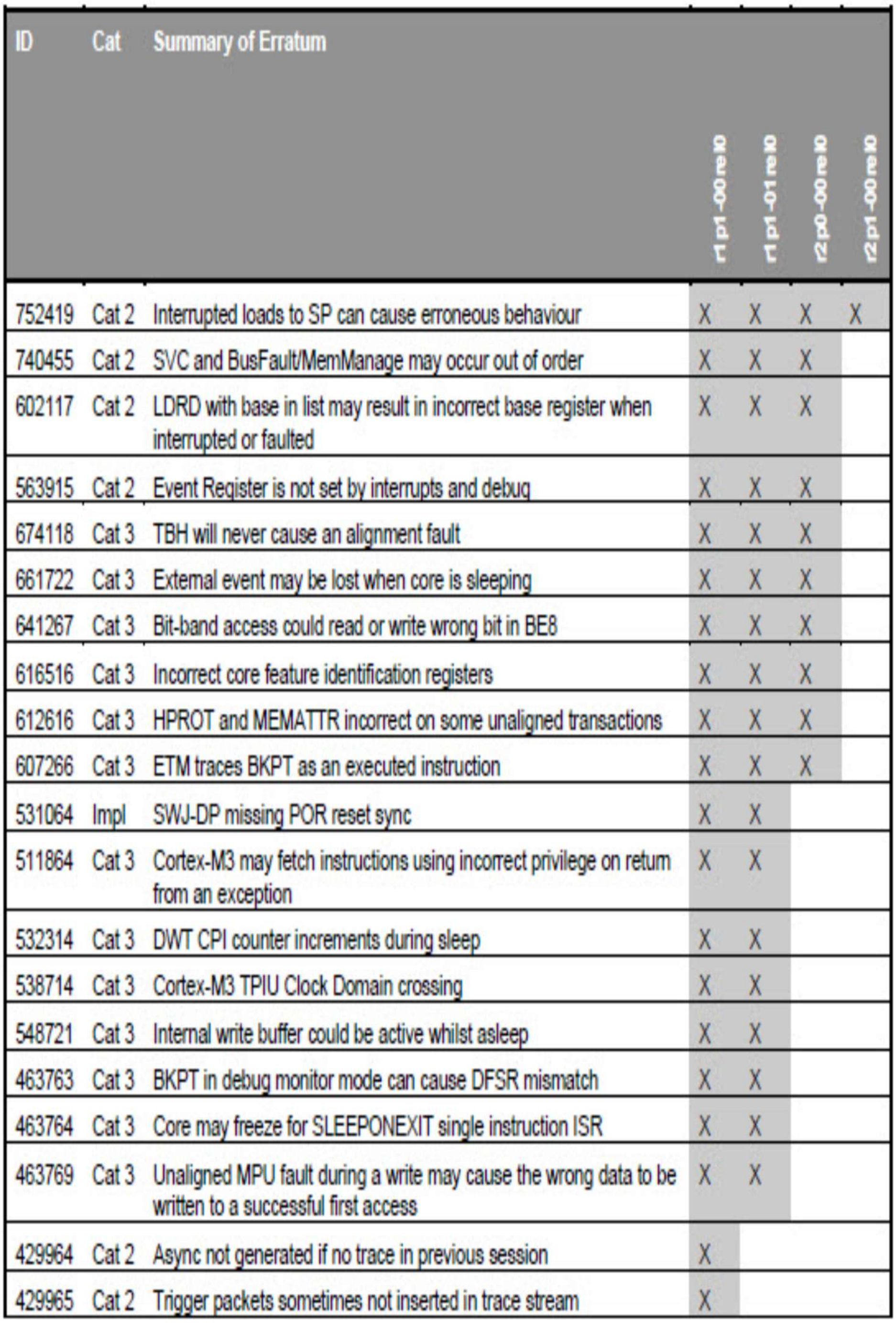
Ethernet and TIMER0 share PB12/PB13, if TIMER0 clock enable, PB12/PB13 are preferentially assigned to TIMER0. If use TIMER0 and Ethernet at same time, must configure TIMER0 for other pin remapping, or else Ethernet is unable to work in normal.

4.4. Power Consumption

After power on MCU, all IOs is default in floating, voltage level of IO port is affected easily, thereby, affect MCU consumption. It is recommended that configure all IO ports as analog input mode. Even if there is some port without extract, such as GD32103C8, do not extract GPIOE, all pins of GPIOE is remain to configure as analog input.

5. Reference pictures

5.1. Core



ID	Cat	Summary of Erratum	r1p1-00rel0	r1p1-01rel0	r2p0-00rel0	r2p1-00rel0
752419	Cat 2	Interrupted loads to SP can cause erroneous behaviour	X	X	X	X
740455	Cat 2	SVC and BusFault/MemManage may occur out of order	X	X	X	
602117	Cat 2	LDRD with base in list may result in incorrect base register when interrupted or faulted	X	X	X	
563915	Cat 2	Event Register is not set by interrupts and debug	X	X	X	
674118	Cat 3	TBH will never cause an alignment fault	X	X	X	
661722	Cat 3	External event may be lost when core is sleeping	X	X	X	
641267	Cat 3	Bit-band access could read or write wrong bit in BE8	X	X	X	
616516	Cat 3	Incorrect core feature identification registers	X	X	X	
612616	Cat 3	HPROT and MEMATTR incorrect on some unaligned transactions	X	X	X	
607266	Cat 3	ETM traces BKPT as an executed instruction	X	X	X	
531064	Impl	SWJ-DP missing POR reset sync	X	X		
511864	Cat 3	Cortex-M3 may fetch instructions using incorrect privilege on return from an exception	X	X		
532314	Cat 3	DWT CPI counter increments during sleep	X	X		
538714	Cat 3	Cortex-M3 TPIU Clock Domain crossing	X	X		
548721	Cat 3	Internal write buffer could be active whilst asleep	X	X		
463763	Cat 3	BKPT in debug monitor mode can cause DFSR mismatch	X	X		
463764	Cat 3	Core may freeze for SLEEPONEXIT single instruction ISR	X	X		
463769	Cat 3	Unaligned MPU fault during a write may cause the wrong data to be written to a successful first access	X	X		
429964	Cat 2	Async not generated if no trace in previous session	X			
429965	Cat 2	Trigger packets sometimes not inserted in trace stream	X			

Fig 1.1

5.2. USART

There is a bit of idle between byte and byte, when USART send data continuously.

As is shown below.

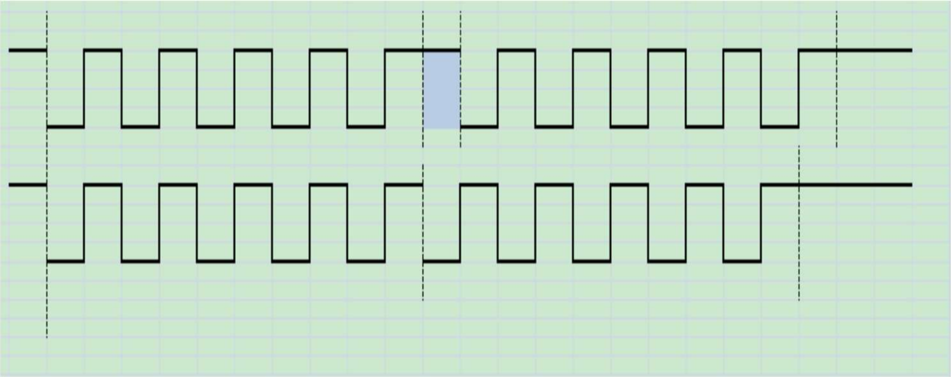
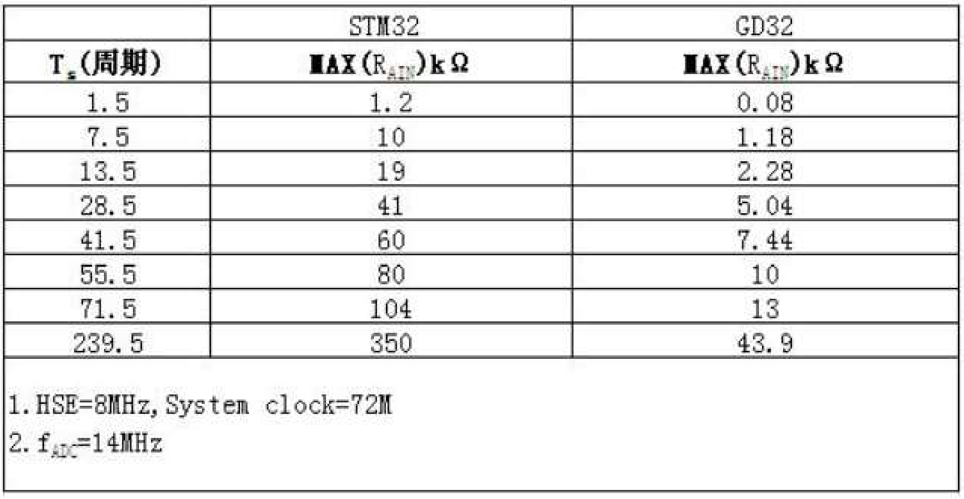


Fig. 1.2

5.3. ADC



	STM32	GD32
T_s(周期)	MAX(R_{AIN})kΩ	MAX(R_{AIN})kΩ
1.5	1.2	0.08
7.5	10	1.18
13.5	19	2.28
28.5	41	5.04
41.5	60	7.44
55.5	80	10
71.5	104	13
239.5	350	43.9

1. HSE=8MHz, System clock=72M

2. f_{ADC}=14MHz

Fig. 1.3