Shanghai Fudan Microelectronics Group Company Limited

复旦微电子

FM13DT160

Dual Frequency Three Interfaces Temperature Sensor and Logger IC

Datasheet

Sep. 2020

Trademarks

Contents

1 Product Overview

1.1 Introduction

FM13DT160 (DT160 for short) is a sensor IC optimized for temperature measurement and logging which has three interfaces. The contact interface follows I2C communication protocol. The contactless interface provides a HF (NFC) port operating in 13.56MHz and an UHF port operating around 900MHz. Benefited from the NFC port, users can read the sensor data log by a NFC smart phone which has Android system or IOS system. Through the UHF port, the sensor data can be gotten by an UHF reader which provides a long distance and dense reading ability.

DT160 can work in a semi-active mode powered by a battery or a passive mode powered by the RF field of the reader. When used as a temperature sensor data logger, it should be powered by a button battery, or a thin-film battery which turns the sensor into a thin stick-on RFID tag. In the passive mode, DT160 can accept the measurement command from the HF or UHF reader and fulfills a temperature measuring operation powered by the RF field.

In the semi-active mode, an embedded RTC (Real-time Clock) can be used to fulfill regular measurement and logging which could be used in cold chain application. The time step can be configured flexibly. The RTC can work with a current lower than 1uA. A continuous temperature logging which starts every 5 minutes can work for one year powered by an 18mAh battery. In this mode, DT160 can also start a single time measurement by receiving a command from the UHF reader. Thanks for the battery assistant, it provides an ultra high sensitivity which means a longer measurement distance and a more reliable communication.

The chip has a high-capacity EEPROM which provides a long time logging ability whose potential is 20,000 temperature points. It can be used to record the temperature data for a period up to 2 months with a 5 minutes measurement interval.

DT160 can be controlled by an off chip MCU communicating by the I2C port. In this application, the EEPROM can also be used to store the temperature data which can be read by the HF and UHF interfaces.

DT160 has HF and UHF energy harvesting ability. The energy can be used to light an off-chip LED which can provide visible information to indicate the current measuring state or provide a warning message.

A light sensor is embedded in the chip. It provides a rough light threshold which can be used to generate a warning data written to the memory together with the temperature data. This feature can be used in some light sensitive pharmaceuticals application.

DT160 has a multi-functional IO pin which can be connected with an off chip humidity sensor. A piezoresistor can also be connected to this IO pin to fulfill a pressure measurement.

1.2 Features

1.2.1 Interface

- **Contactless-UHF**
  - EPC Global C1G2 V1.2.0 & ISO/IEC 18000-63
  - Operating Frequency：840~960MHz
  - Communication baud rate（reader to tag）：40~160k bits/s

- Communication baud rate（tag to reader）：5k~640k bits/s
- Temperature measurement sensitivity (battery assistant mode)：-25dBm

- **Contactless-HF (NFC)**
- ISO/IEC 15693 & NFC FORUM TypeV (T5T) or ISO/IEC14443 & NFC FORUM Type2(T2T)
- Operating Frequency：13.56MHz
- Resonant capacitance：23.5pF
- Temperature measurement distance：5cm（up to the reader and the tag antenna）
- Communication baud rate：26k bits/s for ISO15693; 106k bits/s for ISO14443

- **Contact**
- I2C
- Power supply：2.7V~3.6V
- Communication baud rate：100k bits/s

1.2.2 EEPROM

- Total memory：164kbits
- Two configurable sensor data storage area, the maximum size is 160kbits
- User area size is configurable：0~8kbits
- Potential temperature logging points: 20000 (8 bits dense data mode)
- Potential temperature logging points: 16000 (10 bits dense data mode)
- Potential temperature logging points: 4864 (10 bits sparse data mode, full information)

1.2.3 Temperature Measurement and Logging

- Battery Supply Voltage：1.1V~1.65V
- Precise Temperature Measurement Range：-35℃~50℃
- Temperature Sensor Absolute Accuracy：±0.5℃
- RTC Measurement Interval: 1s ~ 65535s
- Configurable Delay time for starting RTC logging: 1m~65535m
- Configurable stop time for the measurement and logging
- Multi-mode for logging: normal mode and limit mode
- RTC accuracy：±2%@-20℃~50℃

1.2.4 Security

- HF UID is unchangeable
- UHF TID is unchangeable
- Access to user area is protected by password
- Lock mechanism for user memory block (write protection)
- Temperature data area is read-only for the contactless interface

1.2.5 Special feature

- RF field energy harvesting：Output voltage ≤ 5V
- LED lighted function in passive mode and semi-active mode
- Battery low voltage warning function
- Light strength warning function

1.3 Applications

- Cold chain evaluation for medical, fresh food, special chemicals
- Long distance temperature measurement and monitor
- NFC sensor for body or environment temperature measurement
- Pressure measurement

1.4 Package

1.4.1 DFN10

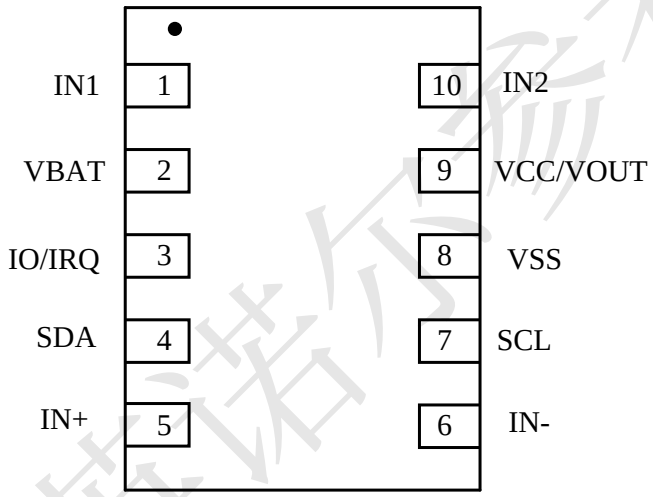


Figure 1-1 DFN10 Package

Number	PIN name	PIN Type	Description
1	IN1	Analog input	HF antenna pin
2	VBAT	Power	Anode of the battery
3	IO/IRQ	Digital inout Analog inout	(1) Analog input, connected with a humidity sensor; (2) Analog output, connected with a piezoresistor; (3) Output a digital IRQ signal with an off-chip pull-up resistor (4) Control the LED's lighting;
4	SDA	Digital input/OD output	I2C data pin
5	IN+	Analog input	UHF antenna pin
6	IN-	Analog input	UHF antenna pin
7	SCL	Digital input	I2C clock pin
8	VSS	Power ground	Ground pin, Cathode of the battery
9	VCC/VOUT	Power in / Power out	Power supply of the I2C interface / Energy harvesting output
10	IN2	Analog input	HF antenna pin

Table 1-1 DFN10 Pin Description

1.4.2 Bare die with Gold bumps

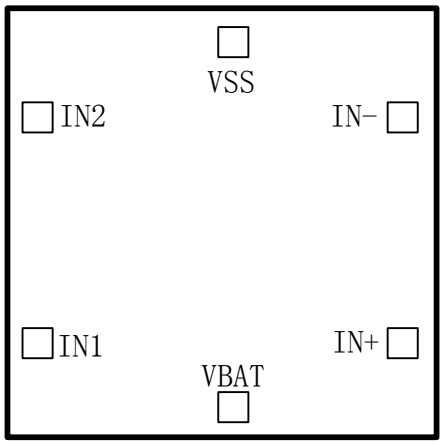


Figure 1-2 Bumps Pin

Number	PIN name	PIN Type	Description
1	IN1	Analog input	HF antenna pin
2	IN2	Analog input	HF antenna pin
3	VBAT	Power	Anode of the battery
4	IN-	Analog input	UHF antenna pin
5	IN+	Analog input	UHF antenna pin
6	VSS	Power ground	Cathode of the battery

Table 1-2 Bump Pin Description

1.4.3 WLCSP

WLCSP（Wafer Level Chip Scale Packaging) is in development.

2 Functional Description

2.1 General Description

The DT160 chip consists of the following blocks:

- RF analog blocks: signal's modulator and demodulator of the HF interface and the UHF interface.
- Power management: Manage the power from four sources: Battery, VCC of contact interface, rectified power from the HF field, rectified power from the UHF field.
- Temperature sensor: Including a semiconductor temperature sensor and an ADC which converts the analog data to digital code.
- RTC: Including a low power RC oscillator which outputs a high accuracy frequency clock, low power digital counter and timer. These blocks are powered by the battery.
- Digital control logic: handling the communication protocol of HF, UHF, and I2C, managing the measurement process and the data logging process.
- EEPROM: storage of the communication protocol information, user data, temperature data etc.

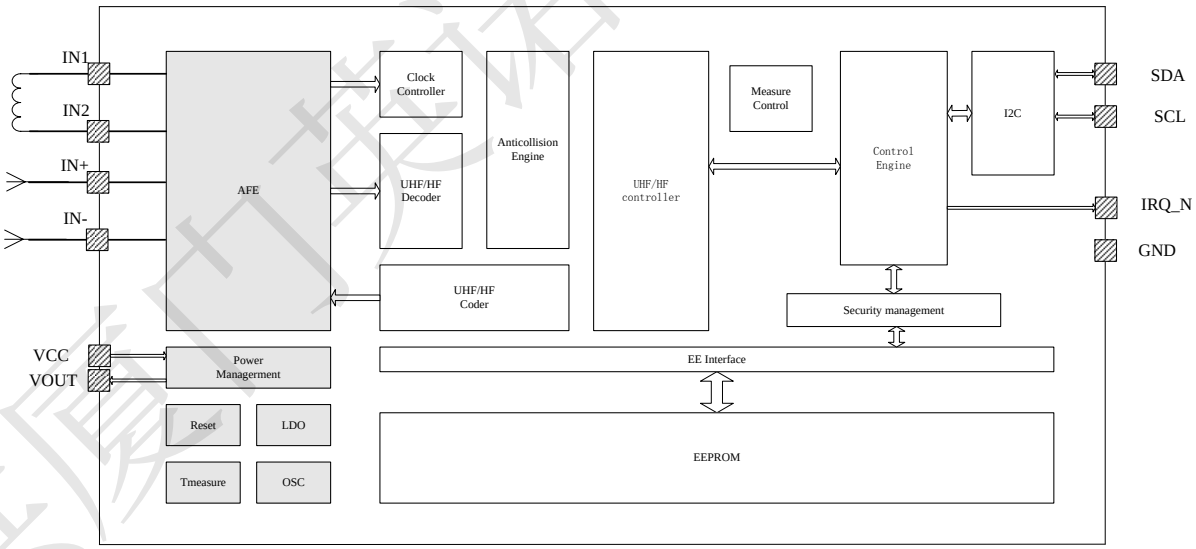


Figure 2-1 DT160 Block Diagram

2.2 Memory

2.2.1 Physical Organization

The memory of DT160 is a physical EEPROM. The 164kbits of the total memory are organized in blocks each 4 bytes which can be accessed by the three interfaces: HF, UHF and I2C. It is

divided into three areas which are configuration area, user area and sensor data area.

The configuration area is used to store the UHF protocol information including TID data, EPC data, and reserved passwords. UID of the HF interface, chip configuration information are also stored in this area.

The user area serves as the storage of the three interface's information such as the user data of HF and UHF interface. Commands such as Read or Write from UHF, HF, and I2C interface can access this area separately. The arbitration mechanism is first come first served. The size of the user area can be configured from 0kbits to 8kbits whose minimum step is 1kbits.

Sensor data area is divided into two areas whose size can be configured flexibly. These two areas are used to store the temperature logging data. If the user area is not needed, its size can be configured to 0kbits. Then the total sensor data area will be 160kbits.

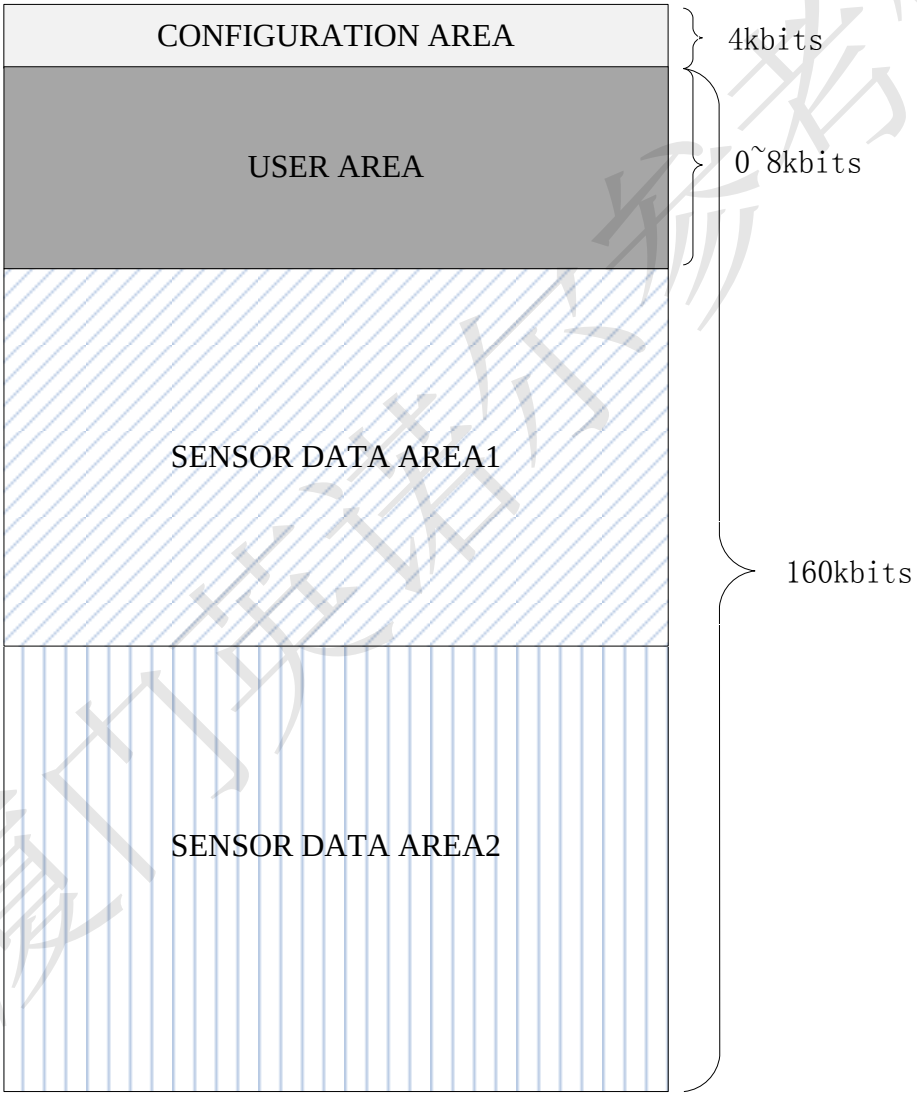


Figure 2-2 DT160 Memory physical organization

2.2.2 UHF Configuration memory

Following the EPC Global protocol, the three configuration areas of UHF are showed below.

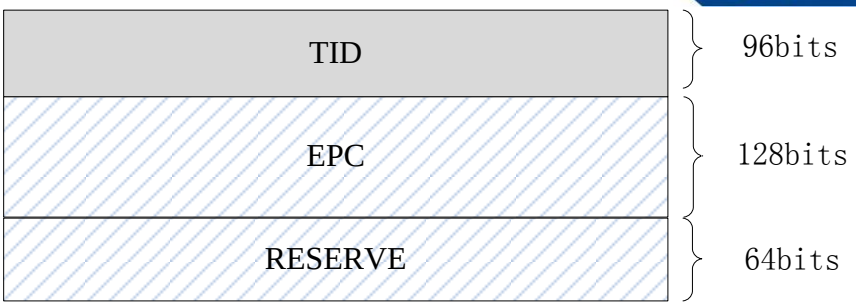


Figure 2-3: UHF Configuration Memory

2.2.3 Logical Organization

DT160's logical mapping is shown below.

The minimum individually-addressable unit of EEPROM is 1byte. However, the minimum addressable unit of REGISTOR is 2bytes.

The user area's address range is 0000h~03ffh whose maximum volume is 1kbytes. The actual volume is up to the configuration.

The temperature data area part0's address range is 1000h~5fffh. Its actual volume is up to the configuration.

The temperature data area part1's address range is 6000h~afffh. Its actual volume is up to the configuration.

The configuration area's address range is b000h~b1ffh whose volumn is 512 bytes.

The register's address range is c000h~c1ffh. It is used to change the chip's configuration instantly.

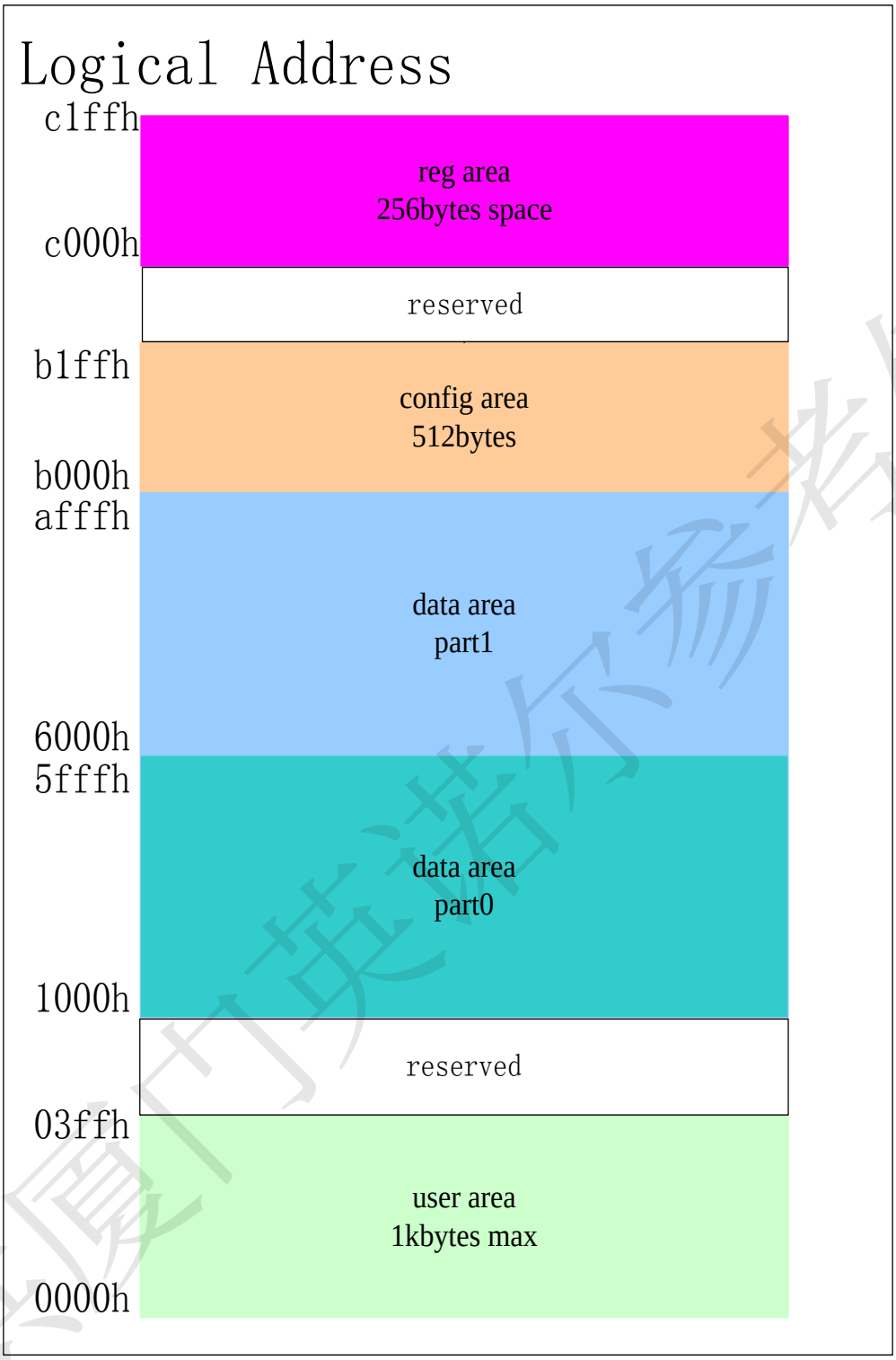


Figure 2-4: Logical Organization

2.2.4 Configuration Area

Description:

1. RFU: reserved for future use, default value is 0.

2. DTU: can not be used by the customer. The initial data should be kept and can not be changed.

3. The data should be read in a sequence of byte0 to byte3. The minimum unit is one byte.

For example:

sector	page	addr	BYTE			
			3	2	1	0
1	7	0xb05c	i2c_cfg		RFU	

If the content of block in the table above will be read, the data should be sent in the sequence from left to right of the below table.

8'h00	8'h00	i2c_cfg[7:0]	i2c_cfg[15:8]

2.2.4.1 Sector1

2.2.4.1.1 Overview

sector	block	addr	BYTE			
			3	2	1	0
1	0	0xb040	~user_cfg1	user_cfg1	~user_cfg0	user_cfg0
	1	0xb044	~user_cfg3	user_cfg3	user_cfg2	user_cfg2
	2	0xb048	vdet_offset		data_area_start_block_pointer	
	3	0xb04c	vdet_B		vdet_A	
	4	0xb050	RFU		RFU	
	5	0xb054	data_area_part0_size	RFU	user_area_size	
	6	0xb058	RFU		RFU	
	7	0xb05c	i2c_cfg		RFU	
	8	0xb060	filed_chk_cfg	led_mode_cfg1	led_mode_cfg2	RFU
	9	0xb064	RFU		RFU	
	10	0xb068	RFU		RFU	
	11	0xb06c	RFU		RFU	
	12	0xb070	RFU		RFU	
	13	0xb074	RFU		RFU	
	14	0xb078	RFU		RFU	
	15	0xb07c	sector1_lock	RFU	RFU	

Note： user_cfg_byte0、user_cfg_byte1、user_cfg_byte2 need ones-complement code verification.

2.2.4.1.2 Description

Name	Size（byte）	Function Description
user_cfg	6	User config info is stored in these 6 bytes(ones-complement code verification is needed). user_cfg0、~user_cfg0（ones-complement code） user_cfg1、~user_cfg1（ones-complement code） user_cfg2、~user_cfg2（ones-complement code）

Name	Size（byte）	Function Description	
		user_cfg3、~user_cfg3（ones-complement code）	
data_area_start_block_pointer	2	These 2bytes defines the start point of the address when the temperature data need to be written to the data area in the temperature logging process. For example, if data_area_start_block_pointer=0，the data will be written from the block0 of data area.	
vdet_A	2	These 2bytes is one of the 3 parameters to calculate the actual temperature data。 T=vdet_a*x(t)+vdet_b+offset The default value of vdet_a is 618.625 which is 26aah in EEPROM storage. The high 12bits is the integer part and the low 4bits is the decimal parts. The MSB is the symbol bit.	
vdet_B	2	These 2bytes is one of the 3 parameters to calculate the actual temperature data。 T=vdet_a*x(t)+vdet_b+offset The default value of vdet_b is -287.125 which is ee0eh in EEPROM storage. The high 12bits is the integer part and the low 4bits is the decimal part. The MSB is the symbol bit.	
vdet_offset	2	These 2bytes is one of the 3 parameters to calculate the actual temperature data。 T=vdet_a*x(t)+vdet_b+offset The default value of offset is 0 which is 0000h in EEPROM storage. The high 12bits is the integer part and the low 4bits is the decimal part. The MSB is the symbol bit.	
user_area_size	2	These 2bytes is used to config the size of the user area	
		[3:0]	Fixed to 4’hf, cannot be changed.
		[7:4]	User area's size is defined to (user_area_size[7:4]+1)*16*block
		[12:8]	Fixed to ’h3f, cannot be changed.
		[15]	To define if the user area exists. 0：user area does not exist.(user_area_size[12:0] is invalid) 1：user_area exists.
data_area_part0_size	1	Define the size of the data area part1. The actual size is data_area_part0_size[4:0]*1024bytes.	
field_chk_cfg	1	field_chk_cfg[7:4]：Configurate the strength threshold of the HF field detection. field_chk_cfg[3:0]：Configurate the strength threshold of the UHF	

Name	Size（byte）	Function Description		
		field detection.		
i2c_cfg	2	[15:10]	RFU	
		[9]	The enable bit of I2C's multi slave mode, active high.	
		[8]	The enable bit of I2C's CRC check, active high.	
		[7]	RFU	
		[6:0]	I2C slave interface address	
led_mode_cfg	2	Used to set the flash mode of the LED in logging process.		
		led_mode_cfg1	[7:4]	RFU
			[3:0]	vdet_flash_len_cfg[3:0] Definition of the duration time of LED lighting which starts after the single step measurement has just completed in the temperature logging process. t=(cfg+1)*100ms
		led_mode_cfg2	[7:4]	vdet_limit_times_cfg[7:4] Definition of times of LED flash as a warning message when the logging temperature is over the limit threshold.
			[3:0]	vdet_limit_len_cfg[3:0] Definition of the pulse width of LED flash when the logging temperature is over the limit threshold. t=(cfg+1)*100ms
sector1_lock	1	The lock byte of sector1. 8'h5a ： The sector1 will be locked to be readonly. other ： The sector1 is not locked.		

2.2.4.1.3 **User_cfg0**

Cfg	bit	bit name	description
user_cfg0	[1:0]	io_pad_cfg[1:0] (default 00)	Define IO/IRQ pad's function： 00：RFU 01：Light the off-chip LED 10：Send an interrupt signal to the off-chip MCU 11：Analog signal input, used to receive the off chip sensor's measurement result.
	[4:2]	temp_format_cfg[2:0] (default 011)	Define the store format of the temperature data: 000： compress mode 0 001： compress mode 1 010： compress mode 2 011： normal mode

Cfg	bit	bit name	description
			100： limit mode 0 101： limit mode 1 110： limit mode 2 111： original data mode
	[5]	RFU	RFU
	[6]	rtc_vdet_auto_finish_en (default 0)	The enable bit of the logging automatically stop function, active high. When the measurement times counter reach the maxium value set in configuration area, the logging process will be terminated automatically and the chip will enter the PD mode. The Wakeup command is needed to wake the chip into standby mode.
	[7]	temp_format_high_preci_en (default 0)	Configurate the accuracy of the temperature data when the storage format is in 10bits mode. The default value is "0". 0: The temperature data include 8bits integer and 2bits decimal. 1: The temperature data include 7bits integer and 3bits decimal.

2.2.4.1.4 **User_cfg1**

cfg	bit	bit name	description
user_cfg1	[0]	DTU	DTU
	[1]	data_area_ovflow_mode (default 0)	Determine how to deal with the new logging data when the data area is full. 0: abandon the new data. The logging process will be terminated automatically when the data area is full. 1: RFU.
	[2]	logging_data_area_sel (default 0)	Select which data area is used to store the logging data. 0: data area0 is selected 1: data area1 is selected
	[3]	DTU	Keep 0
	[4]	vdet_step_auto_cfg _en (default 0)	The enable bit of the function changing the logging step automatically. 0: disable 1: enable If this bit is set to “1”, when the temperature

cfg	bit	bit name	description
			measurement result beyonds the limited range, the logging step will be changed to the value that is defined by vdet_alarm_step_cfg. The limited range is defined by max_alarm_limit and min_alarm_limit.
	[5]	Led_auto_flash_en (default 1)	Define If the LED will be lightened to give a message that logging process is running normally after a single step measurement has finished. 0: disable 1: enable
	[7:6]	RFU	RFU

2.2.4.1.5 **User_cfg2**

cfg	bit	bit name	description
user_cfg2	[3:0]	RFU	RFU
	[6:4]	temp_format_bit_info_sel (default 000)	In compress mode1 and limit mode2, these bits are used to select which message will be logged in the data. Refer to 2.2.5 for detail.
	[7]	light_chk_en (default 0)	If the light strength detect function is enabled in temperature logging process. 0: disable 1: enable

2.2.4.1.6 **User_cfg3**

cfg	bit	bit name	description
user_cfg3	[2:0]	io_int_en (default 111)	Interrupt source selection on IO/IRQ pad: Bit0=1: sending an interrupt signal when entering or leaving the HF field range. Bit1=1: sending an interrupt signal when received an effective ACK command from the UHF reader. Bit2=1: sending an interrupt signal at the end of the single step measurement in the logging process.
	[3]	io_int_mode (default 0)	Interrupt type selection: 0: level-trigged interrupt, a command sent from MCU or Reader is needed to clear this interrupt. 1: edge-trigged interrupt, the pulse width is about 2us.

2.2.4.2 Sector2

2.2.4.2.1 Overview

sector	block	addr	BYTE			
			3	2	1	0
2	0	0xb080	max_limit0		min_limit0	
	1	0xb084	max_limit1		min_limit1	
	2	0xb088	max_limit2		min_limit2	
	3	0xb08c	Max_alarm_limit		Min_alarm_limit	
	4	0xb090	RFU		RFU	
	5	0xb094	RFU		rtc_cnt_limit	
	6	0xb098	RFU		RFU	
	7	0xb09c	RFU		RFU	
	8	0xb0a0	RFU		RFU	
	9	0xb0a4	vdet_alarm_step_cfg		vdet_step_cfg	
	10	0xb0a8	RFU		RFU	
	11	0xb0ac	RFU		RFU	
	12	0xb0b0	RFU		DTU	
	13	0xb0b4	RFU		RFU	
	14	0xb0b8	RFU		RFU	
	15	0xb0bc	Sector2_lock		RFU	

2.2.4.2.2 Description

Name	Byte size	Description

min_limit0/1/2	6	The low threshold value if the data storage format is Limit mode. The value is 10bits length whose MSB is the sign bit.
		For example:
		If the low threshold min_limit0= -15.5^{o}C, then:
		min_limit=$(11110000.10)_2$ when Temp_format_high_preci_en=0;
		min_limit=$(1110000.100)_2$ when Temp_format_high_preci_en=1;
		
		min_limit0 is also set as the default value of register "summary_min_limit_cnt" (0xc09b)

max_limit0/1/2	6	The high threshold value if the data storage format is Limit mode. The value is 10bits length. The MSB is the sign bit.
		For example:
		If the low threshold max_limit0= +25.25^{o}C, then:
		max_limit=$(00011001.01)_2$ when Temp_format_high_preci_en=0;
		max_limit=$(0011001.010)_2$ when Temp_format_high_preci_en=1;
		
		max_limit0 is also set as the default value of register

Name	Byte size	Description
		"summary_max_limit_cnt" (0xc09a)
min_alarm_limit max_alarm_limit	2	The threshold value of logging step auto change function. When the tempeature measurement result beyonds the limited range, the logging step will be changed to the value that is defined by vdet_alarm_step_cfg.
rtc_cnt_limit	2	The maximum logging times. When the counter reaches this value, logging process will be terminated automatically.
vdet_step_cfg	2	The time step of logging process. The minimum time unit is 1s.
vdet_alarm_step_cfg	2	The new time step when the temperature data beyonds the range defined by max_alarm_limit or min_alarm_limit.
Sector2_lock	1	The lock byte of sector2: 8'h5a ： The sector1 will be locked to be read only. other ： The sector1 is not locked.

2.2.4.3 Sector3

2.2.4.3.1 Overview

sector	block	addr	BYTE			
			3	2	1	0
3	0	0xb0c0	DTU		vbe_trim	
	1	0xb0c4	DTU		DTU	
	2	0xb0c8	DTU		DTU	
	3	0xb0cc	DTU	ana_cfg2	DTU	
	4	0xb0d0	DTU		DTU	ana_cfg4
	5	0xb0d4	DTU		DTU	
	6	0xb0d8	DTU		DTU	
	7	0xb0dc	DTU		DTU	
	8	0xb0e0	DTU		DTU	
	9	0xb0e4	DTU		DTU	
	10	0xb0e8	DTU		DTU	
	11	0xb0ec	DTU		DTU	
	12	0xb0f0	DTU		DTU	
	13	0xb0f4	DTU		DTU	
	14	0xb0f8	DTU		DTU	
	15	0xb0fc	Sector3_lock	DTU	DTU	

2.2.4.3.2 Description

Name	byte	Description

	size	
vbe_trim	2	[15:12] RFU [11: 0] the parameter of temperature measurement, The recommended value is 600 for two-point temperature calibration.(the chip is already configed in the factory。This parameter will influence the result when measure the temperature。)
ana_cfg2	1	[15:6]：fixed to 0； [5:0]：configuration of the threshold value of light strength detection
ana_cfg4	1	the parameter of temperature measurement，The recommended value is 49 for two-point temperature calibration.(the chip is already configed in the factory。This parameter will influence the result when measure the temperature。)
Sector3_lock	1	The lock byte of sector3: 8'h5a ： The sector1 will be locked to be readonly. other ： The sector1 is not locked.

2.2.4.4 Sector4

2.2.4.4.1 Overview

sector	block	Addr	BYTE			
			3	2	1	0
4	0	0xb100	lock bit	lock bit	lock bit	lock bit
	1	0xb104	lock bit	lock bit	lock bit	lock bit
	2	0xb108	lock bit	lock bit	lock bit	lock bit
	3	0xb10c	lock bit	lock bit	lock bit	lock bit
	4	0xb110	lock bit	lock bit	lock bit	lock bit
	5	0xb114	lock bit	lock bit	lock bit	lock bit
	6	0xb118	lock bit	lock bit	lock bit	lock bit
	7	0xb11c	lock bit	lock bit	lock bit	lock bit
	8	0xb120	user area password			
	9	0xb124	RFU			
	10	0xb128	RFU			
	11	0xb12c	unlock password			
	12	0xb130	Stop logging password			
	13	0xb134	RFU		RFU	
	14	0xb138	RFU		RFU	auth_rb_cfg
	15	0xb13c	RFU		RFU	

2.2.4.4.2 Description

Name	byte size	Description
Lock	32	The lock bit of the user area in ISO15693 interface. Each bit is used to

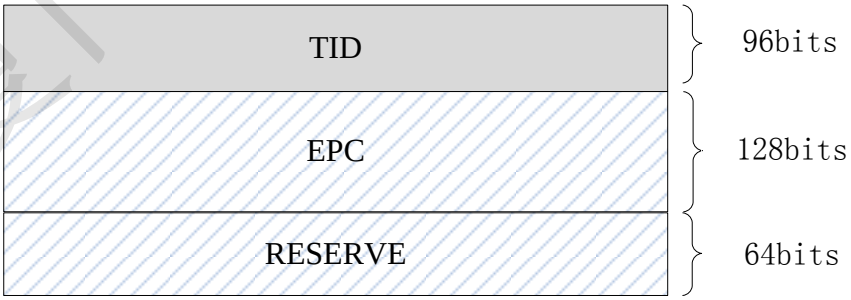
Name	byte size	Description
		lock 4byte of the user memory to read only.
user area password	4	Used to control the W/R authority of the user area. It will be active when the value is not 0.
unlock password	4	Password to control the Sector_lock. When the sector_lock is set, the configuration bit in sector1~sector3 can not be changed anymore. But if the unlock password check has been executed successfully, the sector_lock will be uneffective in this time of power on. The configuration bits can be changed by the received command.
Stop logging password	4	Password to control the authority of clearing the logging process flag(RTC logging). The RTC logging=1 means that the logging process is running. the Star_RTC command will not be executed in this state. When clearing RTC logging by sending stop logging command, if this password is not "0", verification need to be done firstly. Otherwise, error code will be send back and the clearing operation will not be executed successfully.
auth_rb_cfg	1	The mask byte of the 32bits password. Refer to 2.6.5.2 and other document for detail decription.

Note:
Sector4 has not sector_lock bit. The lock bit in this sector can not be read or write directly. Password check by sending an Auth command is needed first.

2.2.4.5 Sector5

2.2.4.5.1 Overview

This sector serves as the configuration memory for UHF interface. Following the EPC Global protocol, the three configuration areas of UHF are showed below. The reserve area stores 32bits kill password and 32bits access password.



sector	block	Addr	BYTE			
			3	2	1	0
4	0	0xb140	kill password[0:31]			
	1	0xb144	access password[0:31]			
	2	0xb148	store pc[0:15]		store crc[0:15]	
	3	0xb14c	EPC	EPC	EPC	EPC

sector	block	Addr	BYTE			
			3	2	1	0
	4	0xb150	EPC	EPC	EPC	EPC
	5	0xb154	EPC	EPC	EPC	EPC
	6	0xb158	RFU	RFU	RFU	RFU
	7	0xb15c	RFU	RFU	RFU	RFU
	8	0xb160	TID[0:31]			
	9	0xb164	TID[63:32]			
	10	0xb168	TID[95:64]			
	11	0xb16c	RFU			
	12	0xb170	RFU			
	13	0xb174	RFU			
	14	0xb178	RFU		lock_cfg[9:0]	
	15	0xb17c	DTU		RFU	

2.2.4.5.2 **Description**

Name	byte size	Description
kill_password	4	Password for controlling kill command
access_password	4	Password for controlling access command
EPC	12	EPC code
TID	12	TID of the tag
lock cfg[9:0]	2	Lock bits

lock cfg										
byte l(0xb178)								byte h(0xb179)		
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	bit7	bit6	bit5-0
kill pwd		access pwd		EPC memory		TID memory		user memory		RFU
R/W	perma lock	R/W	perma lock	W	perma lock	W	perma lock	W	perma lock	RFU

W	permalock	description
0	0	The corresponding memory area is readable and writeable in OPEN/SECURED state
0	1	The corresponding memory area is readable and writeable permanently in OPEN/SECURED state. It is unlockable.
1	0	The corresponding memory area is writeable in SECURED state and is readonly in OPEN state.

1	1	The corresponding memory area is readonly in any state.

R/W	permalock	description
0	0	The corresponding password is readable and writeable in OPEN/SECURED state.
0	1	The corresponding password is readable and writeable permanently in OPEN/SECURED state. It is unlockable.
1	0	The corresponding password is writeable in SECURED state and is not writeable and not readable in OPEN state.
1	1	The corresponding password is not writeable and not readable in any state.

Note：Permalock bits can not be changed if it is set to 1. The chip will not execute a LOCK command which wants to change this “1” state bit and response with an error code.

2.2.4.6 Sector6

2.2.4.6.1 Overview

sector	block	Addr	BYTE					
			3		2	1		0
4	0	0xb180	RFU	min_temperature [9:8]	min_temperature[7:0]	RFU	max_temperature[9:8]	max_temperature[7:0]
	1	0xb184	min_limit_cnt			max_limit_cnt		
	2	0xb188	data_area_pointer_status			data_area_block_pointer		
	3	0xb18c	RFU		RFU	RFU		RFU
	4	0xb190	RFU		RFU	RFU		RFU
	5	0xb194	RFU		RFU	RFU		RFU
	6	0xb198	RFU		RFU	RFU		RFU
	7	0xb19c	RFU		RFU	RFU		RFU
	8	0xb1a0	RFU		RFU	RFU		RFU
	9	0xb1a4	RFU		RFU	RFU		RFU
	10	0xb1a8	RFU		RFU	RFU		RFU
	11	0xb1ac	RFU		RFU	RFU		RFU
	12	0xb1b0	RFU		RFU	RFU		RFU
	13	0xb1b4	RFU		RFU	RFU		RFU
	14	0xb1b8	RFU		RFU	RFU		RFU
	15	0xb1bc	RFU		RFU	RFU		RFU

2.2.4.6.2 Description

This area stores some indication information is read only.

Name	byte size	Description	
max_temperature[9:0]	2	The maximum temperature data in all of the logging data	
min_temperature[9:0]	2	The minimum temperature data in all of the logging data	
max_limit_cnt	2	The times of the logging data beyonds the limit threshold which is defined by Max_alarm_limit.	
min_limit_cnt	2	The times of the logging data beyonds the limit threshold which is defined by Min_alarm_limit.	
data_area_block_pointer	2	The page pointer of the data area which will be written into the next logging data.	
data_area_pointer_status	2	Data_area_pointer_status [15:0] is the indication information of how to store the next logging data.	
		[1:0]	inblock_pointer: When the selected data storage format is to store multiple logging datas in one block, inblock_point is used to indicate the position of the block where the next data will be saved in.
		[4:3]	limit_mode1_area_flag: When the selected data storage format is limit_mode_1, limit_mode1_area_flag is used to indicate which temperature range the last logging data locates in. 10: The logging temperature data is higher than the high threshold. 01: The logging temperature data is lower than the low threshold. 00: The logging temperature data is between the high threshold and the low threshold. 11: RFU
		[7:5]	temp_data_flag: In compress mode2, this section is used to temporary store the part of the last logging data which can not fill in one block. This part of data will be written to the next block with the new data after the next measurement finished. Temp_data_flag [7:5] is used to indicate how many bits need to store in Temp_data_flag [15:8]. 000: no data 001: 2bits 010: 4bits 011: 6bits 100: 8bits

Name	byte size	Description	
			Others: RFU
		[15:8]	Temporary store the part of the last logging data which can not fill in one block. MSB is effective.

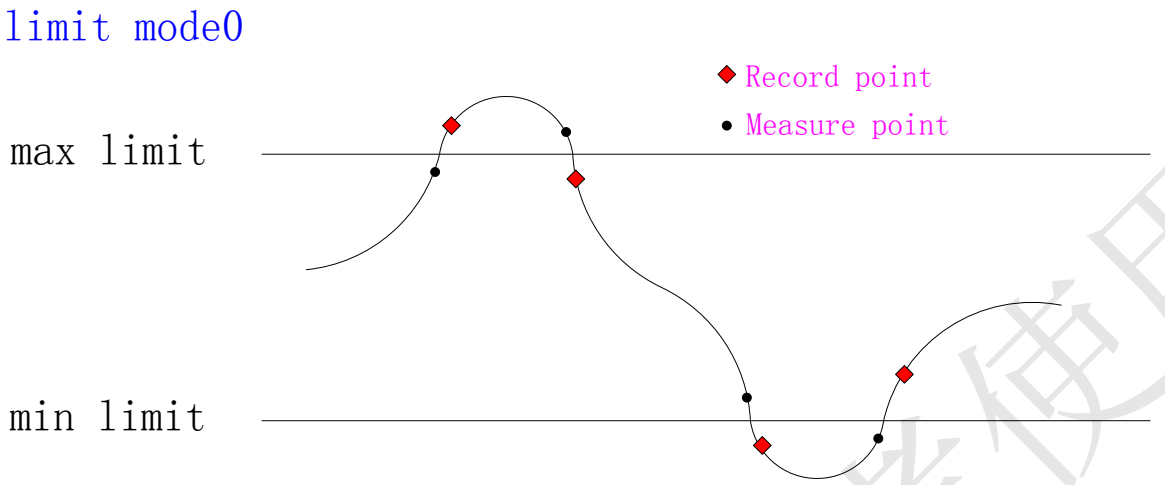
2.2.5 Data Storage Format

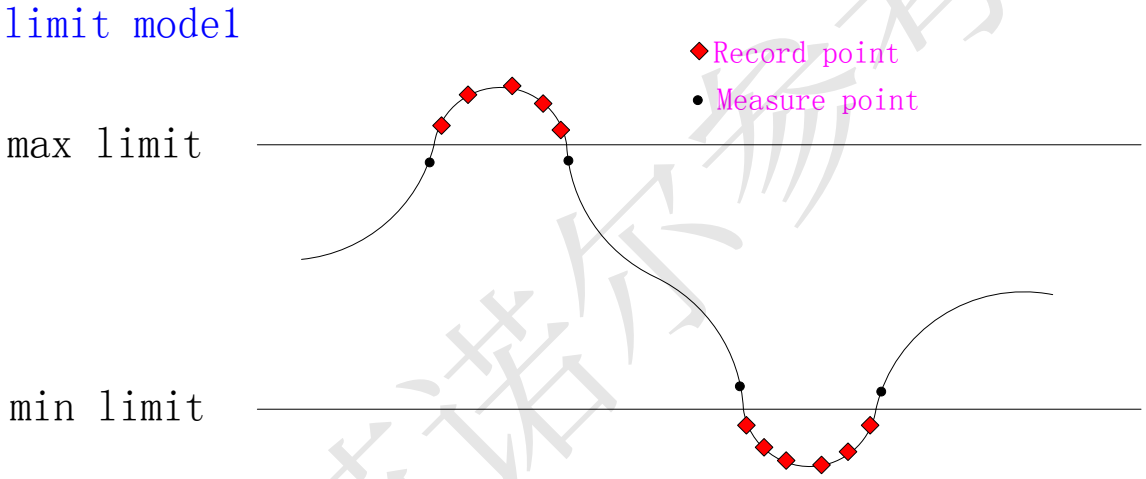
<table>
<tr><th>Storage Format</th><th colspan="5">Description</th><th>Potential Logging points</th></tr>
<tr><td rowspan="3">(000) compress mode 0</td><td colspan="5">The length of the logging data is 8bits.
The 8bits data includes 7bits integer and the MSB is the sign bit when temp_format_high_preci_en=0.
The 8bits data includes 6bits integer, 1bits decimal and the MSB is the sign bit when temp_format_high_preci_en=1.</td><td rowspan="3">20480</td></tr>
<tr><td>bit31-24</td><td colspan="2">bit23-16</td><td>bit15-8</td><td>bit7-0</td></tr>
<tr><td>temperature value3</td><td colspan="2">temperature value2</td><td>temperature value1</td><td>temperature value0</td></tr>
<tr><td rowspan="11">(001) compress mode 1</td><td colspan="5">The logging data is 10bits.
There are 3 temperature datas in one block.
The 10bits data includes 7bits integer, 2bits decimal and the MSB is sign bit when temp_format_high_preci_en=0.
The 10bits data includes 6bits integer, 3bits decimal and the MSB is sign bit when temp_format_high_preci_en=1.</td><td rowspan="11">15360</td></tr>
<tr><td>bit31</td><td>bit30</td><td>bit29-20</td><td>bit19-10</td><td>bit9-0</td></tr>
<tr><td>Flag1</td><td>Flag2</td><td>temperature value2</td><td>temperature value1</td><td>temperature value0</td></tr>
<tr><td colspan="5">Bit31 and bit30 can be selected from the below items:
Parity, voltage_flag, field_ext_flag, light_flag
The selection is controlled by temp_format_bit_info_sel[2:0]:</td></tr>
<tr><td>bit info sel</td><td colspan="2">Flag1</td><td colspan="2">Flag2</td></tr>
<tr><td>000(default)</td><td colspan="2">parity</td><td colspan="2">voltage_flag</td></tr>
<tr><td>001</td><td colspan="2">parity</td><td colspan="2">field_ext_flag</td></tr>
<tr><td>010</td><td colspan="2">parity</td><td colspan="2">light_flag</td></tr>
<tr><td>011</td><td colspan="2">voltage_flag</td><td colspan="2">field_ext_flag</td></tr>
<tr><td>100</td><td colspan="2">voltage_flag</td><td colspan="2">light_flag</td></tr>
<tr><td>101</td><td colspan="2">field_ext_flag</td><td colspan="2">light_flag</td></tr>
<tr><td>110/111</td><td colspan="2">parity</td><td colspan="2">voltage_flag</td></tr>
<tr><td>(010) compress</td><td colspan="5">The logging data is 10bits which is saved in the data area one by one.
The left space that is less than 10bits is used to store a part of the next</td><td>16384</td></tr>
</table>

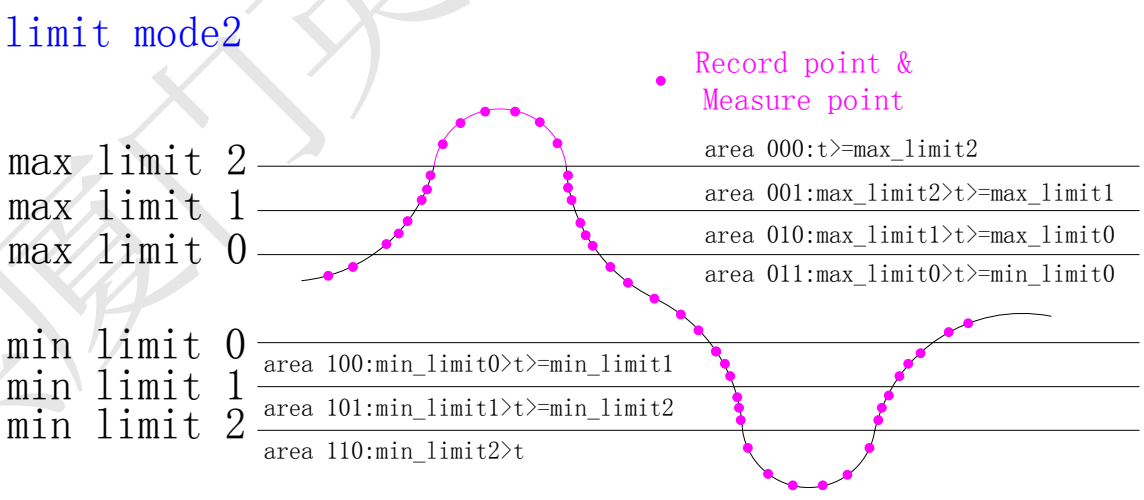
<table>
<tr><th>Storage Format</th><th colspan="4">Description</th><th>Potential Logging points</th></tr>
<tr><td rowspan="3">mode 2</td><td colspan="4">data.
The 10bits data includes 7bits integer, 2bits decimal and the MSB is sign bit when temp_format_high_preci_en=0.
The 10bits data includes 6bits integer, 3bits decimal and the MSB is sign bit when temp_format_high_preci_en=1.</td><td rowspan="3"></td></tr>
<tr><td>bit31-30</td><td>bit29-20</td><td>bit19-10</td><td>bit9-0</td></tr>
<tr><td>temperature value3[1:0]</td><td>temperature value2</td><td>temperature value1</td><td>temperature value0</td></tr>
<tr><td rowspan="3">(011) nomal mode</td><td colspan="4">There is only one logging data saved in one block in this mode.
Bit12~bit15 stores light_flag, field_ext_flag, voltage flag, parity separately.
The 10bits data includes 7bits integer, 2bits decimal and the MSB is sign bit when temp_format_high_preci_en=0.
The 10bits data includes 6bits integer, 3bits decimal and the MSB is sign bit when temp_format_high_preci_en=1.</td><td rowspan="3">5120</td></tr>
<tr><td>bit31</td><td>bit30-16</td><td>bit15-12</td><td>bit9-0</td></tr>
<tr><td>parity</td><td>time numbers</td><td>bit15：parity
bit14：voltage flag
bit13：field_ext_flag
bit12：light_flag</td><td>temperature value</td></tr>
<tr><td rowspan="3">(100) Limit mode 0</td><td colspan="4">There are high threshold and low threshold in the limit mode0. The logging data will be memoryed one time only when its value is just beyond the threshold. Every logging data occupies one block.
The 10bits data includes 7bits integer, 2bits decimal and the MSB is sign bit when temp_format_high_preci_en=0.
The 10bits data includes 6bits integer, 3bits decimal and the MSB is sign bit when temp_format_high_preci_en=1.</td><td rowspan="3">-</td></tr>
<tr><td>bit31</td><td>bit30-16</td><td>bit15-12</td><td>bit9-0</td></tr>
<tr><td>parity</td><td>time number</td><td>bit15：parity
bit14：voltage flag
bit13：field_ext_flag
bit12：light_flag</td><td>temperature value</td></tr>
<tr><td rowspan="2">(101) limit mode 1</td><td colspan="4">There are high threshold and low threshold in the limit mode1. The all logging data will be memorized when their values beyond the range threshold. Every logging data occupies one block.
The 10bits data includes 7bits integer, 2bits decimal and the MSB is sign bit when temp_format_high_preci_en=0.
The 10bits data includes 6bits integer, 3bits decimal and the MSB is sign bit when temp_format_high_preci_en=1.</td><td rowspan="2">-</td></tr>
<tr><td>bit31</td><td>bit30-16</td><td>bit15-12</td><td>bit9-0</td></tr>
</table>

<table>
<tr><th>Storage Format</th><th colspan="5">Description</th><th>Potential Logging points</th></tr>
<tr><td></td><td>parity</td><td colspan="2">time number</td><td>bit15：parity
bit14：voltage flag
bit13：field_ext_flag
bit12：light_flag</td><td>temperature value</td><td></td></tr>
<tr><td>(110) Limit mode 2</td><td colspan="5">The full temperature range is divided into 7 zones by 3 high thresholds and 3 low thresholds. The logging data will be compared with the thresholds to select which zone it will locate in after the measurement completed. Only the zone number will be memoried whose length is 4bits.
<table><tr><td>bit7</td><td>bit6-4</td><td>bit3</td><td>bit2-0</td></tr><tr><td>flag</td><td>temperature area number</td><td>flag</td><td>temperature area number</td></tr></table>One temperature area number is followed by one flag. The flag in bit3 or bit7 can be selected from these items: parity, voltage_flag, field_ext_flag and light_flag which is determined by user_cfg.temp_format_bit_info_sel[2:0]:
000: parity
001: battery_voltage_flag
010: field_ext_flag
others: light_flag</td><td>40960</td></tr>
<tr><td>(111) Original data mode</td><td colspan="5">In this mode, the original measurement data is saved which is the temperature logging data or the measurement result of off-chip sensors such as pressure or humidity.
There are 2 datas in one block.
<table><tr><td>bit31</td><td>bit30</td><td>bit28-bit16</td><td>bit15</td><td>bit14</td><td>bit12-0</td></tr><tr><td>Flag1</td><td>Flag2</td><td>data1</td><td>Flag1</td><td>Flag2</td><td>data0</td></tr></table>Flag1 and Flag2 can be selected from these items: parity, voltage_flag, field_ext_flag and light_flag as shown below.
<table><tr><th>bit info sel</th><th>Flag1</th><th>Flag2</th></tr><tr><td>000(default)</td><td>parity</td><td>voltage_flag</td></tr><tr><td>001</td><td>parity</td><td>field_ext_flag</td></tr><tr><td>010</td><td>parity</td><td>light_flag</td></tr><tr><td>011</td><td>voltage_flag</td><td>field_ext_flag</td></tr><tr><td>100</td><td>voltage_flag</td><td>light_flag</td></tr><tr><td>101</td><td>field_ext_flag</td><td>light_flag</td></tr><tr><td>110/111</td><td>parity</td><td>voltage_flag</td></tr></table></td><td>5000</td></tr>
</table>

More description about how to use limit mode is shown in below 3 figures.







2.2.6 UID&TID

2.2.6.1 UID of HF

FM13DT160 has two type of product whose HF interface can be chosen to be ISO15693 or ISO14443.

For the ISO15693 version, The 64-bit unique identifier (UID) in the configuration area is

programmed during the production process and cannot be changed afterwards.

The 64 bits are numbered according to ISO/IEC 15693-3 starting with LSB 1 and ending with MSB 64. The Fudan Micro Electronics' manufacturer code is"1Dh".

UID7	UID6	UID5	UID4	UID3	UID2	UID1	UID0
E0	1D	Serial Number					

For the ISO14443 version, the unique 7 byte serial number (UID) and its two Block Check Character Bytes (BCC) are programmed during the wafer test procedure. According to ISO/IEC 14443-3 BCC0 is defined as CT⊕SN0⊕SN1⊕SN2. Abbreviations CT stays for Cascade Tag byte (88h) and BCC1 is defined as SN3⊕SN4⊕SN5⊕SN6. SN0 holds the Manufacturer ID for Fudan (1Dh) according to ISO/IEC 14443-3 and ISO/IEC 7816-6 AMD.1.

Block Address	Byte Number inside a page			
	0	1	2	3
0	UID0	UID1	UID2	BCC0
1	UID3	UID4	UID5	UID6
2	BCC1	Internal	Lock Byte	Lock Byte

2.2.6.2 TID of UHF

The 96bits Tag identifier(TID) in the sector5 of the configuration area is programmed during the production process and will be read-only in the user mode.

The 96bits TID are numbered according to EPC Global C1G2 and ISO/IEC 18000-6C. The Fudan Micro Electronics' manufacturer code is "827h". XTID is fixed to "2000h".

TID0	TID1	TID2	TID3	TID4	TID5	TID6	TID7	TID8	TID9	TID10	TID11
Class ID	Msg code			XTID		SN5	SN4	SN3	SN2	SN1	SN0
E2	827	0	01	2000		Serial Number					

2.2.6.3 ID Uniformity

Since the chip's contactless interface conforms to three protocols, there are 2 UID of the HF interface and 1 TID of the UHF interface. For the convenience of chip identification, they share the same seriel number (SN5~SN0) as shown in below 3 tables.

ISO14443 UID：

UID6	UID5	UID4	UID3	UID2	UID1	UID0
SN5 (fixed)	SN4	SN3	SN2	SN1	SN0	Msg Code
70	40bits series number					1D

ISO15693 UID：UID is sent from tag to reader in sequence of UID0 to UID7 by bytes.

UID7	UID6	UID5	UID4	UID3	UID2	UID1	UID0
Class ID	Msg Code	SN5 (fixed)	SN4	SN3	SN2	SN1	SN0
E0	1D	70	40bits series number				

ISO18000-6C TID：TID is sent from tag to reader in sequence of TID0 to TID11 by words.

TID0	TID1	TID2	TID3	TID4	TID5	TID6	TID7	TID8	TID9	TID10	TID11
Class ID	Msg code			XTID		SN5 (fixed)	SN4	SN3	SN2	SN1	SN0
E2	827	0	01	2000		70	40bits series number				

2.3 Register

2.3.1 VCC power domain

2.3.1.1 ana_cfg1

Name	ana_cfg1		
Function	Configuration of Analog		
Address	16'hC012		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:4]	RFU	RW	All bits keep 0
[3]	ana_work_mode	RW	Work mode selection (default 0) 0: enter temperature measurement mode 1: enter other sensor mode, the source is pressure or battery voltage or the input analog signal from off-chip sensor. Get temperature command's parameter determines which source will be measured.
[2:0]	RFU	RW	All bits should keep 0

2.3.1.2 ana_cfg2

Name	ana_cfg2		
Function	Analog configuration		
Address	16'hC00B		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:6]	RFU	RW	All bits should keep 0
[5:0]	light_chk_cfg[5:0]	RW	Light strength detect threshold configuration whose default value is read out from ana_cfg2 in EEPROM's sector3 when the chip powers on. This register value should be configurated first before the light strength detection.

2.3.1.3 io_int_flag

Name	io_int_flag		
Function	IO interrupt flag		
Address	16'hC018		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:1]	-	-	-
[6]	HF_IRQ_flag	R	The interrupt flag of entering or leaving the HF field. Active high.
[5]	UHF_IRQ_flag	R	The interrupt flag of entering the UHF field. It will be changed from "0" to "1" when an ACK command received.
[4]	vdet_end_IRQ_flag	R	The interrupt flag which will be changed from "0" to "1" when the single logging step has been finished.
[3:1]	-	-	-
[0]	Clean_IRQ	RW_0	Used to clean the interrupt flags by writing "0" to this bit.

2.3.1.4 vdet_delay_start_cfg

Name	vdet_delay_start_cfg		
Function	The delay time before starting a logging process		
Address	16'hC084		
Default	0xffff		
Bit	Bit Name	Access	Description
[15:0]	vdet_delay_start_cfg	RW	The delay time to start a logging process. Unit is minute. the actual time is the decimal value converted from the binary [15:0]. For example: If the value is 0000 0001 1110, the delay time is 30m.

2.3.2 Battery power domain

2.3.2.1 vdet_step_cfg

Name	vdet_step_cfg0		
Function	The interval of temperature measurement in logging process		
Address	16'hC085		
Default	0xffff		
Bit	Bit Name	Access	Description
[15:0]	vdet_step_cfg	RW	Configuration of the time interval when logging. Unit is second.

2.3.2.2 vdet_result

Name	vdet_result

Function	Temperature measurement results		
Address	16'hC01E		
Default	0x00_00		
INFO position	-		
Bit	Bit Name	Access	Description
[15:13]	-	-	-
[12:0]	vdet_result [12:0]	R	Temperature measurement results, temperature calculation formula t=A*cnt/8192+B.

2.3.2.3 vdet_times_cnt

Name	vdet_times_cnt		
Function	RTC temperature measurement counter		
Address	16'hC091		
Default	0x00_00		
INFO position	-		
Bit	Bit Name	Access	Description
[15:0]	vdet_times_ cnt[15:0]	R	Record the times of current RTC delayed temperature measurement

2.3.2.4 rtc_delay_cnt

Name	rtc_delay_cnt		
Function	The current RTC timer counts		
Address	16'hC092		
Default	0x00_00		
INFO position	-		
Bit	Bit Name	Access	Description
[15:0]	rtc_delay_ cnt[15:0]	R	The current RTC timer counts in minutes for delayed start-up and in seconds for RTC temperature measurement

2.3.2.5 rtc_flow_block_pointer

Name	rtc_flow_block_pointer		
Function	Block pointer of RTC temperature measurement process		
Address	16'hC096		
Default	0x00_00		
INFO position	-		
Bit	Bit Name	Access	Description
[15:0]	rtc_flow_block_ pointer[15:0]	RW	Block_pointer Storing Current Storage Location in RTC Temperature Measurement Process. Note: The RTC process can not be modified by instructions.

2.3.2.6 summay_max_temperature

Name	summay_max_temperature

Function	The maximum temperature data gotten in logging process		
Address	16'hC098		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:10]	-	-	-
[9:0]	max_temperature[9:0]	RW	The maximu temperature data gotten in logging process. A value which is a slightly larger than the center temperature value should be written into this register before starting a logging process. The new measured temperature data will be compared with the value in this register. If the new data is bigger than the value, the new data will be written to the register automatically. So the value will be updated. This method can keep the register value as the maximum one. For example, if the temperature range of the application is 2℃~8℃ and the accuracy requirement is ±0.5℃, the initial value of the register can be set to be 5.5℃. In logging process, this data will be updated to other value that is larger than the old one.

2.3.2.7 summay_min_temperature

Name	summay_min_temperature		
Function	The storage of the minimum temperature data gotten in logging process		
Address	16'hC099		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:10]	-	-	-
[9:0]	min_temperature[9:0]	RW	The minimum temperature data gotten in logging process. A value which is slightly smaller than the center temperature value should be written into this register before starting a logging process. The new measured temperature data will be compared with the value in this register. If the new data is smaller than the old one, the new data will be written to the register automatically. This method can keep the register value as the minimium one. For example, if the temperature range of the

			application is 2℃~8℃ and the accuracy requirement is ±0.5℃, the initial value of the register can be set to be 4.5℃. In logging process, this data will be update to other value that is smaller than the old one.

2.3.2.8 summay_max_limit_cnt

Name	summay_max_limit_cnt		
Function	The times of the logging data beyonds the limit threshold which is defined by Max_alarm_limit.		
Address	16'hC09A		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:0]	max_limit_ cnt[15:0]	R	Counting automatically when the logging data is beyond the high limit value in the logging process.

2.3.2.9 summay_min_limit_cnt

Name	summay_min_limit_cnt		
Function	The times of the logging data beyonds the limit threshold which is defined by Min_alarm_limit.		
Address	16'hC09B		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:0]	min_limit_ cnt[15:0]	R	Counting automatically when the logging data is lower than the high limit value in the logging process.

2.3.2.10 rtc_flow_status

Name	rtc_flow_status		
Function	Indicate the chip's status in logging process		
Address	16'hC094		
Default	0x0000		
Bit	Bit Name	Access	Description
[15:6]	-	-	-
[5:4]	rtc_flow_status[1:0]	R	2'b00：not in logging process 2'b01：in the delay time waiting to start the logging process 2'b10： in logging process 2'b11： RFU
[3:0]	-	-	-

2.4 Commands

2.4.1 ISO15693 commands

2.4.1.1 overview

Command		Description
Mandatory Commands		
0x01	inventory	Refer to ISO/IEC 15693-3
0x02	stay quiet	Refer to ISO/IEC 15693-3
Optional Commands		
0x20	read single block	Refer to ISO/IEC 15693-3
0x21	write single block	Refer to ISO/IEC 15693-3
0x22	lock block	Refer to ISO/IEC 15693-3
0x23	read multi blocks	Refer to ISO/IEC 15693-3
0x25	select	Refer to ISO/IEC 15693-3
0x26	reset to ready	Refer to ISO/IEC 15693-3
0x27	write AFI	Refer to ISO/IEC 15693-3
0x28	lock AFI	Refer to ISO/IEC 15693-3
0x29	write DSFID	Refer to ISO/IEC 15693-3
0x2A	lock DSFID	Refer to ISO/IEC 15693-3
0x2B	get system information	Refer to ISO/IEC 15693-3
0x2C	Get Multiple Block Security Status	Refer to ISO/IEC 15693-3
Custom Commands		
0xB1	read memory	Custom read command of EEPROM. Enormous data can be read in one time by this command whose unit is byte.
0xB2	get random	Used to get the 32bits random number
0xB3	write memory	Custom write command which can access all of the space of EEPROM. The specified authority is up to the operating mode, chip configuration etc. The minimum unit is byte.
0xB4	auth	Controller of the access authority by verify the user password, etc.
0xC0	get temperature	Single measurement command. Two commands need to execute to fulfill one single measurement process. The first command is used to start the single measurement. The second command is used to sendback the measurement result. The interval time between two commands should be longer than 300ms.
0xC2	Start logging/Stop logging	Start or stop the logging process
0xC3	deep sleep	enter PD mode
0xC4	wakeup	Wake up the chip from the PD mode

Command		Description
0xC5	write reg	Write the register
0xC6	read reg	Read the register
0xC9	led ctrl	Switch on or switch off the LED
0xCF	op mode chk	Check the current mode or refresh the register
0xD0	field strength chk	Check the field strength of the current contactless interface
0xCE	Initialize Reg	Initialize the value of the register under battery domin including EPC code of UHF, analog configuration of the UHF interface, etc.

2.4.1.2 Mandatory Commands

2.4.1.2.1 Inventory

SOF	Flags	Inventory	Optional AFI	Mask Length	Mask Value	CRC16	EOF
	8bits	8bits	8bits	8bits	0-64bits	16bits	
		0x01					

As defined in ISO/IEC 15693-3.

2.4.1.2.2 Stay Quiet

SOF	Flags	Stay Quiet	UID	CRC16	EOF
	8bits	8bits	64bits	16bits	
		02			

As defined in ISO/IEC 15693-3.

2.4.1.3 Optional Command

2.4.1.3.1 Read Single Block

SOF	Flags	Read Single Block	Optional UID	Block Address	CRC16	EOF
	8bits	8bits	64bits	8bits	16bits	
		0x20				

As defined in ISO/IEC 15693-3.
This command is used to read data in the user area whose range is up to the configuration. The maximum range is 1k bytes.

2.4.1.3.2 Write Single Block

SOF	Flags	Write Single Block	Optional UID	Block Address	Data	CRC16	EOF
	8bits	8bits	64bits	8bits	32bits	16bits	

		0x21					

As defined in ISO/IEC 15693-3.
This command is used to write data into the user area whose range is up to the configuration. The maximum range is 1k bytes.

2.4.1.3.3 **Lock Block**

SOF	Flags	Lock Block	Optional UID	Block Address	CRC16	EOF
	8bits	8bits	64bits	8bits	16bits	
		0x22				

As defined in ISO/IEC 15693-3.
This command is used to lock the block of the user area.

2.4.1.3.4 **Read Multiple Blocks**

SOF	Flags	Read Multiple Block	Optional UID	First Block Address	Number of Blocks	CRC16	EOF
	8bits	8bits	64bits	8bits	8bits	16bits	
		0x23					

As defined in ISO/IEC 15693-3.
This command is used to read multiple blocks of the user area whose range is up to the configuration. The maximum range is 1k bytes.

2.4.1.3.5 **Select**

SOF	Flags	Select	UID	CRC16	EOF
	8bits	8bits	64bits	16bits	
		0x25			

As defined in ISO/IEC 15693-3.

2.4.1.3.6 **Reset To Ready**

SOF	Flags	Reset To Ready	Optional UID	CRC16	EOF
	8bits	8bits	64bits	16bits	
		0x26			

As defined in ISO/IEC 15693-3.

2.4.1.3.7 **Write AFI**

SOF	Flags	Write AFI	Optional UID	AFI	CRC16	EOF
	8bits	8bits	64bits	8bits	16bits	
		0x27				

As defined in ISO/IEC 15693-3.

2.4.1.3.8 **Lock AFI**

SOF	Flags	Lock AFI	Optional UID	CRC16	EOF
	8bits	8bits	64bits	16bits	
		0x28			

As defined in ISO/IEC 15693-3.

2.4.1.3.9 **Write DSFID**

SOF	Flags	Write DSFID	Optional UID	DSFID	CRC16	EOF
	8bits	8bits	64bits	8bits	16bits	
		0x29				

As defined in ISO/IEC 15693-3.

2.4.1.3.10 **Lock DSFID**

SOF	Flags	Lock DSFID	Optional UID	CRC16	EOF
	8bits	8bits	64bits	16bits	
		0x2A			

As defined in ISO/IEC 15693-3.

2.4.1.3.11 **Get System Information**

SOF	Flags	Get System Information	Optional UID	CRC16	EOF
	8bits	8bits	64bits	16bits	
		0x2B			

As defined in ISO/IEC 15693-3.
This command returns a number of the system parameters, including the UID, DSFID, AFI, memory size, and IC reference shown as follow:

SOF	Flags	Info Flags	UID	DSFID	AFI	Memory size	IC Reference	CRC16	EOF
	8bits	8bits	64bits	8bits	8bits	16bits	8bits	16bits	

VICC Memory Size is 16 bits whose definition is the same as protocol.

MSB 16 – 14	13 – 9	8 – LSB 1
RFU	Block size in bytes	Number of blocks

Table 2-1 VICC memory size information

The number of blocks is up to the configuration whose maximum size is 1kbytes. The block size is fixed to 32bits. So the Block size in bytes is 5'h03.
The high 3 bits of VICC memory size should be padded into 0.

IC Reference is 8 bits whose definitions as shown follow:

IC Reference		
[2]	rtc_status	Logging process status 0: logging process has not been started 1: logging process has been started
[1:0]	chip_work_mode[1:0]	The chip's operation mode: 10: user_mode

2.4.1.3.12 Get Multiple Block Security Status

SOF	Flags	Get Multiple Block Security Status	Optional UID	First Block Address	Number of Blocks	CRC16	EOF
	8bits	8bits	64bits	8bits	8bits	16bits	
		0x2C					

As defined in ISO/IEC 15693-3.

2.4.1.4 Custom Command

2.4.1.4.1 Read Memory

Customed read memory command is used to read the temperature data out of the EEPROM one time.

SOF	Flags	Command code	IC Mfg code	Optional UID	First Address	Number of byte	CRC16	EOF
	8bits	8bits	8bits	64bits	16bits	16bits	16bits	
		0xB1	0x1D		See below table	See below table		

1. UID is optional which is effective only in the addressed mode.
2. Option_flag is not suitable in this command.
3. First Address and Number of byte must increment by blocks (4bytes per block). For example, if 50 bytes datas need to be read from the address 0001h, First address should be set to 0000h, Number of byte should be set to 0034h.
4. If the sum of First address and Number of byte beyond the effective range such as 20k bytes, the error code will be sent back and no data will be read actually.
5. If the First address beyonds the effective range, the error code will be sent back and no data will be read actually.

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	8'h01	0x0F		

If the command format is correct, the response data is shown below:

Response with read authority:

SOF	Flags	Data	CRC16	EOF
	8bits	8bits* (byte_number+4)	16bits	

Response without read authority:

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
	8'h00	0001h		

Note: if the data to be read is in the configuration area and a part of the data is no authority, there is no error code to be sent back and this part of data will be replaced by 0 in the response data.

2.4.1.4.2 Get Random

Get random command is used to get the 32bits random number from the chip. The password that will be transmitted has to be calculated with the password and the random number (see "Auth").

SOF	Flags	Get Random	IC Mfg Code	Optional UID	CRC16	EOF
	8bits	8bits	8bits	64bits	16bits	
		0xB2	0x1D			

Response when error flag set:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Response when error flag not set：

SOF	Flags	Random Number	CRC16	EOF
	8bits	32bits	16bits	
	0x00	Random number, LSB is sent first		

2.4.1.4.3 Write Memory

Customed write memory command is used to write data into all of the area of the EEPROM that has write authority. The minimum unit is byte.

SOF	Flags	Write Memory	IC Mfg code	Optional UID	First Address	Data Number	Data	CRC16	EOF
	8bits	8bits	8bits	64bits	16bits	8bits	8bits* (Data number +1)	16bits	
		0xB3	0x1D						

1. UID is optional which is effective only in the addressed mode.
2. Option_flag=0 and option_flag=1 are supported in this command.
3. First Address is the start address to be written. The unit is byte.

4. Data Number is the length of the data that need to be written into EEPROM. The unit is byte. The maxium length is 4bytes.
5. If the sum of First address and Data number beyond the effective range, the error code will be sent back and no data will be written actually.
6. If the First address beyonds the effective range, the error code will be sent back and no data will be written actually.

The error code of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	8'h01	0x0F		

Response when the command format is correct:

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
	8'h00	See below table		

Para	byte	description	
result	2	LSB is sent back first.	
		[15:5]	RFU
		[4]	ew_energy_chk_result: if this bit is set to "1", it means that the energy for carrying out a write operation is not enough.
		[3]	cmd_wr_len_overflow: if this bit is set to "1", it means that the length is beyond the limit (4bytes).
		[2]	RFU
		[1]	cmd_wr_access: if this bit is set to "1", it means no write authority.
		[0]	RFU

2.4.1.4.4 **Auth**

Auth command is used to verify the password.

SOF	Flags	Auth	IC Mfg code	Optional UID	cmd cfg	PW Data	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	32bits	16bits	
		0xB4	0x1D		See detail description			

Response of command format error：

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

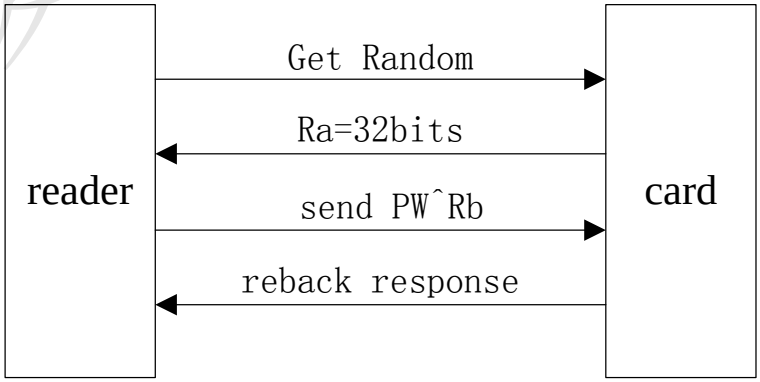
Reponse when command format is correct：

SOF	Flags	Cmd Result	CRC16	EOF
	8bits	16bits	16bits	
	0x00	See detail description		

Detail description：

para	byte	description	
cmd cfg	1	[7:3]	RFU
		[2:0]	Select the type of the password： 000: user area password 011: unlock password 100: stop logging password Others: RFU
PW data	4	32bits password masked by the 32bit random number that obtained by the get random command. For the masked flow, pls refers to the attachment file.	
result	2	[15:8]	RFU
		[7]	auth_pass_flag: “1” means that passed the password verification
		[6]	password_zero: “1” means that the password is 0.
		[5:3]	RFU
		[2:0]	the type of the password: 000: user area password 011: unlock password 100: stop logging password Others: RFU

Password verification flow：



2.4.1.4.5 Get temperature

Get temperature command is used to fulfill a single time measurement of the temperature instantly. Two times of command are needed to complete one measurement. The first command is used to start the measuring process; the second command is used to send back the temperature data. The interval between two commands needs to be greater than 300ms to ensure enough measurement time.

1. When the first command received, the chip can choose to check the contactless field strength before the measurement. Error code will be sent to the reader is the field strength is not enough to fulfill a measurenment successfully.
2. When the fist command received, if the chip is in the RTC logging process, it will check the time interval to the next logging process first. If the time is sufficient to fulfill a single measurement, command will be executed; otherwise, error code will be sent back.
3. When the second command received, the chip will send back the temperature measurement result. You can choose to write the result into the user area.

SOF	Flags	Get Temperature	IC Mfg code	UID	Cmd Cfg	User Area Block Addr	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	8bits	16bits	
		0xC0	0x1D	optional	See detail description			

Response of command format error：

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Reponse when command format is correct：

SOF	Flags	Measurement Result	CRC16	EOF
	8bits	16bits	16bits	
		See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration:	
		[7]	cmd_flow_flag: this bit selects the received command's function. 0: first command, start a temperature measurement 1: second command, get the measurement result
		[6:4]	det_source_sel[2:0]: choose the signal to be measured 000: temperature 001: battery voltage 010: the input voltage signal on the IO pad, such as humidity 011: pressure measurement: a constant current generated in the

para	byte	description	
			chip flows through the out-chip piezoresistor that connected on the IO pad, the chip will measure the voltage on it. other：RFU Note: If the measurement is not temperature, register ana_cfg (0xc021) need to be set first, then set io_pad_cfg=2'b11 in user_cfg0. After that, get temperature command can be send.
		[3]	RFU
		[2]	temperature result type selection: 0: send back the original temperature data which is 13bits 1: send back the transformed data that represent the actual temperature. Note: if the sensor data is not the temperature, this bit must be set to "0".
		[1]	field_chk_en: the field energy check function will be enabled when this bit is set to "1". It is effective only when cmd_flow_flag=0.
		[0]	storage_en: the measurement result will be write into the EEPROM if this bit is set to "1". It is effective only when cmd_flow_flag=1.
User Area Block Addr	1	Select the block address that will store the measurement result.	
Measurement Result	2	LSB is sent first.	
		The first command received to start a measurement. The result is shown below: 16'hfffa: it means the field energy is sufficient for temperature measurement. 16'hfff5: the field energy is not enough 16'hfff0: field energy check function is not enabled	

para	byte	description
		The second command received to send back the result data: temperature result type selection =0： result[15]：store address overflow flag, active high result[14:13]：RFU result[12:0]：the original measurement result data temperature result sel =1： result[15]：store address overflow flag, active high result[14:10]：RFU result[9:0]: transformed data that represent the actual temperature Note: bit9 is the sign bit. bit9=0: the result data is positive number, bit8~bit0 is the absolute value. bit9=1: the result is negative number; the data is saved as complement number. The absolute value is the reverse value of bit8~bit0 and plus 1. The decimal place is start from bit2 or bit1 which is determined by user_cfg0. temp_format_high_preci_en.

2.4.1.4.6 **Start logging**

This command is used to start a temperature logging process.

SOF	Flags	Start logging	IC Mfg code	UID	cmd cfg	RFU	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	32bits	16bits	
		0xC2	0x1D	optional	See detail description			

Response of command format error：

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Reponse when command format is correct：

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
		See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration	
		[7]	Command function selection: 0: start a RTC logging process
		[6:0]	RFU
RFU	4	RFU	
result	2	16'h00 will be sent back if the command is executed successfully	

Note: The initial value of the maximum temperature data and the minimum temperature data should be written into the summary_max_tem and summary_min_tem register if the summary function needs to be used.

2.4.1.4.7 **Stop logging**

This command is used to clear the RTC logging in the logging process.

SOF	Flags	Stop logging	IC Mfg code	UID	cmd cfg	password	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	32bits	16bits	
		0xC2	0x1D	optional	See detail description			

Response of command format error：

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Reponse when command format is correct：

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
		See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration	
		[7]	Command function selection: 1: clear the RTC logging
		[6:0]	RFU
password	4	the function of this parameter is to clear the RTC logging. Before sending this command, password verification should be performed firstly using the Get random command and AUTH command.	

para	byte	description
result		result[15:2]=0 result[1]：password verification not passed，high active. result[0]：the value of Stop logging password is 0，high active

2.4.1.4.8 **Deep Sleep**

The chip will enter PD(power down) mode when Deep sleep command received.

SOF	Flags	Deep Sleep	IC Mfg code	UID	cmd cfg	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	16bits	
		0xC3	0x1D	optional	See detail description		

Response of command format error：

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Reponse when command format is correct：

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
	0x00	16'h0		

Detail description:

para	byte	description	
cmd cfg	1	Command configration	
		[7:1]	RFU
		[0]	Enter PD mode
result	2	default:16'h0	

2.4.1.4.9 **Wake up**

To configure parameters used in the RTC logging process, The chip will quit the PD mode when Wake up command received from HF interface or I2C interface (From UHF interface, the ACK command will do the same job).

Note: Wake up command with option "cmd cfg[7]=0" only could be used before RTC logging process start. Otherwise, it will influence the normal function of UHF interface.

SOF	Flags	Wake Up	IC Mfg code	UID	Cmd Cfg	CRC16	EOF

	8bits	8bits	8bits	64bits	8bits	16bits	
		0xC4	0x1D	optional	See detail description		

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	

Response when command format is correct:

SOF	Flags	Result	CRC16	EOF
	8bits	16bits	16bits	
		See detail description		

Detail description:

Para	byte	description	
cmd cfg	1	Command configuration:	
		[7]	Command function selection: 0: let the chip quit the PD mode 1: check if the chip has quited PD mode successfully
		[6:0]	RFU
result	2	Cmd cfg=0: result=16'h0000 Cmd cfg=1: result=16'h5555: if the chip has quited PD mode successfully result=16'hffff: if the chip has not quited PD mode	

2.4.1.4.10 **Write reg**

Write reg command is used to write information into the RTC relevant register.

SOF	Flags	Write reg	IC Mfg code	UID	Reg Addr	Analog Para	CRC16	EOF
	8bits	8bits	8bits	64bits	16bits	16bits	16bits	
		0xC5	0x1D	optional	See detail desctiption			

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Response when command format is correct:

SOF	Flags	result	CRC16	EOF

	8bits	16bits	16bits	
	8'h0	See detail desctiption		

Detail description:

para	byte	description
reg addr	2	Register's logical address, MSByte is sent first. [15:12]：4'hc [11:8]：RFU [7:0]：register's address
write para	2	The parameter that need to be writen into the register. MSByte is sent first.
result	2	LSByte is sent first. result[15:0]=16'hffff: Reg addr[15:12] is not 4'hc or the chip is in the RTC logging process when write reg command received. result[15:3]='h0 result[2]=1： No write authority to the register address. result[1]=1： The register address is not exist. result[0]=1： RFU result[15:0]=16'h0000：register write successfully

Note: the write to register is not permitted when the chip is in the RTC logging process.

2.4.1.4.11 **Read reg**

Read reg command is used to read the RTC relevant register's value.

SOF	Flags	Read Reg	IC Mfg code	UID	Reg Addr	CRC16	EOF
	8bits	8bits	8bits	64bits	16bits	16bits	
		0xC6	0x1D	optional	See detail desctiption		

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Response when command format is correct:

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
	0x00	See detail desctiption		

Detail description:

para	byte	description
reg addr	2	Register's address, MSByte is sent first [15:12]：4'hc [11:8]：RFU [7:0]：register's address
result	2	result[15:0]: LSByte is sent first result=16'hffff: if the register address is not exist

2.4.1.4.12 **LED ctrl**

Led ctrl is used to control the off chip LED's lighting or not.

SOF	Flags	Led ctrl	IC Mfg code	UID	Cmd Cfg	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	16bits	
		0xC9	0x1D	optional	See detail description		

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Response when command format is correct:

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
	0x00	0x0000		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration	
		[7:2]	RFU
		[1:0]	led_on_en: control the off chip LED's lighting or not by output "0" or "1" signal on the IO_EXT pad. 10: led on 00/01/11: led off
result	2	16'h0000: command is excuted successfully	

2.4.1.4.13 **Op_Mode_Chk**

Op_Mode_Chk is used to check the chip's current operation mode or refresh the temperature logging process's configuration.

SOF	Flags	Op Mode Chk	IC Mfg code	UID	Cmd Cfg	CRC16	EOF
	8bits	8bits	8bits	64bits	24bits	16bits	
		0xCF	0x1D	Optional	See detail description		

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Response when command format is correct:

SOF	Flags	Mode Code	CRC16	EOF
	8bits	16bits	16bits	
		See detail description		

Detail description:

para	byte	description
cmd cfg	3	Cmd_cfg0: 0x00: default value; 0x01: refresh the temperature logging process's configuration. This operation is recommended before sending a Start logging command to start a new logging process.
		Cmd_cfg1, cmd_cfg2: Default value is 0x00. Other value is not recommended.
result	2	MSByte is sent first. result[15:14]：RFU result[13]：user_access_en, active high means that the current user's authority is effective result[12]：RTC logging, active high means that the chip is in RTC logging process. result[11]：vdet_process_flag, active high means a single temperature measurement process has been interrupted. result[10]：RFU result[9]： light_chk_flag, active high means the light strength is beyond the preset threshold value. result[8]： vbat_pwr_flag, active high means the voltage of battery is

		higher than 0.9V. result[7:4]: RFU, keeped to 0 result[3:0]: RFU

2.4.1.4.14 **Field_strength_chk**

Field_strength_chk command is used to measure the HF field strength that help to estimate if the field will affect the temperature measurement's accuracy.

SOF	Flags	Field Strength Chk	IC Mfg code	UID	Cmd Cfg	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	16bits	
		0xD0	0x1D	optional	See detail description		

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Response when command format is correct:

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
	0x00	See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command description	
		[7:1]	RFU
		[0]	field chk sel: select which field to be measured 0: HF(13.56MHz) field 1: RFU
result	2	[15:8]：RFU [7]：field_chk_flag，active high means the measured field strength is effective. [6:4]：RFU [3:0]：field_strength, larger value represents larger field strength.	

2.4.1.4.15 **Initialize Reg**

In the semi-active mode, this command is used to initialize the value of the register under the

battery power domain including EPC code of UHF, analog configuration of the UHF interface, etc. It should be executed first when there is UHF interface and after the battery has been fixed. Otherwise, the UHF interface will not work normally.

SOF	Flags	Init Regfile	IC Mfg code	UID	Cmd Cfg	CRC16	EOF
	8bits	8bits	8bits	64bits	8bits	16bits	
		0xCE		optional	RFU		

Response of command format error:

SOF	Flags	Error Code	CRC16	EOF
	8bits	8bits	16bits	
	0x01	0x0f		

Response when command format is correct:

SOF	Flags	result	CRC16	EOF
	8bits	16bits	16bits	
	0x00	0x0000		

2.4.2 ISO14443 commands

2.4.2.1 Overview

command		description
0x26	reqa	Refer to ISO/IEC 14443-3
0x52	wupa	Refer to ISO/IEC 14443-3
0x93	anti/select level1	Refer to ISO/IEC 14443-3
0x95	anti/select level2	Refer to ISO/IEC 14443-3
0x30	read	Refer to ISO/IEC 14443-3
0x3a	fast read	Refer to ISO/IEC 14443-3
0xa2	write	Refer to ISO/IEC 14443-3
0xa0	compatibility write	Refer to ISO/IEC 14443-3
0xc2	sector select	Refer to ISO/IEC 14443-3
0x50	halt	Refer to ISO/IEC 14443-3
0x40	selfdef_cmd	Customed commands

2.4.2.2 General command

2.4.2.2.1 Reqa

Refer to ISO/IEC 14443-3.

2.4.2.2.2 Wupa

Refer to ISO/IEC 14443-3.

2.4.2.2.3 **Anticollision**

Refer to ISO/IEC 14443-3.

2.4.2.2.4 **Select**

Refer to ISO/IEC 14443-3.

2.4.2.2.5 **Read**

Refer to ISO/IEC 14443-3.

2.4.2.2.6 **Fast read**

Refer to ISO/IEC 14443-3.

2.4.2.2.7 **Write**

Refer to ISO/IEC 14443-3.

2.4.2.2.8 **Compatibility write**

Refer to ISO/IEC 14443-3.

2.4.2.2.9 **Sector select**

Refer to ISO/IEC 14443-3.

2.4.2.2.10 **Halt**

Refer to ISO/IEC 14443-3.

2.4.2.3 Custom command

2.4.2.3.1 **Read memory**

Customed read memory command is used to read the temperature data out of the EEPROM one time.

	SOF	CMD	Read memory	Start address	Read length	Rev.	CRC16	EOF
Of bytes		1	1	2	2	1	2	
		0x40	0xb1	See detail description	See detail description	any value		

1. Start Address and Read length must increment by blocks (4bytes per block). For example, if 50 bytes datas need to be read from the address 0001h, First address should be set to 0000h, Number of byte should be set to 0034h.
2. If the sum of Start Address and Read length beyond the effective range such as 20k bytes, the error code will be sent back and no data will be read actually.

3. If the Start Address beyonds the effective range, the error code will be sent back and no data will be read actually.

Response:

SOF	Result	CRC16	EOF
	(read_len+4)/2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description	
start address	2	Start Address must increment by blocks (4bytes per block).	
read length	2	Read length must increment by blocks (4bytes per block).	
Result	N/2	LSByte is sent first. When Read memory command access an area without read authority, result return 2bytes data, only bit0 is useful When Read memory command access an area with read authority, result return N bytes data, N= read_len+4, the LSB 2bits of read_len will keep 0	
		[0]	cmd_rd_access, active high means no read authority.

Note: if the data to be read is in the configuration area and a part of the data is no authority, there is no error code to be sent back and this part of data will be replaced by 0 in the response data.

2.4.2.3.2 Get random

Get random command is used to get the 32bits random number from the chip. The password that will be transmitted has to be calculated with the password and the random number (see "Auth").

	SOF	CMD	Get random	Rev.	CRC16	EOF
Of bytes		1	1	5	2	
		0x40	0xb2	Any value		

Response:

SOF	Result	CRC16	EOF
	4bytes	2bytes	
	32bits random number, LSByte send first		

2.4.2.3.3 Write memory

Customed write memory command is used to write data into all of the area of the EEPROM that has write authority. The minimum unit is byte.

	SOF	CMD	Write	First	Data	Rev.	Data	CRC16	EOF

			memory	Address	Number				
Of bytes		1	1	2	1	2	Data_num+1	2	
		0x40	0xB3	See detail description	See detail description	Any value	See detail description		

1. First Address is the start address to be written. The unit is byte.
2. Data Number is the length of the data that need to be written into EEPROM. The unit is byte. The maxium length is 4bytes.
3. If the sum of First address and Data number beyond the effective range, the error code will be sent back and no data will be written actually.
4. If the First address beyonds the effective range, the error code will be sent back and no data will be written actually.

Response:

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

Para	Bytes	description	
result	2	LSB is sent back first.	
		[15:5]	RFU
		[4]	ew_energy_chk_result: if this bit is set to “1”, it means that the energy for carrying out a write operation is not enough.
		[3]	cmd_wr_len_overflow: if this bit is set to “1”, it means that the length is beyond the limit (4bytes).
		[2]	RFU
		[1]	cmd_wr_access: if this bit is set to “1”, it means no write authority.
		[0]	RFU

2.4.2.3.4 **Auth**

Auth command is used to verify the password.

	SOF	CMD	auth	cmd cfg	password	CRC16	EOF
Of bytes		1	1	1	4	2	

		0x40	0xB4	See detail description	See detail description		

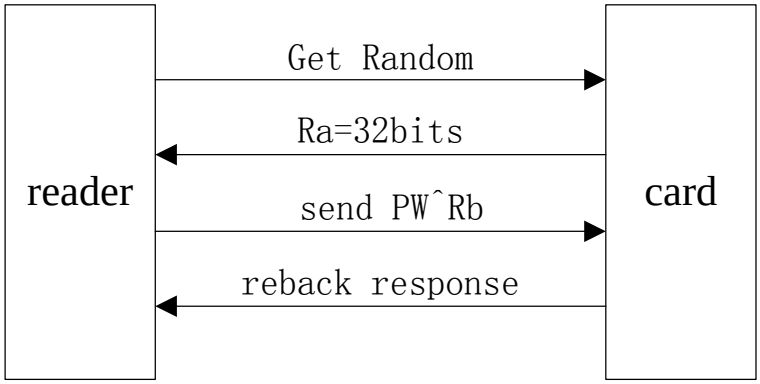
Response:

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration:	
		[7:3]	RFU
		[2:0]	Select the type of the password： 000: user area password 011: unlock password 100: stop logging password Others: RFU
password	4	32bits password masked by the 32bit random number that obtained by the get random command. For the masked flow, pls refers to the attachment file.	
result	2	[15:8]	RFU
		[7]	auth_pass_flag: “1” means that passed the password verification
		[6]	password_zero: “1” means that the password is 0.
		[5:3]	RFU
		[2:0]	the type of the password: 000: user area password 011: unlock password 100: stop logging password Others: RFU

Password verification flow：



2.4.2.3.5 Get temperature

Get temperature command is used to fulfill a single measurement of the temperature instantly. Two times of command are needed to complete one measurement. The first command is used to start the measuring process; the second command is used to send back the temperature data. The interval between two commands needs to be greater than 300ms to ensure enough measurement time.

1. When the first command received, the chip can choose to check the contactless field strength before the measurement. Error code will be sent to the reader is the field strength is not enough to fulfill a measurenment successfully.
2. When the fist command received, if the chip is in the RTC logging process, it will check the time interval to the next logging process first. If the time is sufficient to fulfill a single measurement, command will be executed; otherwise, error code will be sent back.

When the second command received, the chip will send back the temperature measurement result. You can choose to write the result into the user area.

	SOF	CMD	Get temperature	cmd cfg	user block addr	Rev.	CRC16	EOF
Of bytes		1	1	1	1	3	2	
		0x40	0xC0	See detail description	See detail description	Any value		

Response:

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration:	
		[7]	cmd_flow_flag: this bit selects the received command's function. 0: first command, start a temperature measurement 1: second command, get the measurement result
		[6:4]	det_source_sel[2:0]: choose the signal to be measured 000: temperature 001: battery voltage 010: the input voltage signal on the IO pad, such as humidity 011: pressure measurement: a constant current generated in the chip flows through the out-chip piezoresistor that connected on the IO pad, the chip will measure the voltage on it. other：RFU Note: If the measurement is not temperature, register ana_cfg (0xc021) need to be set first, then set io_pad_cfg=2'b11 in user_cfg0. After that, get temperature command can be send.
		[3]	RFU
		[2]	temperature result type selection: 0: send back the original temperature data which is 13bits 1: send back the transformed data that represent the actual temperature. Note: if the sensor data is not the temperature, this bit must be set to "0".
		[1]	field_chk_en: the field energy check function will be enabled when this bit is set to "1". It is effective only when cmd_flow_flag=0.
		[0]	storage_en: the measurement result will be write into the EEPROM if this bit is set to "1". It is effective only when cmd_flow_flag=1.
user block addr	1	Select the block address that will store the measurement result.	
result	2	LSB is sent first.	
		The first command received to start a measurement. The result is shown below: 16'hfffa: it means the field energy is sufficient for temperature measurement. 16'hfff5: the field energy is not enough 16'hfff0: field energy check function is not enabled	

para	byte	description
		The second command received to send back the result data: temperature result type selection =0： result[15]：store address overflow flag, active high result[14:13]：RFU result[12:0]：the original measurement result data temperature result sel =1： result[15]：store address overflow flag, active high result[14:10]：RFU result[9:0]: transformed data that represent the actual temperature Note: bit9 is the sign bit. bit9=0: the result data is positive number, bit8~bit0 is the absolute value. bit9=1: the result is negative number; the data is saved as complement number. The absolute value is the reverse value of bit8~bit0 and plus 1. The decimal place is start from bit2 or bit1 which is determined by user_cfg0. temp_format_high_preci_en.

2.4.2.3.6 **Start logging**

Start logging command used to start a temperature logging process .

	SOF	CMD	Start logging	cmd cfg	Rev.	CRC16	EOF
Of bytes		1	1	1	4	2	
		0x40	0xC2	See detail description	Any value		

Response：

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration	
		[7]	Command function selection: 0: start a RTC logging process
		[6:0]	RFU
Rev.	4	RFU	
result	2	16'h00 will be sent back if the command is executed successfully	

Note: The initial value of the maximum temperature data and the minimum temperature data should be written into the summary_max_tem and summary_min_tem register if the summary function needs to be used.

2.4.2.3.7 **Stop logging**

Start logging command used to clear the RTC logging in the logging process.

	SOF	CMD	Stop logging	cmd cfg	password	CRC16	EOF
Of bytes		1	1	1	4	2	
		0x40	0xC2	See detail description	See detail description		

Response:

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration	
		[7]	Command function selection: 1: clear the RTC logging
		[6:0]	RFU
password	4	the function of this parameter is to clear the RTC logging. Before sending this command, password verification should be performed firstly using the Get random command and AUTH command.	
Result		result[15:2]=0 result[1]: password verification not passed, high active.	

para	byte	description
		result[0]：the value of Stop logging password is 0，high active

2.4.2.3.8 **Deep sleep**

The chip will enter PD(power down) mode when Deep sleep command received.

	SOF	CMD	Deep sleep	cmd cfg	Rev.	CRC16	EOF
Of bytes		1	1	1	4	2	
		0x40	0xC3	See detail description	Any value		

Response：

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configration	
		[7:1]	RFU
		[0]	Enter PD mode
result	2	default:16’h0	

2.4.2.3.9 **Wake up**

To configure parameters used in the RTC logging process, the chip will quit the PD mode when received Wake up command from HF interface (From UHF interface, the ACK command will do the same job).

Note: Wake up command with option "cmd cfg[7]=0" only could be used before RTC logging process start. Otherwise, it will influence the normal function of UHF interface.

	SOF	CMD	Wakeup	cmd cfg	Rev.	CRC16	EOF
Of bytes		1	1	1	4	2	
		0x40	0xC4	See detail description	Any value		

Response：

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

Para	byte	description	
cmd cfg	1	Command configuration:	
		[7]	Command function selection: 0: let the chip quit the PD mode 1: check if the chip has quited PD mode successfully
		[6:0]	RFU
result	2	Cmd cfg=0: result=16'h0000 Cmd cfg=1: result=16'h5555 if the chip has quited PD mode successfully result=16'hffff if the chip has not quited PD mode	

2.4.2.3.10 **Write reg**

Write reg command is used to write information into the RTC relevant register.

	SOF	CMD	Write reg	write reg addr	write reg data	Rev.	CRC 16	EOF
Of bytes		1	1	2	2	1	2	
		0x40	0xC5	See detail description	See detail description	Any value		

Response：

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description：

para	byte	description
reg addr	2	Register's logical address, MSByte is sent first. [15:12]：4'hc [11:8]：RFU [7:0]：register's address

para	byte	description
write data	2	The parameter that need to be writen into the register. MSByte is sent first.
result	2	LSByte is sent first. result[15:0]=16'hffff: Reg addr[15:12] is not 4'hc or the chip is in the RTC logging process when write reg command received. result[15:3]='h0 result[2]=1： No write authority to the register address. result[1]=1： The register address is not exist. result[0]=1： RFU result[15:0]=16'h0000：register write successfully

Note: the write to register is not permitted when the chip is in the RTC logging process.

2.4.2.3.11 **Read reg**

Read reg command is used to read the RTC relevant register's value.

	SOF	CMD	Read reg	Read reg addr	Rev.	CRC16	EOF
Of bytes		1	1	2	3	2	
		0x40	0xC6	See detail description	Any value		

Response：

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description：

para	byte	description
reg addr	2	Register's address, MSByte is sent first [15:12]：4'hc [11:8]：RFU [7:0]：register's address
Result	2	result[15:0]: LSByte is sent first result=16'hffff: if the register address is not exist

2.4.2.3.12 LED ctrl

Led ctrl command is used to control the off chip LED's lighting or not.

	SOF	CMD	Led ctrl	cmd cfg	Rev.	CRC16	EOF
Of bytes		1	1	1	4	2	
		0x40	0xC9	See detail description	Any value		

Response:

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description	
cmd cfg	1	Command configuration	
		[7:2]	RFU
		[1:0]	led_on_en: control the off chip LED's lighting or not by output "0" or "1" signal on the IO_EXT pad. 10: led on 00/01/11: led off
result	2	16'h0000: command is excuted successfully	

2.4.2.3.13 Op_Mode_Chk

Op_Mode_Chk is used to check the chip's current operation mode or refresh the temperature logging process's configuration.

	SOF	CMD	Op_mode_chk	cmd cfg	Rev.	CRC16	EOF
Of bytes		1	1	3	2	2	
		0x40	0xCF	RFU	Any value		

Response:

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description:

para	byte	description
cmd cfg	3	Cmd_cfg0: 0x00: default value; 0x01: refresh the temperature logging process's configuration. This operation is recommended before sending a Start logging command to start a new logging process.
		Cmd_cfg1, cmd_cfg2: Default value is 0x00. Other value is not recommended.
result	2	MSByte is sent first. result[15:14]：RFU result[13]：user_access_en, active high means that the current user's authority is effective result[12]：RTC logging, active high means that the chip is in RTC logging process. result[11]：vdet_process_flag, active high means a single temperature measurement process has been interrupted. result[10]：RFU result[9]： light_chk_flag, active high means the light strength is beyond the preset threshold value. result[8]： vbat_pwr_flag, active high means the voltage of battery is higher than 0.9V. result[7:4]: RFU, keeped to 0 result[3:0]: RFU

2.4.2.3.14 Field_strength_chk

Field_strength_chk command is used to measure the HF field strength that helps to estimate if the field will affect the temperature measurement's accuracy.

	SOF	CMD	Led ctrl	cmd cfg	Rev.	CRC16	EOF
Of bytes		1	1	1	4	2	
		0x40	0xD0	See detail description	Any value		

Response：

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	See detail description		

Detail description：

para	byte	description	
cmd cfg	1	Command description	
		[7:1]	RFU
		[0]	field chk sel: select which field to be measured 0: HF(13.56MHz) field 1: RFU
result	2	[15:8]：RFU [7]：field_chk_flag，active high means the measured field strength is effective. [6:4]：RFU [3:0]：field_strength, larger value represents larger field strength.	

2.4.2.3.15 Initialize Reg

In the semi-active mode, this command is used to initialize the value of the register under the battery power domain including EPC code of UHF, analog configuration of the UHF interface, etc. It should be executed first when there is UHF interface and after the battery has been fixed. Otherwise, the UHF interface will not work normally.

	SOF	CMD	Init regfile	cmd cfg	Rev.	CRC16	EOF
Of bytes		1	1	1	4	2	
		0x40	0xCE	RFU	Any value		

Response：

SOF	Result	CRC16	EOF
	2bytes	2bytes	
	0x0000		

2.4.3 I2C commands

2.4.3.1 Overview

Command code	Command name	Function description
0xb1	Read memory	Custom read command of EEPROM. Enormous data can be read in one time by this command whose unit is byte.
0xb3	Write memory	Custom write command which can access all of the space of EEPROM. The specified authority is up to the operating mode, chip configuration etc. The minimum unit is byte.
0xc0	Get temperature	Single measurement command. Two commands need to execute to fulfill one single measurement process. The

Command code	Command name	Function description
		first command is used to start the single measurement. The second command is used to sendback the measurement result. The interval time between two commands should be longer than 300ms.
0xc2	Start logging	Start the logging process
0xc3	Deep sleep	enter PD mode
0xc4	Wake up	Wake up the chip from the PD mode
0xc5	Write reg	Write the register
0xc6	Read reg	Read the register
0xd0	Op mode chk	Check the current mode or refresh the register

2.4.3.2 Custom command

2.4.3.2.1 Read memory

Start	Addr(W)	Cmd code	Start address	Read length	Crc16	Stop	Dealy
	1byte	1byte	2bytes	2bytes	2bytes		
		0xB1					100us

Command without error

Start	Addr(R)	Right flag	Data for read	CRC 16	Stop
	1byte	1byte	Length bytes	2bytes	

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

1. Start Address and Read length must increment by blocks (4bytes per block). For example, if 50 bytes datas need to be read from the address 0001h, First address should be set to 0000h, Number of byte should be set to 0034h.
2. If the sum of Start Address and Read length beyond the effective range such as 20k bytes, the error code will be sent back and no data will be read actually.
3. If the Start Address beyonds the effective range, the error code will be sent back and no data will be read actually.

Description：

Para	byte	description
address	1	Slave address

Para	byte	description	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
start address	2	Start Address must increment by blocks (4bytes per block).	
read length	2	Read length must increment by blocks (4bytes per block).	
right flag	1	Flag=8'h00 if no error	
data for read	length	length+4 bytes data will be sent, LSByte first.	
error result	2	[15:1]	RFU
		[0]	cmd_rd_access，active high means no read authority
error flag	1	Command check error	
		[7:1]	RFU
		[0]	Active high means CRC error

2.4.3.2.2 **Write memory**

Start	Addr(W)	Cmd code	Start address	Write length	Write data	Crc16	Stop	Dealy
	1byte	1byte	2bytes	1bytes	Length byte	2bytes		
		0xB3						10ms

Command without error

Start	Addr(R)	Right flag	Result	CRC 16	Stop
	1byte	1byte	2 bytes	2bytes	

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

1. Start Address is the start address to be written. The unit is byte.
2. Write length is the length of the data that need to be written into EEPROM. The unit is byte. The maxium length is 4bytes.
3. If the sum of First address and Data number beyond the effective range, the error code will be sent back and no data will be written actually.

4. If the First address beyonds the effective range, the error code will be sent back and no data will be written actually.

Description：

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
start address	2	Write start logical address, MSByte send first.	
write length	1	Data's byte length, write_length+1, The maxium length is 4bytes.	
write data	length	Data to be writen	
right flag	1	Flag=8'h00 if no error	
result	2	[15:5]	RFU
		[4]	ew_energy_chk_result: if this bit is set to "1", it means that the energy for carrying out a write operation is not enough.
		[3]	cmd_wr_len_overflow: if this bit is set to "1", it means that the length is beyond the limit (4bytes).
		[2]	RFU
		[1]	cmd_wr_access: if this bit is set to "1", it means no write authority.
		[0]	RFU
error flag	1	Command check error	
		[7:1]	RFU
		[0]	Active high means CRC error

2.4.3.2.3 **Get temperature**

Get temperature command is used to fulfill a single measurement of the temperature instantly. Two times of command are needed to complete one measurement. The first command is used to start the measuring process; the second command is used to send back the temperature data. The interval between two commands needs to be greater than 300ms to ensure enough measurement time.

1. When the first command received, the chip can choose to check the contactless field strength before the measurement. Error code will be sent to the reader is the field strength is not enough to fulfill a measurenment successfully.

2. When the fist command received, if the chip is in the RTC logging process, it will check the time interval to the next logging process first. If the time is sufficient to fulfill a single measurement, command will be executed; otherwise, error code will be sent back.

When the second command received, the chip will send back the temperature measurement result. You can choose to write the result into the user area.

Start	Addr(W)	Cmd code	Cmd cfg	ew block addr	Crc16	Stop	Dealy time
	1byte	1byte	1byte	1byte	2bytes		
		0xC0					See discription

Command without error

Start	Addr(R)	Right flag	Result	CRC 16	Stop
	1byte	1byte	2 bytes	2bytes	

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

Description:

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
cmd cfg	1	[7]	cmd_flow_flag: this bit selects the received command's function. 0: first command, start a temperature measurement 1: second command, get the measurement result
		[6:4]	det_source_sel[2:0]: choose the signal to be measured 000: temperature 001: battery voltage 010: the input voltage signal on the IO pad, such as humidity 011: pressure measurement: a constant current generated in the chip flows through the out-chip piezoresistor that connected on the IO pad, the chip will measure the voltage on it.

Para	byte	description	
			other：RFU Note: If the measurement is not temperature, register ana_cfg (0xc021) need to be set first, then set io_pad_cfg=2'b11 in user_cfg0. After that, get temperature command can be send.
		[3]	RFU
		[2]	temperature result type selection: 0: send back the original temperature data which is 13bits 1: send back the transformed data that represent the actual temperature. Note: if the sensor data is not the temperature, this bit must be set to "0".
		[1]	field_chk_en: the field energy check function will be enabled when this bit is set to "1". It is effective only when cmd_flow_flag=0.
		[0]	storage_en: the measurement result will be write into the EEPROM if this bit is set to "1". It is effective only when cmd_flow_flag=1.
ew block addr	1	Select the block address that will store the measurement result.	
delay time	-	Start measurement： field_chk_en=1: delay_time=200us field_chk_en=0: delay_time=100us send back result： storge_en=1: delay_time is up to the write and erase time of the EEPROM storge_en=0: delay_time=100us	
right flag	1	Flag=8'h00 if no error	
result	2	LSB is sent first.	
		The first command received to start a measurement. The result is shown below: 16'hfffa: it means the field energy is sufficient for temperature measurement. 16'hfff5: the field energy is not enough 16'hfff0: field energy check function is not enabled	
		The second command received to send back the result data: temperature result type selection =0： result[15]：store address overflow flag, active high	

Para	byte	description
		result[14:13]：RFU result[12:0]：the original measurement result data temperature result sel =1： result[15]：store address overflow flag, active high result[14:10]：RFU result[9:0]: transformed data that represent the actual temperature Note: bit9 is the sign bit. bit9=0: the result data is positive number, bit8~bit0 is the absolute value. bit9=1: the result is negative number; the data is saved as complement number. The absolute value is the reverse value of bit8~bit0 and plus 1. The decimal place is start from bit2 or bit1 which is determined by user_cfg0. temp_format_high_preci_en.
error flag	1	Command check error
		[7:1] RFU
		[0] Active high means CRC error

2.4.3.2.4 **Start logging**

Start logging command used to start a temperature logging process.

Start	Addr(W)	Cmd code	Cmd cfg	Crc16	Stop	Delay time
	1byte	1byte	1byte	2bytes		
		0xC2				See discription

Command without error

Start	Addr(R)	Right flag	Result	CRC 16	Stop
	1byte	1byte	2 bytes	2bytes	

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

Description：

Para	byte	description

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
cmd cfg	1	[7]	Command function selection: 0: start a RTC logging process
		[6:0]	RFU
delay time	-	Delay_time is relative to the EEPROM's write and erase time because RTC logging need to be written.	
right flag	1	Flag=8'h00 if no error	
result	1	Send back 16'h00 if the command executes successfully	
error flag	1	Command check error	
		[7:1]	RFU
		[0]	Active high means CRC error

2.4.3.2.5 **Stop logging**

Start logging command used to clear a temperature logging process.

Start	Addr(W)	Cmd code	Cmd cfg	Crc16	Stop	Delay time
	1byte	1byte	1byte	2bytes		
		0xC2				100us

Command without error

Start	Addr(R)	Right flag	Result	CRC 16	Stop
	1byte	1byte	2 bytes	2bytes	

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

Description：

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address

Para	byte	description	
		[0]	Selection between write and read: 0: write 1: read
cmd cfg	1	[7]	Command function selection: 1: clear the RTC logging
		[6:0]	RFU
right flag	1	Flag=8'h00 if no error	
result	1	result[15:2]=0 result[1]：password verification not passed，high active. result[0]：the value of Stop logging password is 0，high active	
error flag	1	Command check error	
		[7:1]	RFU
		[0]	Active high means CRC error

2.4.3.2.6 **Deep sleep**

The chip will enter PD(power down) mode when Deep sleep command received.

Start	Addr(W)	Cmd code	Cmd cfg	Crc16	Stop	Dealy time
	1byte	1byte	1byte	2bytes		
		0xC3				100us

Command without error

Start	Addr(R)	Right flag	Result	CRC 16	Stop
	1byte	1byte	2 bytes	2bytes	

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

Description：

Para	byte	description

Para	byte	description	
Address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
cmd cfg	1	[7:1]	RFU
		[0]	Enter PD mode
delay time	-	100us	
right flag	1	Flag=8'h00 if no error	
error flag	1	Command check error	
		[7:1]	RFU
		[0]	Active high means CRC error
result	2	result[15:0]=16'h0000	

2.4.3.2.7 **Wake up**

To configure parameters used in the RTC logging process, The chip will quit the PD mode when Wake up command received from I2C interface (From UHF interface, the ACK command will do the same job).
Note: Wake up command with option "cmd cfg[7]=0" only could be used before RTC logging process start. Otherwise, it will influence the normal function of UHF interface.

	Addr(W)	Cmd code	Cmd cfg	Crc16		Dealy time
Start	1byte	1byte	1byte	2bytes	Stop	
		0xC4				100us

Command without error

	Addr(R)	Right flag	Result	CRC 16	
Start	1byte	1byte	2 bytes	2bytes	Stop

Command with error

	Addr(R)	Error flag	CRC16	
Start	1byte	1byte	2bytes	Stop

Description：

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
cmd cfg	1	[7]	Command function selection: 0: let the chip quit the PD mode 1: check if the chip has quited PD mode successfully
		[6:0]	RFU
delay time	-	100us	
result	2	Cmd cfg=0: result=16'h0000 Cmd cfg=1: result=16'h5555 if the chip has quited PD mode successfully result=16'hffff if the chip has not quited PD mode	
right flag	1	Flag=8'h00 if no error	
error flag	1	Command check error	
		[7:1]	RFU
		[0]	Active high means CRC error

2.4.3.2.8 **Write reg**

Write reg command is used to write information into the RTC relevant register.

	Addr(W)	Cmd code	Reg addr	Write data	Crc16		Dealy time
Start	1byte	1byte	2bytes	2bytes	2bytes	Stop	
		0xC5					100us

Command without error

	Addr(R)	Right flag	Result	CRC 16	
Start	1byte	1byte	2 bytes	2bytes	Stop

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

Description：

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
reg addr	2	Register's logical address, MSByte is sent first. [15:12]：4'hc [11:8]：RFU [7:0]：register's address	
write data	2	The parameter that need to be writen into the register. MSByte is sent first.	
delay time	-	100us	
right flag	1	Flag=8'h00 if no error	
result	2	LSByte is sent first. result[15:0]=16'hffff: Reg addr[15:12] is not 4'hc or the chip is in the RTC logging process when write reg command received. result[15:3]='h0 result[2]=1： No write authority to the register address. result[1]=1： The register address is not exist. result[0]=1： RFU result[15:0]=16'h0000：register write successfully	
error flag	1	Command check error	
		[7:1]	RFU
		[0]	Active high means CRC error

2.4.3.2.9 **Read_reg**

Read reg command is used to read the RTC relevant register's value.

Start	Addr(W)	Cmd code	Reg addr	Crc16	Stop	Dealy time
	1byte	1byte	2byte	2bytes		
		0xC6				100us

Command without error

Start	Addr(R)	Right flag	Result	CRC 16	Stop
	1byte	1byte	2 bytes	2bytes	

Command with error

Start	Addr(R)	Error flag	CRC16	Stop
	1byte	1byte	2bytes	

Description：

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
reg addr	2	Register's logical address, MSByte is sent first. [15:12]：4'hc [11:8]：RFU [7:0]：register's address	
delay time	-	100us	
right flag	1	Flag=8'h00 if no error	
result	2	result[15:0]: LSByte is sent first result=16'hffff: if the register address is not exist	
error flag	1	Command check error	
		[7:1]	RFU

		[0]	Active high means CRC error

2.4.3.2.10 **Op_mode_chk**

Op_Mode_Chk is used to check the chip's current operation mode or refresh the temperature logging process's configuration.

	Addr(W)	Cmd code	Cmd cfg	Crc16		Dealy time
Start	1byte	1byte	3byte	2bytes	**Stop**	
		0xCF				See discription

Command without error

	Addr(R)	Right flag	Result	CRC 16	
Start	1byte	1byte	2 bytes	2bytes	**Stop**

Command with error

	Addr(R)	Error flag	CRC16	
Start	1byte	1byte	2bytes	**Stop**

Description：

Para	byte	description	
address	1	Slave address	
		[7:1]	7bit slave address
		[0]	Selection between write and read: 0: write 1: read
cmd cfg	3	Cmd_cfg0: 0x00: default value; 0x01: refresh the temperature logging process's configuration. This operation is recommended before sending a Start logging command to start a new logging process.	
		Cmd_cfg1, cmd_cfg2:	

Para	byte	description
		Default value is 0x00. Other value is not recommended.
delay time	-	cmd_cfg0[0]=0: delay_time=100us cmd_cfg0[0]=1: delay_time=1.2ms
right flag	1	Flag=8'h00 if no error
result	2	MSByte is sent first. result[15:14]：RFU result[13]：user_access_en, active high means that the current user's authority is effective result[12]：RTC logging, active high means that the chip is in RTC logging process. result[11]：vdet_process_flag, active high means a single temperature measurement process has been interrupted. result[10]：RFU result[9]： light_chk_flag, active high means the light strength is beyond the preset threshold value. result[8]： vbat_pwr_flag, active high means the voltage of battery is higher than 0.9V. result[7:4]: RFU result[3:0]: RFU
error flag	1	Command check error
		[7:1] RFU
		[0] Active high means CRC error

2.4.3.2.11 **initialize Reg**

In the semi-active mode, this command is used to initialize the value of the register under the battery power domain including EPC code of UHF, analog configuration of the UHF interface, etc. It should be executed first when there is UHF interface and after the battery has been fixed. Otherwise, the UHF interface will not work normally.

S	Addr(W)	Cmd code	Cmd cfg	Crc16	E	Dealy time
	1byte	1byte	1byte	2bytes		
		0xCE	Any value			

Command without error

S	Addr(R)	Right flag	Result	CRC 16	E
	1byte	1byte	2 bytes	2bytes	

Command with error

S	Addr(R)	Error flag	CRC16	E

	1byte	1byte	2bytes	

2.4.4 ISO18000-6C commands

2.4.4.1 Overview

command		description
4'hA	select	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
4'h8	Query	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
4'h9	Query adjust	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
2'h0	Queryreq	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
2'h1	Ack	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc0	Nak	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc1	Req_rn	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc2	Read	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc3	Write	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc5	Lock	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc6	Access	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc7	Blockwrite	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
8'hc8	Blockerase	Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0
16'he0xx	Custom command	See the detail description

2.4.4.2 Mandatory Commands

2.4.4.2.1 Select

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.2 Query

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.3 QueryAdjust

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.4 QueryRep

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.5 ACK

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.6 NAK

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.7 **Req_RN**

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.8 **Read**

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.9 **Write**

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.2.10 **Lock**

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.3 Optional command

2.4.4.3.1 **Access**

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.3.2 **BlockWrite**

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.3.3 **BlockErase**

Refer to ISO/IEC18000-6C or EPC Global Class1 Gen2 V1.2.0

2.4.4.4 Custom command

2.4.4.4.1 **Read Memory**

Customed read memory command is used to read the temperature data out of the EEPROM one time.

	Command	Start address	Read length	RN	CRC-16
#of bits	16	16	16	16	16
description	16'he000	See detail description	See detail description	Handle	

1. First Address and Number of byte must increment by blocks (4bytes per block). For example, if 50 bytes datas need to be read from the address 0001h, First address should be set to 0000h, Number of byte should be set to 0034h.
2. If the sum of First address and Number of byte beyond the effective range such as 20k bytes, the error code will be sent back and no data will be read actually.
3. If the First address beyonds the effective range, the error code will be sent back and no data will be read actually.

Response with authority:

	Header	Memory Words	RN	CRC-16
# of bits	1	Variable	16	16
description	0	Data	Handle	

Response without authority:

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	1	See detail description	Handle	

Detail description:

para	byte	description	
start address	2	Start Address must increment by blocks (4bytes per block).	
read length	2	Read length must increment by blocks (4bytes per block). the actual read length should be "read length + 4" LSB 2bits always keep 0	
result	2	LSByte is sent first.	
		[0]	cmd_rd_access, active high means no read authority.

Note: if the data to be read is in the configuration area and a part of the data is no authority, there is no error code to be sent back and this part of data will be replaced by 0 in the response data.

2.4.4.4.2 **Write Memory**

Customed write memory command is used to write data into all of the area of the EEPROM that has write authority. The minimum unit is byte.

	Command	Start address	Data Length	Data	RN	CRC-16
#of bits	16	16	16	8*(Data length+1)	16	16
description	16'he001	See detail description	See detail description	See detail description	Handle	

1. Start Address is the start address to be written. The unit is byte.
2. Data length is the length of the data that need to be written into EEPROM. The unit is byte. The maxium length is 4bytes.
3. If the sum of Start Address and Data length beyond the effective range, the error code will be sent back and no data will be written actually.
4. If the Start Address beyonds the effective range, the error code will be sent back and no data will be written actually.

Response with authority:

	Header	Result	RN	CRC-16
# of bits	1	16	16	16

description	0	See detail description	Handle	

Response without authority:

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	1	See detail description	Handle	

Detail description:

para	byte	Description	
start address	2	Write start logical address, MSByte send first.	
Data length	2	Data's byte length, will transfer "data_length+1" bytes For UHF interface, The maxium length is 4bytes. so the data length only need to define the first byte, and keep the second byte be 8'h00	
data	length	Data to be written	
result	2	LSByte send first	
		[15:5]	RFU
		[4]	ew_energy_chk_result: if this bit is set to "1", it means that the energy for carrying out a write operation is not enough.
		[3]	cmd_wr_len_overflow: if this bit is set to "1", it means that the length is beyond the limit (4bytes).
		[2]	RFU
		[1]	cmd_wr_access: if this bit is set to "1", it means no write authority.
		[0]	RFU

2.4.4.4.3 **Auth**

Auth command is used to verify the password.

	Command	CMD Cfg	password	RN	CRC-16
#of bits	16	8	32	16	16
description	16'he002	See detail description	See detail description	Handle	

Response:

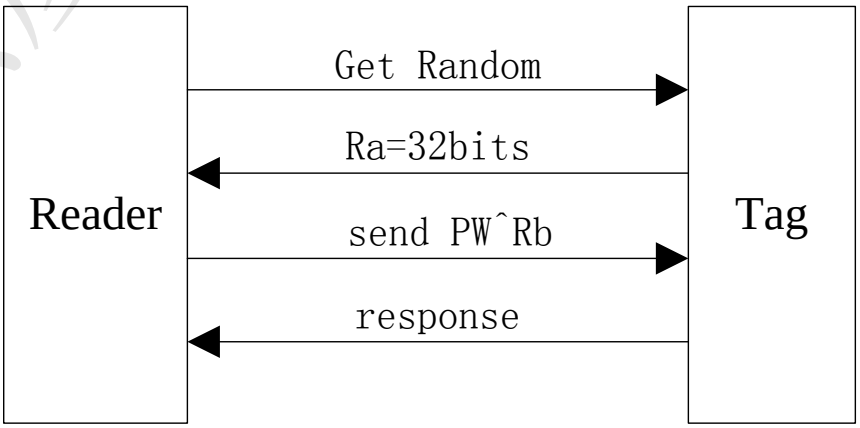
	Header	Result	RN	CRC-16

# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description:

para	byte	description	
cmd cfg	1	Command configuration:	
		[7:3]	RFU
		[2:0]	Select the type of the password： 000: user area password 011: unlock password 100: stop logging password Others: RFU
Password	4	32bits password masked by the 32bit random number that obtained by the get random command. For the masked flow, pls refers to the attachment file.	
result	2	[15:8]	RFU
		[7]	auth_pass_flag: “1” means that passed the password verification
		[6]	password_zero: “1” means that the password is 0.
		[5:3]	RFU
		[2:0]	the type of the password: 000: user area password 011: unlock password 100: stop logging password Others: RFU

Password verification flow：



2.4.4.4.4 GetTemperature

Get temperature command is used to fulfill a single measurement of the temperature instantly. Two times of command are needed to complete one measurement. The first command is used to start the measuring process; the second command is used to send back the temperature data. The interval between two commands needs to be greater than 300ms to ensure enough measurement time.

1. When the first command received, the chip can choose to check the contactless field strength before the measurement. Error code will be sent to the reader is the field strength is not enough to fulfill a measurenment successfully.
2. When the fist command received, if the chip is in the RTC logging process, it will check the time interval to the next logging process first. If the time is sufficient to fulfill a single measurement, command will be executed; otherwise, error code will be sent back.

When the second command received, the chip will send back the temperature measurement result. You can choose to write the result into the user area.

	Command	cmd Cfg	Store Block Address	RN	CRC-16
#of bits	16	8	8	16	16
description	16'h e003	See detail description	See detail description	Handle	

Response:

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description：

para	byte	description	
cmd cfg	1	Command configuration	
		[7]	cmd_flow_flag: this bit selects the received command's function. 0: first command, start a temperature measurement 1: second command, get the measurement result
		[6:4]	det_source_sel[2:0]: choose the signal to be measured 000: temperature 001: battery voltage 010: the input voltage signal on the IO pad, such as humidity 011: pressure measurement: a constant current generated in the chip flows through the out-chip piezoresistor that connected on the IO pad, the chip will measure the voltage on it.

para	byte	description	
			other：RFU Note: If the measurement is not temperature, register ana_cfg (0xc021) need to be set first, then set io_pad_cfg=2'b11 in user_cfg0. After that, get temperature command can be send.
		[3]	RFU
		[2]	temperature result type selection: 0: send back the original temperature data which is 13bits 1: send back the transformed data that represent the actual temperature. Note: if the sensor data is not the temperature, this bit must be set to "0".
		[1]	field_chk_en: the field energy check function will be enabled when this bit is set to "1". It is effective only when cmd_flow_flag=0.
		[0]	storage_en: the measurement result will be write into the EEPROM if this bit is set to "1". It is effective only when cmd_flow_flag=1.
Store Block Addr	1	Select the block address that will store the measurement result.	
result	2	MSB is sent first.	
		The first command received to start a measurement. The result is shown below: 16'hfffa: it means the field energy is sufficient for temperature measurement. 16'hfff5: the field energy is not enough 16'hfff0: field energy check function is not enabled	
		The second command received to send back the result data: temperature result type selection =0： result[15]：store address overflow flag, active high result[14:13]：RFU result[12:0]：the original measurement result data temperature result sel =1： result[15]：store address overflow flag, active high result[14:10]：RFU result[9:0]: transformed data that represent the actual temperature	

para	byte	description
		Note: bit9 is the sign bit. bit9=0: the result data is positive number, bit8~bit0 is the absolute value. bit9=1: the result is negative number; the data is saved as complement number. The absolute value is the reverse value of bit8~bit0 and plus 1. The decimal place is start from bit2 or bit1 which is determined by user_cfg0. temp_format_high_preci_en.

2.4.4.4.5 **Start logging**

Start logging command used to start a temperature logging process

	Command	cmd Cfg	RFU	RN	CRC-16
#of bits	16	8	32	16	16
description	16'h e006	See detail description	Any value	Handle	

Response:

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description：

para	byte	description	
cmd cfg	1	Command configuration	
		[7]	Command function selection: 0: start a RTC logging process
		[6:0]	RFU
result	2	16'h00 will be sent back if the command is executed successfully	

Note: The initial value of the maximum temperature data and the minimum temperature data should be written into the summary_max_tem and summary_min_tem register if the summary function needs to be used.

2.4.4.4.6 **Stop logging**

Start logging command used to clear the RTC logging in the logging process.

	Command	cmd Cfg	Password	RN	CRC-16
#of bits	16	8	32	16	16
description	16'h e006	See detail description	See detail description	Handle	

Response:

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description：

para	byte	description	
cmd cfg	1	Command configuration	
		[7]	Command function selection: 1: clear the RTC logging
		[6:0]	RFU
Password	4	the function of this parameter is to clear the RTC logging. Before sending this command, password verification should be performed firstly using the Get random command and AUTH command.	
result	2	result[15:2]=0 result[1]：password verification not passed，high active. result[0]：the value of Stop logging password is 0，high active	

2.4.4.4.7 **WriteReg**

Write reg command is used to write information into the RTC relevant register.

	Command	Reg addr	Reg write data	RN	CRC-16
#of bits	16	16	16	16	16
description	16'he008	See detail description	See detail description	Handle	

Response：

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description：

para	byte	description
reg addr	2	Register's logical address, MSByte is sent first. [15:12]：4'hc [11:8]：RFU [7:0]：register's address
write data	2	The parameter that need to be writen into the register. MSByte is sent first.
result	2	LSByte is sent first. result[15:0]=16'hffff: Reg addr[15:12] is not 4'hc or the chip is in the RTC logging process when write reg command received. result[15:3]='h0 result[2]=1： No write authority to the register address. result[1]=1： The register address is not exist. result[0]=1： RFU result[15:0]=16'h0000：register write successfully

Note: the write to register is not permitted when the chip is in the RTC logging process.

2.4.4.4.8 **ReadReg**

Read reg command is used to read the RTC relevant register's value.

	Command	Reg addr	RN	CRC-16
#of bits	16	16	16	16
description	16'he009	See detail description	Handle	

Response：

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description：

para	byte	description
reg addr	2	Register's address, MSByte is sent first [15:12]：4'hc

para	byte	description
		[11:8]：RFU [7:0]：register's address
result	2	result[15:0]: LSByte is sent first result=16'hffff: if the register address is not exist

2.4.4.4.9 **DeepSleep**

The chip will enter PD(power down) mode when Deep sleep command received.

	Command	cmd Cfg	RN	CRC-16
#of bits	16	8	16	16
description	16'he00a	See detail description	Handle	

Response：

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description：

para	byte	description	
cmd cfg	1	Command configration	
		[7:1]	RFU
		[0]	Enter PD mode
result	2	default:16'h0	

2.4.4.4.10 **Op_Mode_Chk**

Op_Mode_Chk is used to check the chip's current operation mode or refresh the temperature logging process's configuration.

	Command	cmd Cfg	RN	CRC-16
#of bits	16	24	16	16
description	16'he00c	See detail description	Handle	

Response：

	Header	Result	RN	CRC-16
# of bits	1	16	16	16

description	0	See detail description	Handle	

Detail description：

para	byte	description
cmd cfg0	1	0x00: default value; 0x01: refresh the temperature logging process's configuration. This operation is recommended before sending a Start logging command to start a new logging process.
cmd cfg1	1	Default value is 0x00. Other value is not recommended.
cmd cfg2	1	Default value is 0x00. Other value is not recommended.
result	2	MSByte is sent first. result[15:14]：RFU result[13]：user_access_en, active high means that the current user's authority is effective result[12]：RTC logging, active high means that the chip is in RTC logging process. result[11]：vdet_process_flag, active high means a single temperature measurement process has been interrupted. result[10]：RFU result[9]：light_chk_flag, active high means the light strength is beyond the preset threshold value. result[8]：vbat_pwr_flag, active high means the voltage of battery is higher than 0.9V. result[7:4]: RFU result[3:0]: RFU

2.4.4.4.11 **GetRandom**

Get random command is used to get the 32bits random number from the chip. The password that will be transmitted has to be calculated with the password and the random number (see "Auth").

	Command	cmd Cfg	RN	CRC-16
#of bits	16	8	16	16
description	16'he00d	See detail description	Handle	

Response：

	Header	Result	RN	CRC-16
# of bits	1	32	16	16
description	0	32bits random number, LSByte send first	Handle	

2.4.4.4.12 **LED Ctrl**

Led ctrl command is used to control the off chip LED's lighting or not.

	Command	cmd Cfg	RN	CRC-16
#of bits	16	8	16	16
description	16'he00e	See detail description	Handle	

response：

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	See detail description	Handle	

Detail description：

para	byte	description	
cmd cfg	1	Command configuration	
		[7:2]	RFU
		[1:0]	led_on_en: control the off chip LED's lighting or not by output "0" or "1" signal on the IO_EXT pad. 10: led on 00/01/11: led off
result	2	16'h0000: command is excuted successfully	

2.4.4.4.13 **Initialize Reg**

In the semi-active mode, this command is used to initialize the value of the register under the battery power domain including EPC code of UHF, analog configuration of the UHF interface, etc. It should be executed first when there is UHF interface and after the battery has been fixed. Otherwise, the UHF interface will not work normally.

	Command	cmd Cfg	RN	CRC-16
#of bits	16	8	16	16
description	16'he011	8'h0	Handle	

Response：

	Header	Result	RN	CRC-16
# of bits	1	16	16	16
description	0	16'h0	Handle	

2.5 Power management

DT160 supports several power supply modes:

1. Only supply by VBAT pin
   When the tag is in the HF field, the chip is powered by the rectified energy of the HF field.
   When the tag is only in the UHF field or the chip is in RTC logging process, the chip is powered by the energy supply of VBAT pin.
   In this mode, VCC can be used as VOUT that can output the energy from VBAT pin, HF field or UHF field.

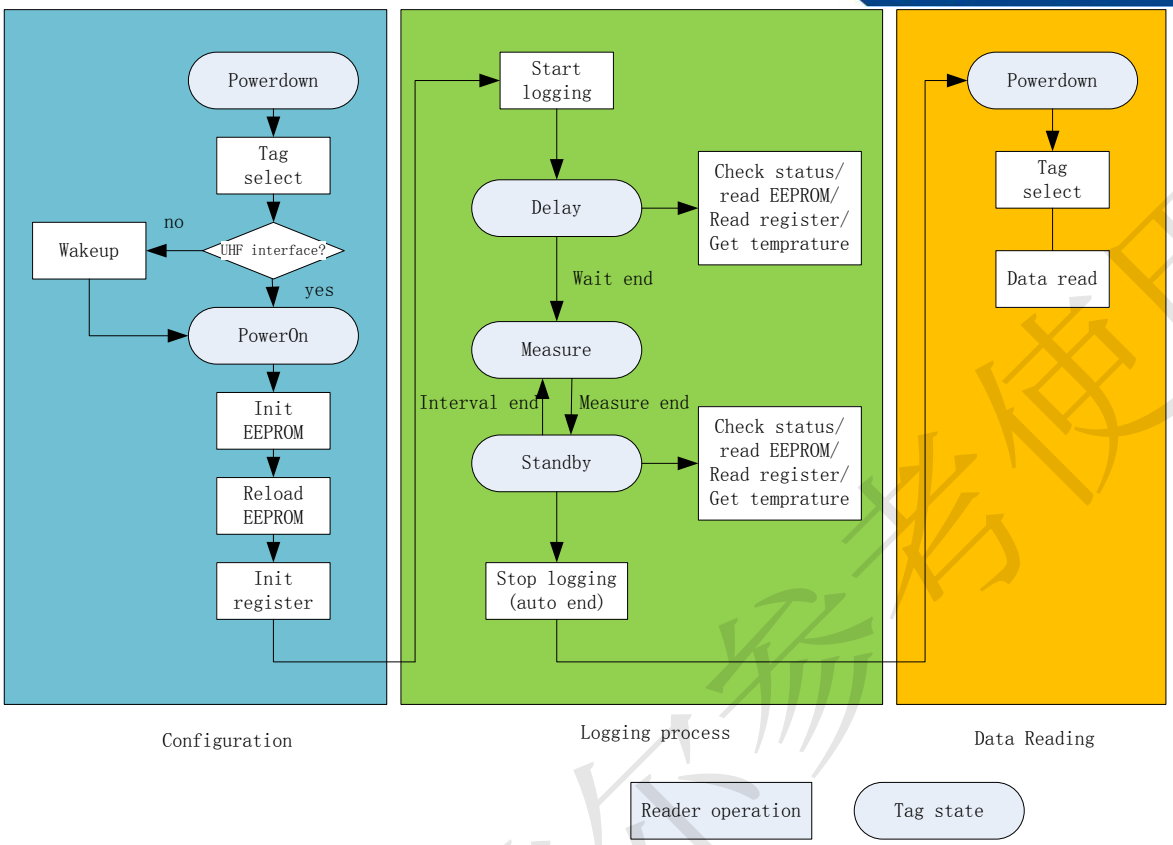
2. Only supply by VCC pin
   In this mode, RTC logging process can not work because VBAT is not supplied.
   When there is a power supply on VCC pin, the chip works with the energy from VCC pin. Otherwise, the chip will work with the rectified energy of HF or UHF field. HF energy will be used when the two fields all exist.

3. VCC and VBAT are all supplied
   In this mode, VCC can not be used as VOUT to output energy.
   When the chip is in RTC logging process, it is powered by VBAT. VCC, HF field and UHF field energy will be shielded.
   When the chip is in standby mode or PD mode, it will powered by VCC or HF field or UHF field.

2.6 Function Description

2.6.1 RTC logging process

DT160's RTC logging process can be divided into three steps：Configuration, Temperature logging and Data reading. The figure below describes the detail:



2.6.1.1 Configuration

The following configuration should be completed before starting a RTC logging process.

There are two types of configuration:
One type is the configuration bits in the EEPROM which is realized by Write Memory command sent from one of the three interfaces. The new configuration will be effective only after the chip powers on again. If it is needed to be effective immediatly, Op_Mode_Chk command can be used to load the info of EEPROM into the register. After the configuration completed, it is suggested to lock the sector to readonly by writting 8'h5A to the last block of the sector. This will avoid the configuration bits to be changed by mistake. If the configuration bits need to be changed after locked, unlock password should be verified by sending AUTH command firstly.

The other type is the register's value under the battery power domain which is realized by Write reg command sent from one of the three interfaces. The configuration will be effective immediatly after the writting operation completed. The register’s value is not able to be changed after the RTC logging process has started.

Note: The register under VBAT power domain is effective only when VBAT powers on. So these register should be re-configurated when VBAT power off and power on again.

1. **The temperature data’s storage position in EEPROM**

   There are 2 data area that can be used to store the temperature data. It needs to be determined in advance that which area will be used in the application, the size of the area and the start position to save the data. You can merge the two data area by set the data area part0’size to 20k byte when the user area is not exist. If the user area’s size is

1k byte, data area part0's size should to be set to 19k byte to fulfill the merge operation.

Configuration	EEPROM address	Function Description
data_area_start_block_pointer	0xb048~0xb049	These 2bytes defines the start point of the address when the temperature data need to be written to the data area in the temperature logging process. For example, if data_area_start_block_pointer=0，the data will be written from the block0 of data area.
user_cfg1.logging_data_area_sel	0xb044~0xb047	Select which data area is used to store the logging data. 0: data area0 is selected 1: data area1 is selected
data_area_part0_size	0xb057	Define the size of the data area part1. The actual size is data_area_part0_size[4:0]*1024bytes.

2. The temperature data's storage format

1) DT160 supports multiple storage formats (see 2.2.5). It needs to be chosen which data format will be used in the logging process. Then detail options will be selected such as the extra information (parity, battery voltage, light strength etc.). The limit value should be selected in the limit mode.
2) Application demands can be defined further according to the different data storage format. For example, in the compress mode and normal mode, the extra space can be used to store the parity check bit, the low-voltage flag of the battery, the over-limit flag of the HF field strength which maybe inflect the temperature measurement accuracy, the over-limit flag of the light strength, etc.

Configuration	EEPROM address	Function Description
user_cfg0.temp_format_cfg[2:0]	0xb040~0xb043	Configuration of the temperature data storage format
user_cfg2.temp_format_bit_info_sel[2:0]	0xb048~0xb04b	In compress mode and normal mode, to select the type of assistant information such as the parity check bit, the low-voltage flag of the battery, the over-limit flag of the HF field strength which maybe inflect the temperature measurement accuracy, the over-limit flag of the light strength, etc.

3) If the selection mode is limit mode, the up and down threshold should be configurated before starting the logging process.

Configuration	EEPROM address	Function Description
max/min_limit0/1/2/	0xb080~0xb08b	The configuration of the up and down temperature threshold.

4) The parameter that will be used to calculate the actual temperature value. These parameters have been initialized on the wafer test stage. If customer needs to re-trim the on chip temperature sensor, these parameters can be updated, otherwise it is not commended to change them.

Configuration	EEPROM address	Function Description
vdet_offset	0xb04a~0xb04b	The parameter "offset" in the temperature computational formula: t=A*cnt/8192+B+offset.
vdet_a	0xb04c~0xb04d	The parameter "A" in the temperature computational formula: t=A*cnt/8192+B+offset.
vdet_b	0xb04e~0xb04f	The parameter "B" in the temperature computational formula: t=A*cnt/8192+B+offset.

5) Selection of the decimal precision of the temperature data. This configuration is used to make compromises between the measurement range and the measurement accuracy. If the configuration is 8bits integer and 2bits decimal, the result will be a large temperature range and relatively poor accuracy. On the other hand, if the configuration is 7bits integer and 3bits decimal, the result will be a small temperature range and better accuracy.

Configuration	EEPROM address	Function Description
user_cfg0.temp_format_high_preci_en	0xb040~0xb043	Configuration of the temperature data storage accuracy: 0：10bit temperature data, 8bits integer and 2bits decimal 1：10bit temperature data, 7bits integer and 3bits decimal

6) Definition of how to deal with the situation that the data area is full of logging temperature data.

Configuration	EEPROM address	Function Description
user_cfg1.data_area_ovflow_mode	0xb044-0xb047	0：new data is discarded, RTC logging process is terminated automaticly 1：RFU

3. The parameter in logging process

1) Configuration of the delay time to start the logging process. For example, if the delay time is 30 minutes, the logging process will start after 30 minutes when Start logging command received.
The configuration is fulfilled by writing the vdet_delay_start_cfg register.

Confiuration	REG Address	Function Description

Confiuration	REG Address	Function Description
vdet_delay_start_cfg	0xc084	The delay time to start the logging process whose minimum uint is 1m.

2) Configuration of the interval time of the logging process. This value of the interval time is defined by vdet_step_cfg0 register which should be configurated before the logging process starting.

Confiuration	REG Address	Function Description
vdet_step_cfg0	0xc085	Configuration of the interval time of the logging process.

DT160 has a special function which is automatic changing the interval time when logging temperature data has been over-limit. Before starting the logging process, the vdet_step_cfg in the EEPROM should be configurated to the same value as that in the register such as 5 minutes. Then vdet_alarm_step_cfg is configurated to a smaller step such as 1 minute. In the logging process, if the temperature data is out of the range which is determined by max_alarm_limit or min_alarm_limit, the interval time will be changed from 5m to 1m automaticly. When the logging temperature data is back into the normal range, the interval time will be changed from 1m to 5m automaticly.

Configuration	EEPROM address	Function Description
user_cfg1.vdet_step_auto_cfg_en	0xb044-0xb047	Automatic interval changed function's enable bit, higactive
vdet_step_cfg/vdet_alarm_step_cfg	0xb0a4-0xb0a7	Configuration of the normal interval time and the abnormal interval time
Max_alarm_limit/min_alarm_limit	0xb08c-0xb08f	The up and down temperature limit's threshold

3) Configuration of the maxium measurement times in the current logging process. The whole RTC logging time can be calculated with the maxium measurement times. The logging process will be terminated automaticly when the maxium time arrived and the chip will enter PD mode.

Configuration	EEPROM address	Function Description
rtc_vdet_auto_finish_en	0xb040~0xb041	The enable bit of the RTC logging process automatic terminated function, high active
rtc_cnt_limit	0xb094-0xb095	Configuration of the maxium measurement times of the logging process

4) Configuration of the lighting an off-chip LED function in logging process. There are 2 purposes to light the LED: one is a flag that indicate the logging process is going normally by lighting the LED at the end of every measurement step. The other is giving a warning message when the logging temperature is over limit by making the LED shinning several times.
Note: Lighting the LED one time is to indicate the logging process working normally.

To differentiat from it, the shinning times should be set to more than 1 time. 5 times is recommended to make this function effective.
This function is fulfilled by writing the configuration bit in the EEPROM. First of all, the io_pad_cfg[1:0] should be written to be "01" in the sector1's user_cfg0. That aims to set the IO/IRQ pad's function to be lighting LED.
For the working state flag function, the led_auto_flash_en should be set to 1 in the sector1's user_cfg1 firstly. Second, the led_mode_cfg1 in sector1 should be written to configurate the pulse width of lighting.
For the warning function, the led_mode_cfg2 should be written to fulfill the configuration of the shinning times and the pulse width. If the led_mode_cfg2[7:4] is set to 000, the warning function will be disabled.

Configuration	EEPROM address	Function Description
user_cfg0.io_pad_cfg[1:0]	0xb040-0xb043	IO pad will be used to light the off-chip LED if this bit is set to 01
user_cfg1.led_auto_flash_en	0xb044-0xb047	The enable bit of lighting the LED in the end of every step of the logging process. High acive.
sector1. led_mode_cfg1	0xb062	Configuration of the pulse width of the single lighting the LED
sector1. led_mode_cfg2	0xb061	Configuration of the shinning times and the pulse width

5) Configuration of the summary information. The initial value of the maximum and the minimum temperature data is defined by writing the summary_max_temp and summary_min_temp register. You can get the max or min value of the logging data by reading the register. And the times of over-limit can also be gotten by reading summary_max_limit_cnt and summary_min_limit_cnt register.

Configuration	REG address	Function Description
summary_max_temp	0xc098	The maximum value of the temperature data in logging process
summary_min_temp	0xc099	The minimum value of the temperature data in logging process
summary_max_limit_cnt	0xc09a	The times of temperature data beyonding the high limit threshold in logging process
summary_min_limit_cnt	0xc09b	The times of temperature data beyonding the low limit threshold in logging process

2.6.1.2 Logging process

1. Stop logging command and Op_mode_chk command whose cmd_cfg is 000001h should be sent to the chip firstly after the configuration to ensure the chip is in the right state. Then Start logging command needs to be sent to start the RTC logging process. These commands can be sent from the 3 interfaces which is UHF, HF or I2C.

2. After the logging process started, DT160 will start the process following the step that be set previously such as 5 minutes. The result of measurement will be written into the EEPROM automaticly. Some other infomation such as if there is a strong HF field in the logging process if there is a strong light, if the battery's voltage is too low, etc. If the field strength is too strong, the chip will be heated to make the result suspectable. When the reader or the smart phone gets this information, it can determine how to deal with the logging data. If the cold chain object is some pharmacautical which is sensitive to the strong light, the light information will be helpful.

3. If there is a LED on the tag whose function is enabled, you can judge if the tag works normally or if the temperature is over limited.

4. In the logging process, if the chip is in the standby state, it can receive the command sent from one of the three interfaces. Get temperature command can be used to acquire the current envionment temperature. Read memory command can be used to read the logging data from the EEPROM. Read reg command can be used to read the infomation stored in the registers such as the maxium temperature value. Op_mode_chk command can be used to inquire if the is in the RTC logging process.

2.6.1.3 Data Reading

1. The logging temperature data can be read from one of the three interfaces. UHF interface is suitable to be used in the long distance, big quantity applications such as collecting the logging temperature data by an UHF reader which is fixed at the gate of the warehouse when the goods entering or getting out of the warehouse. Thanks for the NFC function of the smart phone, the end customer can read the logging data when they receiving the goods. This enables the customer to check the state of the cold chain. I2C interface can be used to communicate with the off-chip MCU to fulfill some special function.

2. NFC smart phone and UHF reader can select the following method to read the logging temperature data out of the chip.

(1) Transparent transfer

Read_memory command can be used to read the data by an android smart phone or an UHF reader with transparent transfer function.

(2) Summary of the data

If the data volume is too large, you can select to read the summary only which includes the maxium value, the minimum value and the times of over temperature limit threshold.

3. Finding the position of the abnormal data tag

If there are some tags with abnormal temperature data such as beyonding the limit range, select command can be used to select the abnormal tag, and then use the Led Ctrl command to light the LED on the tag to find the tag's position.

2.6.1.4 Raw data processing

Through 2.6.1.1 parameter configuration, temperature data and decimal precision selection under Normal mode are defined: Select user_cfg0.temp_format_high_preci_en=0 for large temperature range but low decimal precision (2 decimal, 0.25 degree), Select user_cfg0.temp_format_high_preci_en=1 for higher decimal precision (3 decimal, 0.125 degree).Customers can also customize the processing based on the original temperature data according to their needs.。

vdet_A	2	One of the Three Parameters of the Formula for Calculating

		Temperature, T = vdet_a*cnt/8192+vdet_b+offset. Vdet_a default value is 618.625, EE stored value is 2a0a, high 12 bits for the integer part, low 4 bits for the decimal part. The default value of vdet_b is - 287.125. The value of ee5e is stored in ee. The integer part is 12 bits high and the decimal part is 4 bits low. Offset default value is 0, EE storage value is 0000, 12 bits high is the integer part and 4 bits low is the decimal part. The three parameters are all signed numbers with the highest bit being the symbol bit.
vdet_B	2	One of the three parameters of the formula for calculating temperature is defined as the same Ditto.
vdet_offset	2	One of the three parameters of the formula for calculating temperature is defined as the same Ditto.

2.6.2 Single time measurement

DT160 can measure the temperature by the Get_temperature command single timely. It also can also be used to measure the analog signal on the IO pad which is input from the off chip sensor, or the battery's voltage. The commands in the three interfaces are almost the same whose actual function is selected by the Cmd_cfg configuration. What is needed to noticed is that this command needs to send twice to fulfill one single time measurement. The first command is to start the measurement and the second command is to send back the result. The interval time between the two commands should be greater than 300ms to ensure enough measurement time.

As what mentiond before, the HF field strength can affect the accuracy of the temperature measurement. It is recommanded to send the Field_strength_chk command to measure the HF field strength firstly before sending Get_temperature command. The temperature result can be corrected based on the field strength. Since the actual temperature rising is related to many factors such as the package type of the chip, the size of the tag's antenna etc, user need to do some experiment to creat a table which includes the correlation datas of the field strength and temperature rising.

2.6.3 Light detection

DT160 can detect the strong light which can be used in the light sensitive applications. There are two methods to get fulfill this function:

1. In the RTC logging process, strong light will be detected automaticly and the result will be writen into the data area with the temperature data.

How to use this function is shown below:

(1) The enable bit named light_chk_en in user_cfg2 of sector1 should be set to "1" to enable the light detection function.

(2) Setting the threshold of the strong light detection by the configuration of light_chk_cfg[5:0] in ana_cfg2 of sector3.

(3) In the logging process, the strong light detection will be done automaticly and the result will be writen in the EEPROM.

2. Accquiring the strong light warning signal instantly
(1) Setting light_chk_en in user_cfg2 of sector1 to "1" to enable the light detection function.
(2) Setting the threshold of the strong light detection by the configuration of light_chk_cfg[5:0] in ana_cfg2 of sector3.
(3) Chip power down, then power up again. Sending op_mode_chk command, accquiring the result by the result[9]'s value in light_chk_flag.

2.6.4 Passive mode

1. HF passive mode：
Despite of the VBAT pin connecting battery or not, HF interface (ISO15693 or ISO14443) can all work in passive mode to fulfill functions such as communication, temperature measurement or W/R EEPROM.
When using HF passive mode to measure the temperature, if the tag is near the read's antenna, the field's energy will heat the chip that result in low accuracy temperature measurement. To solve this problem, a special function is designed in DT160. Before the temperature measurement starting, Field_strength command can be sent firstly to acquire the field strength's value. Then send Get_temperature command to get the current temperature. Based on the experience data of the correlation between field strength and the temperature, correction can be done with the data. Then the influence of the field heating will be weakened partly.

2. UHF passive mode：
The UHF interface of DT160 is mainly focused on the application in active mode. So the sensitivity of passive mode is relatively low which makes it not suitable for the purely passive application. If the tag has only the UHF interface, the passive mode can be used to read the temperature data or other infomation when the battery is disconnected with the tag or the battery runs out of the power.

2.6.5 Authority management

2.6.5.1 Password management

There are three types of password in DT160 which are stop logging password, user area password and unlock password. Users can configurate them in the configuration area section 4.

1. User area password

(1) This password is used to control the authority of the read and write of the user area. Read_memory and Write_memory command from all of the three interfaces will not be able to be executed successfully if the password verification not passed.

(2) The write and compatibility write command of the ISO14443 interface will be forbidden if the password verification not passed. Otherwise, read and fast read command can be executed normally.

(3) The read single block, write single block, read multi blocks and write multi blocks command can be executed if the password verification not passed. If you need to control the write authority, you can use the lock command to lock the user area to be readonly.

(4) The write and read command of the ISO18000-6C interface can be executed successfully if the password verification not passed. If you need to control the authority of writting the user area, you can use the lock command to lock the user area to be readonly.

2. Unlock password

This password is used to control the authority of writting the configuration area. There is a sector_lock byte in configuration area1, configuration area2 and configuration area3 individually. The configuration area is read only if this byte is 8'h5A. If you want to change it to writable, you need to do the unlock password verification using the Auth command.

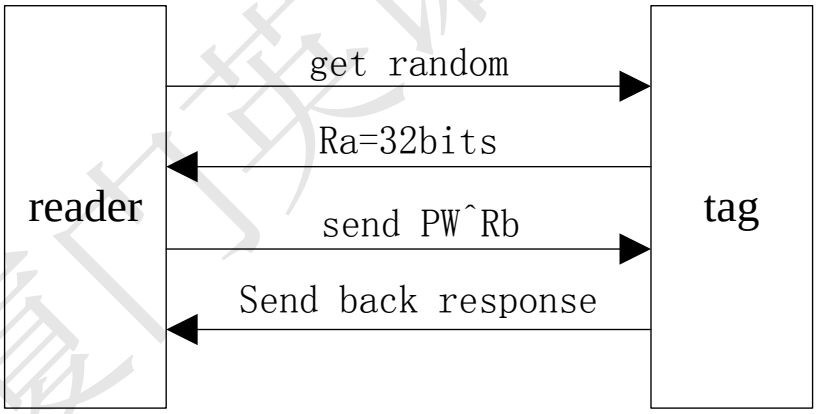
The password is stored in configuration area4. When the password is not 0 and you want to change it, you need to do this password's verification by the Auth command firstly. Then you have the right to change it.

3. Stop logging password

This password is used to control the execution authority of the Stop logging command. If the password is not 0 and you want to execute this command, you need to send the Auth command firstly to verificate the password.

2.6.5.2 Password verification flow

The password verification flow is shown below, Please consult Fudan Micro for detail description document.

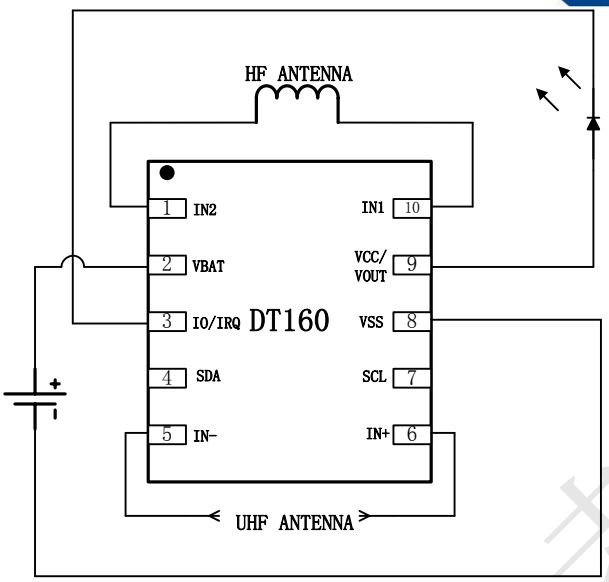


2.6.6 IO/IRQ applications

The IO/IRQ pin of DT160 can be used to lighting off-chip LED, analog signal input and send a wakeup signal to the off-chip MCU.

2.6.6.1 Lighting LED

To use the lighting LED function, a LED should be connected between the VCC pin and the IO_IRQ pin.



There are three types of situation that need to light the LED which are the flag of RTC logging process operatting normally, the warning message of temperature over-limit in logging process, the sight position of the tag with abnormal temperature data.

1. To use the the flag of RTC logging process operatting normally function and the warning message of temperature over-limit in logging process function, the chip need to set io_pad_cfg of user_cfg0 in sector1 to be 2'b01 and set led_auto_flash_en of user_cfg2 in sector1 to 1, then start the RTC logging process.
   After configuration, the chip will light the LED according to the led_mode_cfg (sector1)'s set at the end of every measurement step in RTC logging process.

vdet_flash_len_cfg[3:0] the time length of lighting LED at the end of every measurement step in RTC logging process, t=(cfg+1)*100ms
vdet_limit_times_cfg[7:4] The LED's shinning times when the temperature is out of the limited range. If cfg=0, the LED will not shine.
vdet_limit_len_cfg[3:0] The shinning pulse's width when the temperature is out of the limited range. t=(cfg+1)*100ms

2. To realize the tag's sight position function, read should send the led_ctrl command, lighting or shutting the LED.

2.6.6.2 Analog signal input

DT160 can record the analog signal input from the off-chip sensor or send this signal's value to the reader.
The io_pad_cfg of user_cfg0 should be set to 2'b11. Then power down and power up again. The next step is to set the ana_cfg1(0xc012) to 0x0004 by the write_reg command. In the end, sending get temperature command whose cmd_cfg[6:4] is 010 to fulfill the measurement and recode of the off-chip sensor's signal.

2.6.6.3 Sending the wakeup signal

When the chip is used as the temperature sensor and the contactless interface, the chip can send a wakeup signal as an interrupt of the MCU by the IO/IRQ pin.
User can define the interrupt source and the interrupt signal by writting user_cfg3 in sector1.

cfg	bit	bit name	description
user_cfg3	[2:0]	io_int_en (default 111)	Interrupt source selection on IO/IRQ pad: Bit0=1: sending an interrupt signal when entering or leaving the HF field range. Bit1=1: sending an interrupt signal when received an effective ACK command from the UHF reader. Bit2=1: sending an interrupt signal at the end of the single step measurement in the logging process.
	[3]	io_int_mode (default 0)	Interrupt type selection: 0: level-trigged interrupt, a command sent from MCU or Reader is needed to clear this interrupt. 1: edge-trigged interrupt, the pulse width is about 2us.

2.6.7 I2C interface

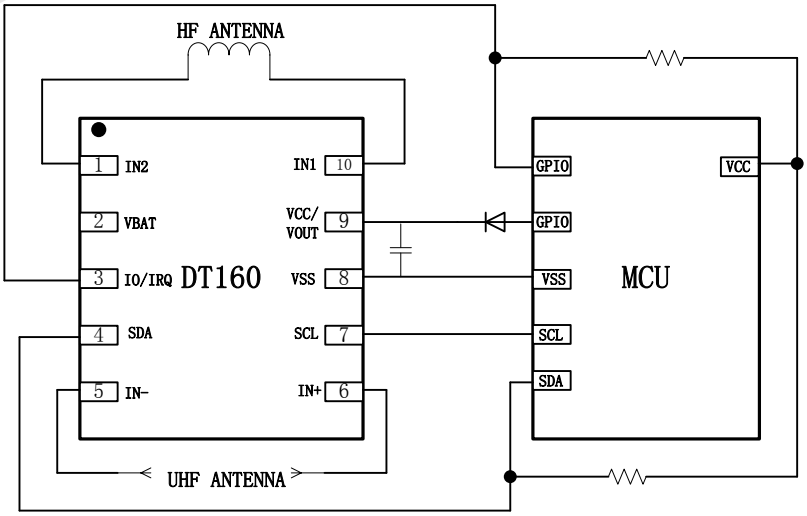
2.6.7.1 I2C applications

DT160 communicates with an off chip MCU by the I2C interface. MCU can send Get temperature command to DT160 to fulfill a single time temperature measurement or the measurement of the analog signal on the IO pad. MCU can read or write DT160's EEPROM without authority check.

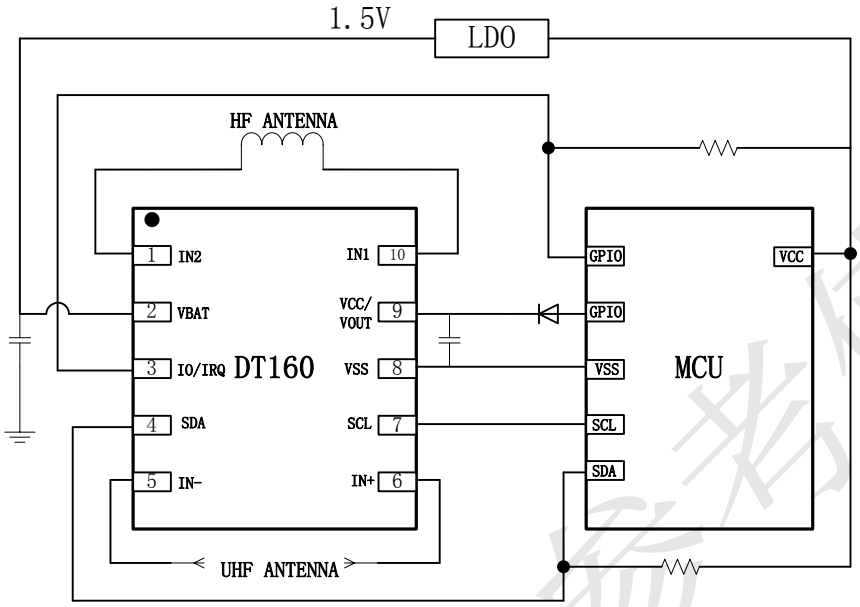
Features：

- I2C slave mode
- 7 bits slave address
- Standard-mode(100Kbps)
- Follows UM10204 Rev. 03

DT160 has two types of connection with the off-chip MCU:
1. The off-chip MCU has a RTC itself and do not need to use DT160's on-chip RTC：



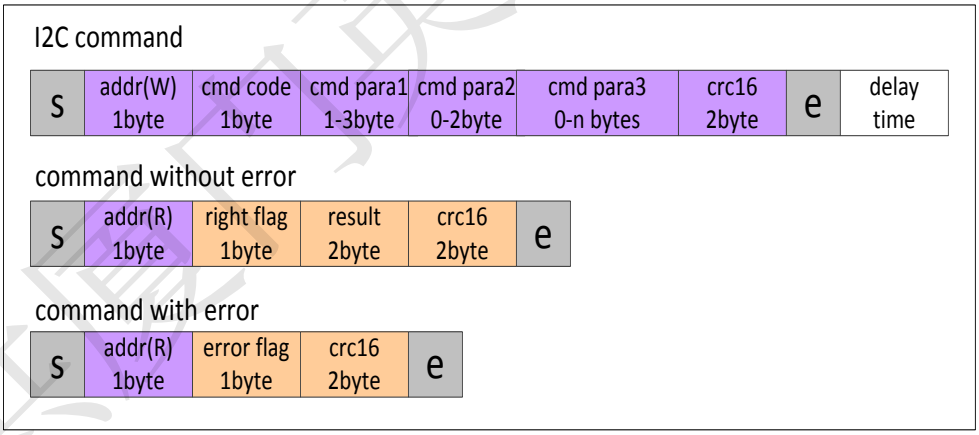
2. The off-chip MCU has not RTC and it need to use DT160's on-chip RTC：



In this kind of application, an interrupt signal such as entering UHF field or HF field can be sent to wake up the MCU. Then one operation such as single time temperature measurement can be fulfilled by the command from I2C interface. After that operation completed, Deep sleep command is suggested to be sent to let DT160 get into PD mode that can ensure low power consumption. Then off-chip MCU can also enter sleep mode and wait for the next wakeup signal coming.

2.6.7.2 I2C command format

One standard I2C command is shown in below figure. Master such as MCU need to send the command firstly, then send the second command to get the result after a period of delay time.



I2C's slave address can be configured in sector1 of the configuration area. If the bit9 is low, the slave address will be invalid and I2C can be addressed with arbitrary address.
CRC check can be disabled by set the bit8 to low.

i2c_cfg	2	[15:10]	RFU
		[9]	The enable bit of I2C's multi slave mode, active high.
		[8]	The enable bit of I2C's CRC check, active high.
		[7]	RFU
		[6:0]	I2C slave interface address

The initial value of CRC check is fixed to 0x6363 and the polynomial is also fixed to 0x8408.
An example is shown below:

MCU	DT160
If the data to be sent is “00 cf 00” whose CRC is f374. The CRC’s reverse order is 2ecf. Then the data with CRC to be sent is “00 cf 00 2e cf”.	-
-	The slave receives the data and calculates the CRC. If the result is 0000, CRC check is passed.
-	The slave will send the data “01 00 78 56” whose CRC is b8b3. Its reverse order is cd1d. Then the data with CRC to be sent is 01 00 78 56 cd 1d.
Master receives the data and calculates the CRC. If the result is 0000, CRC check is passed.	-

2.6.7.3 I2C command

Command code	Command name	Function description
0xb1	Read memory	Custom read command of EEPROM. Enormous data can be read in one time by this command whose unit is byte.
0xb3	Write memory	Custom write command which can access all of the space of EEPROM. The specified authority is up to the operating mode, chip configuration etc. The minimum unit is byte.
0xc0	Get temperature	Single measurement command. Two commands need to execute to fulfill one single measurement process. The first command is used to start the single measurement. The second command is used to sendback the measurement result. The interval time between two commands should be longer than 300ms.
0xc2	Start logging	Start the logging process
0xc3	Deep sleep	enter PD mode
0xc4	Wake up	Wake up the chip from the PD mode
0xc5	Write reg	Write the register
0xc6	Read reg	Read the register
0xd0	Op mode chk	Check the current mode or refresh the register

See 2.4.3 for detail description.

3 Characteristics

3.1 Limiting values

Symbol	Parameter	Conditions	Min	Max	Unit
T_{stg}	storage temperature		-55	+125	°C
I_{I_HF}	HF interface input current （IN1 to IN2）	IN1 to IN2; RMS	-	30	mA
P_{i_UHF}	Maximum input power	IN+ to IN-		20	dBm
V_{ESD}	ESD （HBM）		【2】	±2	KV

Table 3-1 DT160 Limiting values【1】

【1】: Stresses above one or more of the limiting values may cause permanent damage to the device.

【2】: Human body model: C = 100 pF, R = 1.5kΩ

3.2 Electrical characteristics

3.2.1 Pin characteristics

Symbol	Parameter	Conditions	Min	TYP	Max	Unit
f_{i_HF}	HF input frequency	【1】	13.553	13.56	13.567	MHz
C_{i_HF}	HF input capacitance	Between IN1 and IN2 【2】	22.3	23.5	24.7	pF
f_{i_UHF}	input frequency of UHF		840		960	MHz
Ri	UHF input resistance	Between IN+ and IN-, 25℃，920MHz		10-j*72		Ω
Digital Input						
V_{IL}	Input low voltage		0		0.3Vcc	V
V_{IH}	Input high voltage		0.7Vcc		Vcc	V
I_{leak}	Input leakage current				1	uA
Open Drain output						
VOL	OD output	Vcc=3.3V，I_O=4mA	0		0.3 Vcc	

Table 3-2 Pin characteristics

【1】 Bandwidth limitation (±7 kHz) according to ISM band regulations

【2】 Measured with Agilent E5061B at 13.56 MHz and 0.707V RMS.

3.2.2 Electrical characteristics

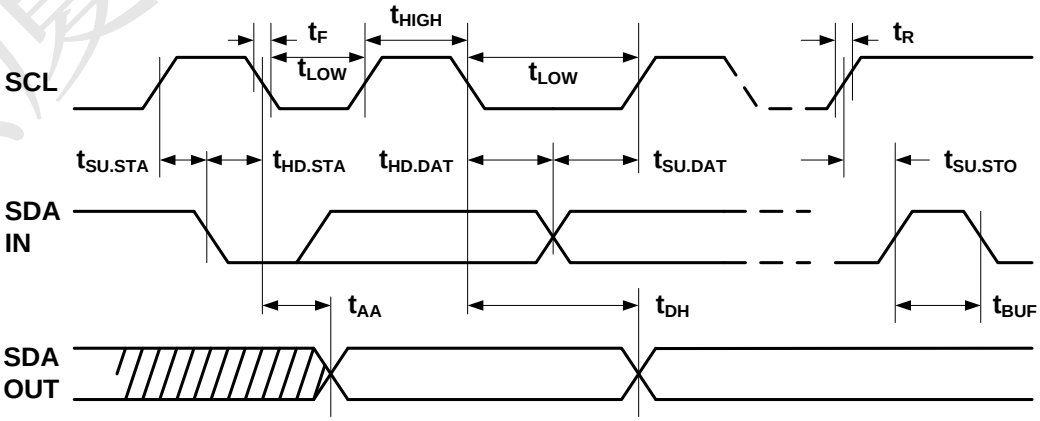
Symbol	Parameter	Conditions	Min	TYP	Max	Unit
V_{cc}[1]	Contact interface supply voltage		2.7	3.3	3.6	V

Symbol	Parameter	Conditions	Min	TYP	Max	Unit
Vbat	Battery supply voltage		1.1	1.5	1.65	V
I_{BAT-PD}	Supply current in power down mode	25℃, Vbat=1.5v		0.06	0.1	uA
$I_{BAT-STD}$	Supply current in standby mode, RTC is working	25℃, VBat=1.5v		0.6	1	uA
I_{BAT-OP}	Battery current in measurement	25℃，VBat=1.5v		440		uA
I_{EE_WR}	Battery current when Writing EEPROM	25℃，VBat=1.5v		640		uA
V_{out}	Harvesting energy output voltage	25℃			5	V
T_{SR}	Temperature measurement range		-35		50	℃
T_{ACC}	Temperature accuracy	-35~50℃	-0.5		0.5	℃
T_{RTC-I}	Logging step	configurable	1		65535	s
T_{delay}	Logging start delay time	configurable	0		65535	m
T_{meas}	The duration time of single step temperature measurement in the logging process			180		ms
t_{RTC}	RTC accuracy	-20~50℃, Vbat=1.1~1.65	-2%		2%	

Table 3-3 Electrical characteristics

【1】 When testing the input voltage Vcc of the chip, it is necessary to add a reverse diode from Vcc to the external power supply to prevent the internal voltage output from affecting the test results of the external input voltage.

3.2.3 I2C interface characteristics



Operating condition：T_{BAB}= -40°C ~ +85°C，V_{BCCB} = +3.0V ~ +3.6V，CL = 100 pF

Symbol	Parameter	Standard(100kHz)	Unit

		Min	Typ	Max	
f_{SCL}	Clock Frequency, SCL			400	kHz
t_{LOW}	Clock Pulse Width Low	1.3			μs
t_{HIGH}	Clock Pulse Width High	0.6			μs
t_{AA}	Clock Low to Data Out Valid	0.1		0.9	μs
t_{BUF}[1]	Time the bus must be free before a new transmission can Start	1.3			μs
$t_{HD.STA}$	Start Hold Time	0.6			μs
$t_{SU.STA}$	Start Setup Time	0.6			μs
$t_{HD.DAT}$	Data In Hold Time	0			μs
$t_{SU.DAT}$	Data In Setup Time	100			ns
t_R	Inputs Rise Time			0.3	μs
t_F	Inputs Fall Time			0.3	μs
$t_{SU.STO}$	Stop Setup Time	0.6			μs
t_{DH}	Data Out Hold Time	100			ns
t_{WR}	Write Cycle Time			8	ms

Table 3-4 I2C interface characteristics

【1】 These parameters are up to the sample test and not tested 100% on wafer.

【2】 Test condition：

RL (connect to VCC)：1.3 kΩ

Input pulse voltage：0.3 VCC ~ 0.7 VCC

Input rising time/falling time：≤ 50 ns

Reference voltage of input /output：0.5 VCC

3.3 EEPROM characteristics

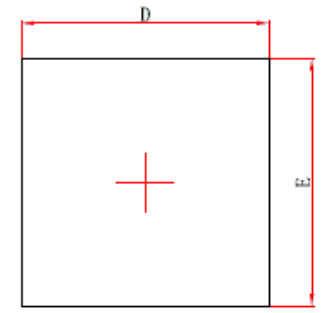
Symbol	Parameter	Conditions	Min	Typ	Max	Unit
t_{ret}	retention time	T_{amb} = 55℃	10			year
$N_{endu(W)}$	write endurance	T_{amb} = 25℃	200000			cycle

Table 3-5 EEPROM characteristics

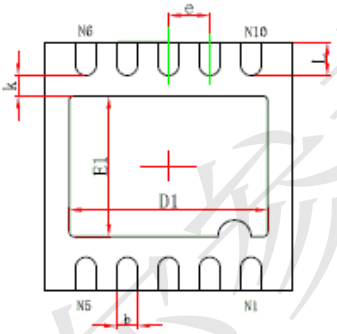
4 Package information

4.1 TDFN10

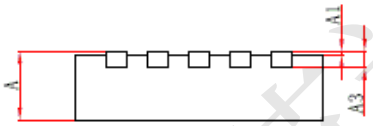
DFNWB3×3-10L(P0.50T0.75) PACKAGE OUTLINE DIMENSIONS



Top View



Bottom View



Side View

Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min.	Max.	Min.	Max.
A	0.700	0.800	0.028	0.031
A1	0.000	0.050	0.000	0.002
A3	0.203REF.		0.008REF.	
D	2.924	3.076	0.115	0.121
E	2.924	3.076	0.115	0.121
D1	2.300	2.500	0.091	0.098
E1	1.600	1.800	0.063	0.071
k	0.200MIN.		0.008MIN.	
b	0.200	0.300	0.008	0.012
e	0.500TYP.		0.020TYP.	
L	0.324	0.476	0.013	0.019

5 Ordering information

5.1 TDFN10

Type name	Package	Pack
FM13DT160-T2T-DNC-T-G	TDFN10	Reel
FM13DT160-T5T-DNC-T-G	TDFN10	Reel

FM 13DT 160 -X -X -X -X

Company Name

FM=Shanghai Fudan Microelectronics Group Company Limited

Product Family Name

13DT=Dual frequency temperature sensor and logger IC

Memory

160=160k bits EEPROM

HF type

T2T= ISO14443 communication protocol

T5T= ISO15693 communication protocol

Package type

DNC=TDFN10

Carrier

T= Reel

U= Tube

HSF Code

G=ROHS Compliant, Halogen-free, Antimony-free

5.2 Bare die with Gold bump

Type name	Wafer type	Description
FM13DT160-T2T-WIB5-DP	Bump wafer	8inch bump wafer(sawn,150um thickness)
FM13DT160-T2T-WIS5	Sawn wafer	8inch wafer(sawn,150um thickness)
FM13DT160-T5T-WIB5-DP	Bump wafer	8inch bump wafer(sawn,150um thickness)
FM13DT160-T5T-WIS5	Sawn wafer	8inch wafer(sawn,150um thickness)

FM 13DT 160 -X -X -X

Company Name

FM=Shanghai Fudan Microelectronics

Product Family Name

13DT=Dual frequency temperature sensor and logger IC

Memory

160=160k bits EEPROM

HF type

T2T= ISO14443 communication protocol

T5T= ISO15693 communication protocol

Wafer type

WIS2= sawn wafer

WIB2= bump wafer

Bump type

DP= Dual frequency with power

Revision history

Rev	Release date	Pages	Modifications
1.0	May. 2019	114	original version
1.1	Oct. 2019	117	Add content
1.2	Aug. 2020	116	1. Figure 2-3 has been updated. 2. Update the description of Wake up command. It can only be used before Start Logging.
1.3	Sep.2020	116	1. In chapter 2.4.4.4.1 and 2.4.4.4.2, correct the header of the response without authority from 0 to 1. 2. in chapter 2.2.1, add description: the minimum step for user area adjustion is 1kbits. 3. update the sales and service infor in the last page.

Sales and Service

Shanghai Fudan Microelectronics Group Co., Ltd.
Address: Bldg No. 4, 127 Guotai Rd,
Shanghai City China.
Postcode: 200433
Tel: (86-021) 6565 5050
Fax: (86-021) 6565 9115

Shanghai Fudan Microelectronics (HK) Co., Ltd.
Address: Unit 506, 5/F., East Ocean Centre, 98 Granville Road, Tsimshatsui East, Kowloon, Hong Kong
Tel: (852) 2116 3288 2116 3338
Fax: (852) 2116 0882

Beijing Office
Address: Room 423, Bldg B, Gehua Building,
1 QingLong Hutong, Dongzhimen Alley north Street,
Dongcheng District, Beijing City, China.
Postcode: 100007
Tel: (86-010) 8418 6608
Fax: (86-010) 8418 6211

Shenzhen Office
Address: Room.2306-08, A7 Bldg, Chuangzhiyuncheng, LiuXian Rd, Xili street, Nanshan District,
Shenzhen City, China.
Postcode: 518000
Tel: (86-0755) 8335 0911 8335 1011 8335 2011 8335 0611
Fax: (86-0755) 8335 9011

Shanghai Fudan Microelectronics (HK) Ltd Taiwan Representative Office
Address: Unit 1225, 12F., No 252, Sec.1 Neihu Rd., Neihu Dist., Taipei City 114, Taiwan
Tel : (886-2) 7721 1889
Fax: (886-2) 7722 3888

Shanghai Fudan Microelectronics (HK) Ltd Singapore Representative Office
Address : 47 Kallang Pudding Road, #08-06,The Crescent @ Kallang ,Singapore 349318
Tel : (65) 6443 0860
Fax: (65) 6443 1215

Shanghai Fudan Microelectronics Group Co., Ltd NA Office
Address : 97 E Brokaw Road, Suite 320,San Jose,CA 95112
Tel : (+1)408-335-6936

Web Site: http://www.fmsh.com/