|  |  |
| --- | --- |
| 产品名称Product name | 密级Confidentiality level |
| 集中器改进 |  |
| 产品版本Product version | Total 22 pages 共 22 页 |
| V1.00 |

集中器改进详细设计方案

嵌入式软件方面

**产品名称：集中器改进**

**编 写：靳占军**

**日 期：2015.04.10**

**烟台航天德鲁节能科技有限公司**

Revision Record 修订记录

| Date  日期 | Revision Version  修订 版本 | Sec No.  修改 章节 | Change Description  修改描述 | Author  作者 |
| --- | --- | --- | --- | --- |
| 2015.04.08 | 初始版本 |  |  | 靳占军 |
| 2015.04.10 | 按照评审意见修改完善 | MBUS短路保护 | 1、抄表过程补抄次数可设定；  2、补抄时如果BUS短路，则不进行补抄。 | 靳占军 |
| RTC用纽扣电池异常检测 | 新增纽扣电池异常检查。 | 靳占军 |
| 完善表参数属性 | 在“抄表数据格式”中加入“仪表类型”，方便以后兼容多种类仪表。 | 靳占军 |
| 2015.05.13 | 抄表数据中加入抄热表时间和抄阀控时间 |  | 抄表数据中加入抄热表时间和抄阀控时间 | 靳占军 |
| 2015.07.10 | 修改MBUS短路保护方式 |  | 根据日照工程测试情况，另加任务判断通道短路。 | 靳占军 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 1 引言

## 1.1 背景

本公司已开发的集中器抄表系统，在工程项目中已安装使用了千余套，在项目使用过程中发现了如下问题：MBUS短路处理不严谨、远程通信不稳定、多厂家终端设备兼容不便、现场调试不便等问题。针对这些问题，决定对集中器进行改进。

## 1.2编写目的

本文档对集中器改进中嵌入式软件方面工作逐一提出具体解决方法，备项目相关人员共同评审，以求找到最合理的解决方式，为具体程序的编写、调整提供依据。

本文档分以下9方面分别给出详细解决方法。

* + - GPRS信号与登陆指示
    - 完善表参数属性
    - 抄表轮显与查询
    - 远程升级断点续传
    - MBUS短路保护
    - 定时抄表、立即抄表冲突
    - SD卡检测及指示
    - 蜂鸣器
    - 纽扣电池异常检测

## 1.3 参考资料

《集中器改进立项报告》

# 2 GPRS信号与登陆指示

## 2.1 GPRS信号强度获取

集中器使用西门子公司的MC52I无线通讯模块，根据官方AT指令集《MC52i AT Command Set》，使用“AT+CSQ”指令可获取信号强度。

向模块发送指令：AT+CSQ

模块返回：+CSQ: <rssi>,<ber>，其中rssi为信号强度，ber为错误率。rssi值与信号强度对照表如表2.1所示。

表2.1 rssi值与信号强度对照表

|  |  |
| --- | --- |
| **rssi值** | **信号强度** |
| 0 | -113 dBm or less |
| 1 | -111 dBm |
| 2..30 | -109... -53 dBm |
| 31 | -51 dBm or greater |
| 99 | not known or not detectable |

手机信号强度正常范围为 -40dBm至-85dBm，小于-85dBm则认为信号弱覆盖。使用MC52I模块测得办公室信号为 -75dBm左右。

## 2.2 服务器登陆判断

当前集中器中，GPRS模块初始化完成后，会主动向服务器发送“登陆帧”，如果服务器收到该登陆帧，则立即向集中器发送“登陆应答帧”。如果收不到登陆应答，则说明登陆不成功。

服务器登陆成功后，会每分钟主动向服务器发送一帧“心跳帧”，服务器收到心跳帧后立即向该集中器发送心跳应答，或其他按照协议约定好的数据帧，以此维持与服务器的连接。

增加集中器登陆服务器的状态显示，可以利用此“登陆帧”和“心跳帧”与服务器的交互是否成功进行判断。

## 2.3 信号强度及登陆状态指示处理流程

GPRS信号强度和服务器登陆状态指示处理流程如图2.1所示。在“GPRS模块管理任务”中获取信号强度，并进行服务器登陆的相关处理。在“LCD TOP任务”中显示信号强度与登陆状态。

另外，将当前信号强度加入“心跳帧”，随心跳帧发送到服务器，便于远程了解信号情况。

GPRS信号强度及登陆状态指示效果，如图2.2所示。



图2.1 GPRS信号强度与登陆状态指示流程图

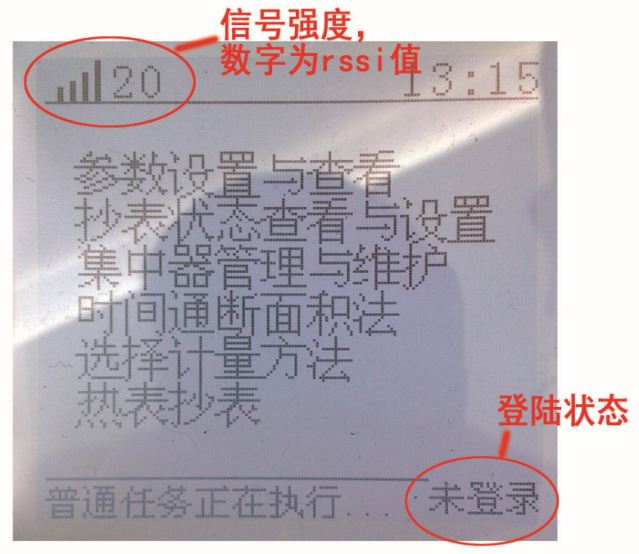


图2.2 GPRS信号强度与登陆状态图例

# 3 完善表参数属性

## 3.1 完善表参数

当前集中器中，热表参数有热表地址、厂商代码、协议版本、设备类型、通道号、阀门地址、面板地址 共7项，如结构体MeterFileType中所示。

typedef struct{

uint8 MeterAddr[7]; //热计量表地址

uint8 Manufacturer; //厂商代码

uint8 ProtocolVer; //协议版本

uint8 EquipmentType; //设备类型

uint8 ChannelIndex; //通道号

uint8 ValveAddr[7]; //阀门地址

uint8 ControlPanelAddr[7]; //控制面板地址

}MeterFileType;

为方便对单个热表进行操作，增加“热表编号”属性。因为单台集中器4个通道挂载的热表数量可能超过256只，而单字节计数范围为0-255，所以“热表编号”属性需要分配2个字节。添加“热表编号”属性后，上位机下发全部表信息时，按照此编号递增的顺序下发。

为实现集中器同时兼容多厂家不同协议的阀控器，增加“阀控协议版本”属性，占用1个字节。

为便于工程调试中确定热表详细位置，也为了在新增的“抄表轮显”中显示对应的用户信息，增加“楼号、单元号、房间号”3个属性，其中“楼号”、“单元号”各占用1个字节，“房间号”占用2字节。

为便于以后扩充其他参数属性，预留8个字节。修改后的热表参数结构体如下，其中浅蓝色表示新增项。

typedef struct{

uint16 MeterID; //热表编号

uint8 MeterAddr[7]; //热计量表地址

uint8 Manufacturer; //厂商代码

uint8 ProtocolVer; //协议版本

uint8 EquipmentType; //设备类型

uint8 ChannelIndex; //通道号

uint8 ValveProtocol; //阀控协议版本

uint8 ValveAddr[7]; //阀门地址

uint8 ControlPanelAddr[7]; //控制面板地址

uint8 BuildID; //楼号

uint8 UnitID; //单元号

uint16 RoomID; //房间号

uint16 Reserved1;

uint16 Reserved2;

uint16 Reserved3;

uint16 Reserved4;

}MeterFileType;

## 3.2 抄表数据格式更改

当前集中器抄表数据存储格式如表3.1所示，热量数据长度N现固定为43，共占用58字节。但在SD卡中每个表分配128字节的存储空间，以备扩充。

表3.1 现SD卡中存储格式

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **数据项** | 数据长度 | 热表地址 | 热量数据长度 | 热量数据 | 温度数据 | 阀状态 | 预留 | CS校验 |
| **字节数** | 1 | 7 | 1 | N | 3 | 1 | 1 | 1 |

完善表参数后，为达到参数完善目的，抄表数据存储格式也需要相应调整。抄表存储格式中也加入“热表编号、楼号、单元号、房间号” ，同时完善阀控器相关字段。

为便于以后多种类表（热表、水表、电表等）同时抄读上传，格式中加入仪表类型1字节，通过判断该字段仪表类型，按照对应约定解析即可。当前热表的仪表类型固定为20，表3.2中格式仅为热表的数据格式，其他类型表带具体用时定义。

如表3.2中，浅蓝色为新增字段，调整后最多72字节。其中热量数据N，当抄表成功时固定为43字节，当抄表失败时为0字节。有时需要指示一些信息（如MBUS短路等），此时热量数据N为1字节，具体详见“6 MBUS短路保护”一节。

表3.2 更改后SD卡中存储的抄表数据格式。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **数据**  **项** | 数据长度 | Meter  ID | 仪表类型 | 热表地址 | 楼号 | 单元号 | 房间号 | 抄热表  时间 | 热量数据长度 | 热量数据 | 抄阀  时间 | 户内温度 | 阀开度 | 阀控状态 | 预留 | CS校验 |
| **字节**  **数** | 1 | 2 | 1 | 7 | 1 | 1 | 2 | 3 | 1 | N | 3 | 3 | 1 | 1 | 1 | 1 |

注：1、上传到上位机数据不包括表3.2中的第1字节“数据长度”和最后一字节“CS校验”。

2、抄热表时间、抄阀时间都是3字节，依次是“秒、分、时”，BCD码格式。

其中，阀控器相关字段参考多个厂家阀控协议（亿林、京源、琅卡博、百暖汇、德鲁）制定。户内温度占用3字节，与之前保持一致。阀控状态字节具体表示信息，如表3.3所示。

表3.3 阀控状态各比特位意义对应表

|  |  |
| --- | --- |
| **阀控状态位** | **意义** |
| bit0 | 阀控/面板无线故障，1-故障，0-正常。 |
| bit1 | 面板开关状态，1-关，0-开。 |
| bit2 | 面板锁定状态，1-锁定，0-未锁定 |
| bit3 | 负载锁定，1-阀门锁定开或关，0-阀门开关由面板控制。 |
| bit4 | 面板是否显示热量，1-显示，0-不显示。 |
| bit5 | 面板温度异常，1-异常，0-正常。 |
| bit6 | 阀门是否堵转，1-堵转，0-正常。 |
| bit7 | Reserved |

更改后的集中器中，上传到服务器的抄表信息也按照表3.2的格式上传（上传到上位机的数据不包括表3.2中的第1字节“数据长度”和最后一字节“CS校验”）。

## 3.3 兼容方法

经过3.1、3.2节中调整，新、老版集中器，在下发表地址、抄表上传数据格式 两方面存在差异，不能兼容。为解决这个问题，更改后集中器数据帧协议中“VER版本”由0x03改为0x04，如图3.1所示，当上位机接收到的数据帧“VER版本”为0x04时，则按照更改后的格式处理。



图3.1 集中器数据帧传输协议

# 4 抄表轮显与查询

为改变集中器抄表时只显示“抄表总数”和“抄表成功数”，不能看到详细抄表情况（每户抄表数据、哪些抄表失败等）的弊端，新增“抄表轮显与查询”功能。菜单设置如图4.1所示。图中显示内容只是示意性描述，可根据需要调整。

集中器默认不显示抄表轮显菜单。使用按键切换到此菜单后，如果正在抄表（全抄），则轮显正在执行的抄表信息；如果没在抄表，则显示最近一次的抄表信息，通过按键可按照编号切换显示各户的抄表情况。

图4.1 抄表轮显与查询 菜单设置

如果在现场调试时，按键选择“立即抄表”，则自动切换到抄表信息轮显页面。如果想查询最近一次抄表的失败信息，则可选择“失败信息查询”菜单，通过按键可逐个查询，如图4.2所示。



图4.2 抄表失败信息查询菜单

# 5远程升级断点续传

当前集中器程序远程升级时，有200多帧数据需要发送，常有GPRS信号不稳导致升级失败的情况。如果失败后再次升级则需从头重新开始，不仅效率较低，还浪费GPRS流量。

与上位机一起商定协议与实现过程，添加“断点续传”功能，当升级异常中断需要继续升级时，可以从断点处继续。

为实现程序远程升级及断点续传，定义表5.1中4种指令。

表5.1 远程升级相关指令列表

|  |  |  |  |
| --- | --- | --- | --- |
| **指令字** | **是否有应答** | **响应字** | **说明** |
| 0x82 | 有 | 0x82 | 主站发送升级开始标志 |
| 0x85 | 有 | 0x85 | 主站询问文件缺包情况 |
| 0x83 | 无 | 无 | 主站以一定时间间隔逐包发送升级程序，集中器不返回应答帧 |
| 0x84 | 有 | 0x84 | 主站发送升级结束标志 |

* 升级开始数据帧（0x82）中包括 升级版本号、文件总大小、最大数据包长度、总包数4种信息。集中器收到升级开始数据帧后，按照数据帧内容初始化相关参数，在后续的升级过程中按照这些参数进行处理。另外，集中器中定义有“标记各升级包是否收到(uint8 PacketReceiveFlag[1000])”的数组，接收到升级开始标志后对此数组进行初始化。
* 为实现“断点续传”功能，设置“询问缺包”指令（0x85）。集中器收到此指令后，查询当前的缺包情况，并返回到主站，如存在缺包，主站依次补发相应包即可。
* 为提高远程升级速度，主站发送升级包（0x83）时集中器没有应答指令帧，主站以一定的时间间隔发送升级包即可，时间间隔需要满足集中器的处理速度，经测试10ms即满足。升级包数据经验证正确后，存储到SD卡相应位置。
* 文件传输全部完成并没有缺包时，主站发送升级结束数据帧（0x84）。升级结束数据帧中包括 文件总大小、总包数、版本号、整个文件的CRC校验码，集中器收到后，首先对比文件版本号、总大小、总包数与“升级开始标志”中是否一致，如一致则从SD卡中读出之前存入的升级数据，进行全CRC校验，并与收到的“整个文件CRC校验码”比较，校验正确才会标记升级成功，并重启集中器。重启集中器时bootloader自动将SD卡中升级文件复制到单片机中存储并运行。

远程升级及断点续传流程，如图5.1所示。



图5.1 远程升级及断点续传流程图

# 6 MBUS短路保护

集中器MBUS通道短路时，如果不能及时切断MBUS电压，则会损伤甚至烧坏MBUS模块。当前硬件中已经有MBUS短路保护功能，当有短路发生时，硬件上可以立即断开MBUS的正电压。

为了使MBUS的短路保护更可靠，防止因MBUS虚短路造成MBUS硬件保护频繁通断损坏MBUS模块，在软件方面也给予辅助处理。

MBUS模块接口及说明，如图6.1所示。



图6.1 MBUS模块接口及管脚说明

如图，当MBUS短路发生时，SMBUSIN管脚的电平由低变高，产生一个上升沿。判断短路是否发生可使用查询、中断2种方式。

为了在短路发生时尽快切断所有通道，我们使用中断的方式检测SMBUSIN管脚的电平变化，然后在中断外判断是否真的发生短路（因为通道闭合时震荡也可能触发中断）。处理流程如图6.2所示，除了断开所有通道外，在中断函数中还设置一些变量，以便在抄表任务中进行相关处理。



图6.2 MBUS短路保护相关处理流程

MBUS短路发生后需要在抄表时对抄表数据进行相应处理，抄表部分流程如图6.3所示。MBUS短路保护软件辅助处理的目的是为了更好的保护MBUS模块，如果检测到MBUS短路，则断开本通道，并对本通道中的所有热表进行数据处理并保存到SD卡中；如果需要补抄没有抄成功的表，补抄时也不再抄短路的通道。

本次抄表过程结束后，流程中不设置 返回重抄短路通道热表的过程，当下一个定时抄表时间点到后，再次开始全部抄表即可。

另外，为了适应不同厂家的热表及阀控，有的单次抄表成功率较低，需要补抄，但补抄必然延长抄表时间，为此我们将“补抄次数”参数改成可手动更改的参数，根据具体抄表情况设置补抄次数。按键和上位机指令都可以设置“补抄次数”参数。

需要注意的是，如MBUS通道短路，不能直接跳出本通道，依然需要对本通道的热表数据进行处理，并保存到SD卡中。如果没有本通道的抄表数据，当上传历史数据时可能会发生异常。SD卡中抄表数据存储格式如表3.2所示。历史数据上传流程如图6.4。

如MBUS短路，可将短路信息加在表数据中存入SD卡，随上位机抄历史数据信息一起上传到服务器 并解析，上位机即可获知此MBUS短路信息。

当MBUS短路时，存储到SD卡中的热表数据格式如表6.1所示，热量数据长度固定为1字节，热量数据固定为FF,如示例6.1。

表6.1 MBUS短路时存储到SD卡中热表数据格式

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **数据**  **项** | 数据长度 | Meter  ID | 仪表类型 | 热表地址 | 楼号 | 单元号 | 房间号 | 抄热表  时间 | 热量数据长度 | 热量数据 | 抄阀  时间 | 户内温度 | 阀开度 | 阀控状态 | 预留 | CS校验 |
| **字节**  **数** | 1 | 2 | 1 | 7 | 1 | 1 | 2 | 3 | 1 | 1 | 3 | 3 | 1 | 1 | 1 | 1 |

注：上传到上位机数据不包括表6.1中的第1字节“数据长度”和最后一字节“CS校验”。

示例6.1：MBUS短路，热表地址为11110012345678.





图6.3 MBUS短路保护中断函数外处理流程图



图6.4 抄表历史数据上传处理流程简图

如果有MBUS短路发生，则在LCD显示屏上予以指示，如图6.5。如果使能了蜂鸣器报警功能，则同时发出蜂鸣声提示。



图6.5 MBUS短路屏显方式示意图

# 7 定时抄表、立即抄表冲突

立即抄表是为方便现场调试而添加的功能，正常使用时是不用的。但是，在“立即抄表”过程中同时开始“定时抄表”，会导致LCD屏显混乱。另外也要避免在“立即抄表”或“定时抄表”的同时进行实时抄单表的发生。

解决以上冲突只需要增加一个信号量即可，相关流程如图7.1和图7.2所示。

定时抄表时间到后，先请求该信号量，如请求到则开始定时抄表，如果没请求到，则判断是否在“立即抄表”，如果正在立即抄表，则不启动当前时间节点的定时抄（立即抄表结束后也不开始定时抄），以免影响现场调试。

开始立即抄表时，先请求该信号量，如请求到则开始“立即抄表”，如果没请求到，则一直等待该信号量。如果现场调试必须立即重新抄表，重启集中器后开始“立即抄表”即可。



图7.1 立即抄表、定时抄表流程图



图7.2 实时抄表流程图

# 8 SD卡检测及指示

## 8.1、SD卡初始化

SD卡读写有两种模式：SD模式和SPI模式。其中SD模式又可以分为1bit和4bit两种传输模式。当前集中器使用的是SPI模式。

SD卡指令由6个字节(Byte)组成。Byte1为0 1 x x x x x x(命令号，由指令标志定义，如CMD39为100111即16进制0x27，那么完整的CMD39第一字节为01100111，即0x27+0x40)。Byte2-5为命令参数， Byte6前7位为CRC校验位，最后一位为停止位0。

SD卡上电后处于空闲状态，上电延时大于74个时钟周期后，主机发送CMD0指令复位SD卡，如果复位成功则返回0x01。SD卡复位时序如图8.1所示。

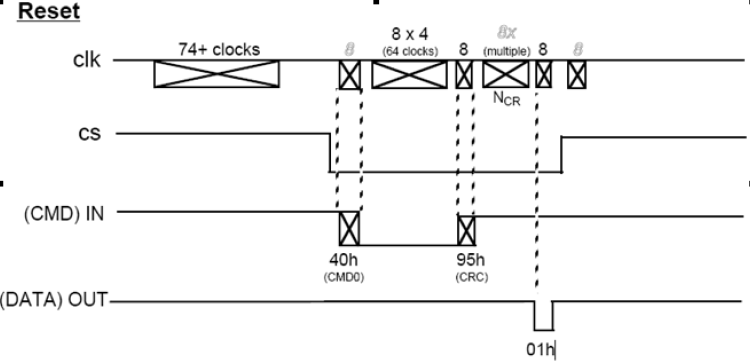


图8.1 SD卡复位时序图

SD卡复位成功后，通过CMD55和ACMD41判断当前电压是否在卡的工作范围内，如果卡返回0x00说明工作正常。

SD卡复位成功并且验证工作正常后，主机可以继续通过CMD10读取SD卡的CID寄存器，通过CMD16设置数据块长度，通过CMD9读取卡的CSD寄存器。从CSD寄存器中，主机可以获知卡容量，支持的命令集等重要参数。

SD卡是否存在及工作是否正常，可以通过SD卡复位指令CMD0及工作电压验证指令CMD55+ACMD41来实现。

## 8.2、SD卡读写验证

集中器在使用过程中，遇到过SD卡上锁的情况，如图8.2所示。SD卡上锁后可正常检测和读出数据，但数据不能保存。

为了避免这种情况发生，在初始化SD卡完成后，向卡中写入某数据并读出，对比写入读出是否一致，如果不一致则指示SD卡读写异常。因为SD卡擦写次数为100000次，如果在集中器工作过程中频繁读写检测SD卡，则会影响SD卡使用寿命，所以仅在初始化时检测SD卡读写情况。



图8.2 SD 卡锁写入保护

## 8.3 SD卡检测及显示流程

SD卡在初始化中检测是否正常，并在LCD TOP任务中显示，处理流程及屏显方式如图8.3所示。如果使能了蜂鸣器报警功能，则同时发出蜂鸣声提示。



图8.3 SD卡状态检测及显示流程图

## 8.4 备用方案

SD卡插座的10号引脚用于卡写保护识别，11号脚用于卡插人检测识别，如图8.4。相应更改硬件中SD卡座相关连线，通过这2个引脚可以检测到SD是否插入及SD卡写保护锁是否锁上，但不能识别卡是否有问题。个人认为此方案虽然直接简单，但不够稳妥。



图8.4 SD卡插座引脚

# 9 蜂鸣器

应工程及市场需求，集中器中加入蜂鸣器，用来及时以不同蜂鸣声的形式提示不同异常发生。蜂鸣器相关原理图如图9.1所示，FMQBJ管脚连接STM32控制器的PB5脚。蜂鸣器为电压驱动型，给蜂鸣器上电则发出声音，断电声音停止。

蜂鸣器相关流程如图9.2所示。

蜂鸣器的使用在嵌入式软件中有以下2点需要注意：

* + - 蜂鸣器控制过程，要放在较高优先级的任务中（如WDT任务中），以防蜂鸣时间受其他任务影响，造成错误声提示。
    - 蜂鸣器功能默认不使能，需要按键选定使能后才起作用，以防止平时集中器发生故障，蜂鸣响起影响业主。

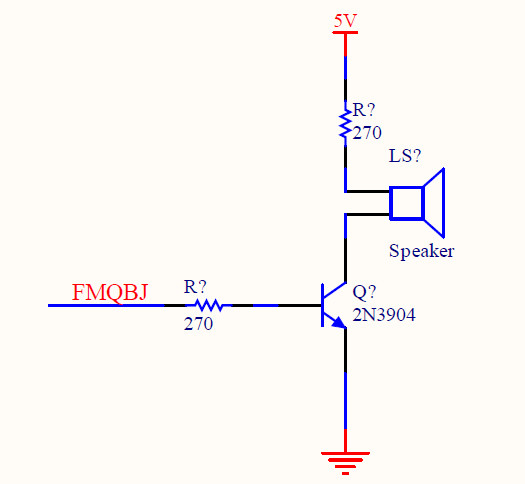


图9.1 蜂鸣器相关原理图



图9.2 蜂鸣器相关流程图

# 10 RTC用纽扣电池异常检测

如图10.1，当前集中器实时时钟（RTC）使用的是RX8025，配有纽扣电池当集中器断电后维持正常时钟。

为更可靠的判断纽扣电池是否正常（是否存在、是否有电等），在程序中加入相应检测，通过判断刚“上电开机”时的时间是否与RX8025的出厂设置时间相近判断。

实际测试发现，RX8025掉电时时间停留在掉电时的时间值，并没有恢复为出厂设置时间，所以，此方案不能实现。

另外，查RX8025手册，发现REG-F寄存器中/XST位用来指示是否有震动停止发生，但这个位掉电不可保存，也不能满足我们的计划需求。

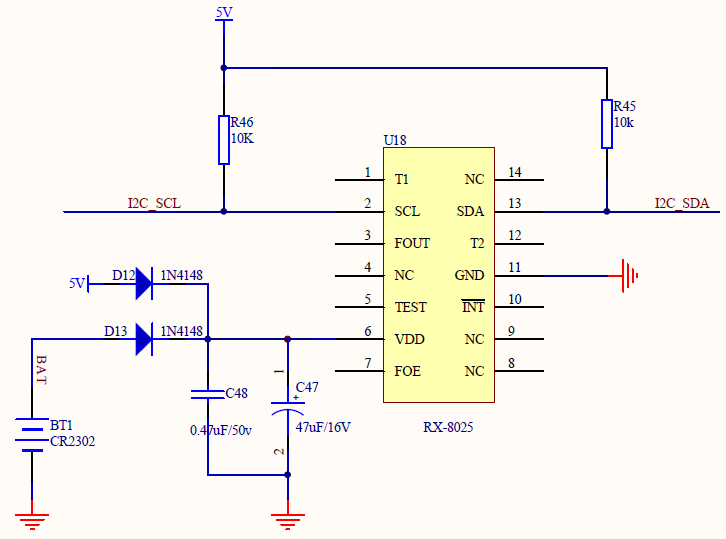


图10.1 RX8025实时时钟相关电路原理图