智能手表市场是当前智能穿戴设备市场的一个重要组成部分。根据市场调研机构IDC的数据显示，2021年全球智能手表出货量达到了1.14亿台，同比增长16.9%。随着人们对健康的关注度不断提高，智能手表的功能也不断升级，从最初的时间显示、计步、心率监测，到现在的睡眠监测、血氧监测、健身运动、通讯等多种功能，智能手表已成为人们日常生活中不可或缺的一部分。预计到2025年，全球智能手表市场规模将达到310.1亿美元。

然而，现有的智能手表产品大多功能过于杂乱，使用体验不佳，价格昂贵，不适合普通消费者使用。同时，市场上存在着许多基于封闭式设计的智能手表产品，缺乏开源和可定制的特点，限制了智能手表的应用和推广。

基于调查，人们对智能手表的需求主要体现在以下几个方面：

健康监测：智能手表可以监测人体健康状况，如心率、血氧、血压、睡眠等数据，提醒用户及时关注自身健康状况。

运动追踪：智能手表可以记录用户的运动数据，如步数、跑步里程、卡路里消耗等，帮助用户更好地管理自己的运动状态。

社交娱乐：智能手表可以支持社交软件的消息推送和简单回复，还可以播放音乐、控制智能家居等，提供更多娱乐和便利功能。

跟踪功能：智能手表可以通过蓝牙连接手机，实现防丢和寻找手机等功能，避免用户遗失手机或者手表。

L-ink Watch作为一款基于开源硬件和物联网技术的智能手表，具有更高的可定制性和价格优势，可以满足更多消费者的需求。L-ink Watch除了支持基本的时间显示、数据传输等功能外，还具有柔性墨水屏幕、低功耗设计、开源硬件等特点，可以支持更多的个性化定制和二次开发，满足用户的多种需求。

通过采用柔性墨水屏作为显示器，L-ink Watch可以实现信息的实时更新和展示，延长电池使用寿命，同时支持多种显示效果和动画效果，增强了信息的表现力和吸引力。此外，L-ink Watch采用低功耗设计，可以实现更长的电池寿命，同时保持良好的显示效果和使用体验。

此外，区别于其他手表的最本质区别在于，L-ink Watch实现了NFC传输数据功能，并且能够将多张NFC卡集成，解决了手机NFC无法进行加密卡以及半加密卡的复制，极大的提高了手表功能的扩展性。

综上所述，L-ink Watch作为一款基于开源硬件和物联网技术的智能手表，具有更高的可定制性和价格优势，能够满足消费者对于智能手表的多种需求，并为开发者和创客提供更多的可能性。在当前智能手表市场竞争激烈的情况下，L-ink Watch的出现将为市场注入新的活力和创新力，成为市场上备受关注和认可的一款产品。

**低功耗硬件方案**

L-ink Watch项目采用了STM32L051C6作为主控制芯片，ST-ST25DV64KC作为NFC读取芯片，XC6206P332MR作为电源管理芯片，低功耗时钟芯片DS1302。针对这些芯片的特点，L-ink Watch采用了多种低功耗设计方案，以尽可能地延长设备的电池寿命。此外显示器选择方面，我们区别于传统电子手表的LED显示屏，我们选取电子墨水屏作为显示设备，优化用户产品体验。

1、L-ink Watch采用了STM32L051C6作为主控制芯片。该芯片具有多种低功耗模式，例如待机模式、休眠模式和停止模式等，可以根据应用需求选择合适的模式以最小化能耗。此外，为了进一步降低能耗，L-ink Watch还使用了优化的时钟设置，例如采用低频晶振以降低系统时钟频率，以减少能量消耗。此外该芯片在运行时具备极低的功耗，与传统的低功耗MSP430芯片进行对比如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 运行模式 | 待机模式 | 停止模式 | 时钟频率 |
| MSP430 | 1 MHz、2.2 V 时为 250 µA | 0.7 µA | **0.1 µA** | 16 MHz |
| STM32L051K8 | **88 μA/MHz** | **0.27 μA** | 0.4 μA | **32MHE** |

1. L-ink Watch采用了ST-ST25DV64KC作为NFC读取芯片。该芯片具有超低低功耗，并且能够进行能量转换，将磁场能量转化为电能，在保证读写侠效率的同时进一步的降低系统的功耗。同时该芯片具有多种功耗模式，可以根据要求选择模式以最小化功耗。为了最小化NFC读取的功耗，L-ink Watch还使用了高效的NFC读取算法以及优化的NFC读取频率和时序等参数。

3、L-ink Watch采用了LP2985-10DBVR作为电源管理芯片。该芯片具有以下特点：

1.低静态电流：LP2985-10DBVR芯片具有非常低的静态电流，只有0.5μA，适用于低功耗系统设计，可以延长电池寿命。

2.低压差：LP2985-10DBVR芯片的工作电压范围为2.5V到16V，输出电压为1.0V，具有很低的压差，可以提供更为稳定的电源。

3.过载保护：LP2985-10DBVR芯片具有过载保护功能，可以保护芯片和负载免受过载电流的损害。

4.短路保护：LP2985-10DBVR芯片还具有短路保护功能，可以在负载短路时自动关闭输出，避免负载和芯片的损坏。

同时，为了优化用户体验，我们将传统的纽扣电池换为了可充放电的蓄电池，采取TP4056芯片作为充电管理芯片，在保证92%的充电效率外，拥有低至2uA的静态电流。此外选取合适的电感和电容，优化开关频率以最小化系统的能耗。

L-ink Watch采用了多种低功耗设计方案，包括优化的时钟芯片设置、高效的NFC读取算法、合适的电感和电容以及选择低功耗显示设备等，确保在不影响用户体验下使用最低功耗。

### 2.2低功耗设计

根据与系统功耗有关的因素，可引入以下5个主要的优化措施：选择低功耗的电源供给电路、DPM（Dynamic Power Management，动态电源管理）、DVS（Dynamic Voltage Scaling，动态电压缩放）、低功耗硬件设计、低功耗软件设计。

### 2.2.1选择低功耗的电源供给电路

(1)开关电源

开关电源可分为 AC/DC 和 DC/DC 两大类，在便携式电子产品中，开关稳压电源主要指 DC/DC 变换器。DC/DC 转换器包括升压、降压和反相等电路，DC/DC电源芯片的优点是转换效率极高，在压差较大和输出电流较大的时候不易发烫，但相对而言，DC/DC 电源芯片的纹波系数会相对较高。

(2)LDO（Low Dropout Regulator，低压差线性稳压器）

LDO 电源芯片最大的特点就是低成本、低噪声和低静态电流。此外，LDO 还能保证电路转换前与转换后的电流大致不变，但是 LDO 只能在降压电路中使用。由于 LDO 良好的抗干扰能力，在一些电路设计中采用 LDO 电源芯片来实现有源滤波，不仅可以极大的提高输出电压的精度，还可以提高供电系统的效率。

我们在此项目中采用可充电的蓄电池进行供电，取LP2985-10DBVR作为电源管理芯片，采用TP4056作为充电电路管理芯片。

### 2.2.2动态电源管理

动态电源管理是通过动态的配置系统使其通过最少的活动部件来为到来的请求服务，以节省电能。动态电源基本原理是有选择性地关闭空闲的系统部件或降低其性能，在需要服务时再重新打开，达到减少功耗的目的。

### 2.2.3动态电压缩放

动态电压缩放是基于器件工作电压越高，则功耗越高的原理，通过合理地调节器件工作时的电压来降低功耗。当然在调节器件工作电压时首先要分析系统的工作状态，而不是一味地去降低电压，如果为了降低功耗而使系统工作在不稳定的状态，那就适得其反了，所以要寻找在保证系统正常情况下的最低电压。

### 2.2.4低功耗硬件设计

动态电源管理和动态电压缩放是从系统供电的电压上着手，通过合理的变换电压以及供电状态来最大限度的限制电压利用率来降低系统的功耗。除此之外，可以采用带低功耗的处理器、低功耗的外部器件以及带有低功耗模式的系统三种方法来实现系统硬件的节能设计，并且可以对硬件的设计进行低功耗方案设计。

(1)选择低功耗的MCU

(2)尽量降低MCU的工作频率和系统电压

(3)外围电路尽可能选择静态功耗低的器件

(4)合理处理不用的I／O口

(5)设计外部中断唤醒电路

(6)选择合适的MCU的工作模式

### 2.2.5低功耗软件设计

通过硬件上的选型选择带有低功耗模式的MCU，通过MCU工作模式的合理转变可以有效地降低整个系统的功耗。此外，编程设计时的代码优化也是低功耗软件设计上容易被人们所忽略的一个部分，因为代码的优化不像软件整体设计或者是硬件电路设计上那么直观，容易让人察觉到其功耗的耗损。具体分析如下：

(1)根据 MCU 工作模式的转变降低功耗：

ST-STM32L051k8是一款低功耗微控制器，具有多种功耗模式：低功耗运行模式、低功耗待机模式和 停机模式，以及低功耗休眠模式，当MCU不需要进行数据处理以及通信的时候，为了降低功耗，我们将将其工作模式切换到低功耗运行模式、低功耗待机模式或者低功耗休眠模式

（2）根据优化代码质量来降低功耗：

硬件系统设计上的考虑只是为降低功耗提供了条件，真正实施要通过软件来完成，要想实现对芯片的访问以及 少CPU的运行时间 完成对信号的转换都是由软件来控制的。因此，软件设计单片机应用系统设计中，必须要考虑低功耗的问题。软件设计的基本思想就是尽量缩短MCu的运行时间，让Mcu能大部分时间处于休眠状态，系统的各个部分在需要的时候再让其运行。

另外，在微处理器执行的程序中，每一条指令都将激活微处理器中的某些硬件部件。因此，可以认为每一条指令都有一个固定的功率消耗量，正确选择指令可以降低微处理器的功耗。我们通过以下方法降低软件功耗：

①让 I/O 模块间歇运行；

②减少 CPU 的运算量，在进行数据读取方面，我们尽量采取DMA进行直接读取内存；

③我们设置了多个中断，只有在中断开启时才唤醒某些外设以降低功耗；

④硬件定时器延时的方法来编写延时程序，以减 真正实施要通过软件来完成，要想实现对芯片的访问以及少CPU的运行时间

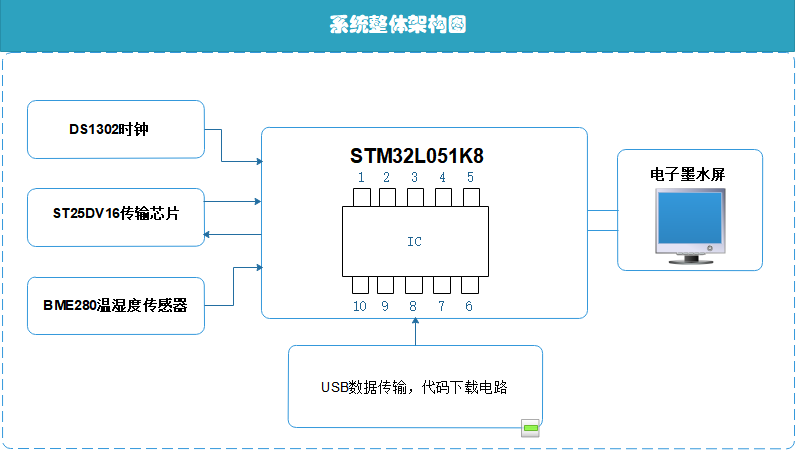
我们所选取的ST-STM32L051C6具有超低功耗的特性，具有 5 个可选阈值的超安全、低功耗 BOR（掉电复位），超低功耗 POR/PDR，内部低功耗 37 kHz RC，内部多速低功耗 65 kHz 至 4.2 MHz RC，并且具有超低功耗外设：12 位 ADC 1.14 Msps，最多 16 个通道（低至 1.65 V），2 个超低功耗比较器（窗口模式和唤醒功能，低至 1.65 V）以及9x 定时器：1x 16 位最多 4 个通道，2x 16 位最多 2 个通道， 1x 16 位超低功耗定时器、1x SysTick、1x RTC、1x 16 位基本和 2x 看门狗（独立/窗口），在我们产品的低功耗需求中提供了巨大的帮助！

# 三、系统总体框架与硬件电路设计

## 3.1计划实现功能与设计框图

本设计最终目的是实现一个集成NFC卡片切换、计时定时、温湿度监控以及作为数据扩展屏功能的高精度，低功耗，便携的L-ink Watch手表。

本系统以超低功耗芯片STM32L051K8U6（32引脚）为主控芯片,其具有32M的控制主频，选择低功耗是时钟芯片DS1302进行高精度计时功能、选取具有能量采集功能的ST25DV16芯片作为NFC功能实现以及温湿度传感器BME280对周围环境进行检测，根据需要开启相应芯片，采集对应数据后传输到主控芯片上进行数据处理。而后使用耗电极低的电子墨水屏作为显示设备，将数据进行显示。并且预留与电脑上位机交互的Type-C接口，供实现数据显示扩展功能。系统总体设计框图如图所示：



## 3.2硬件电路设计

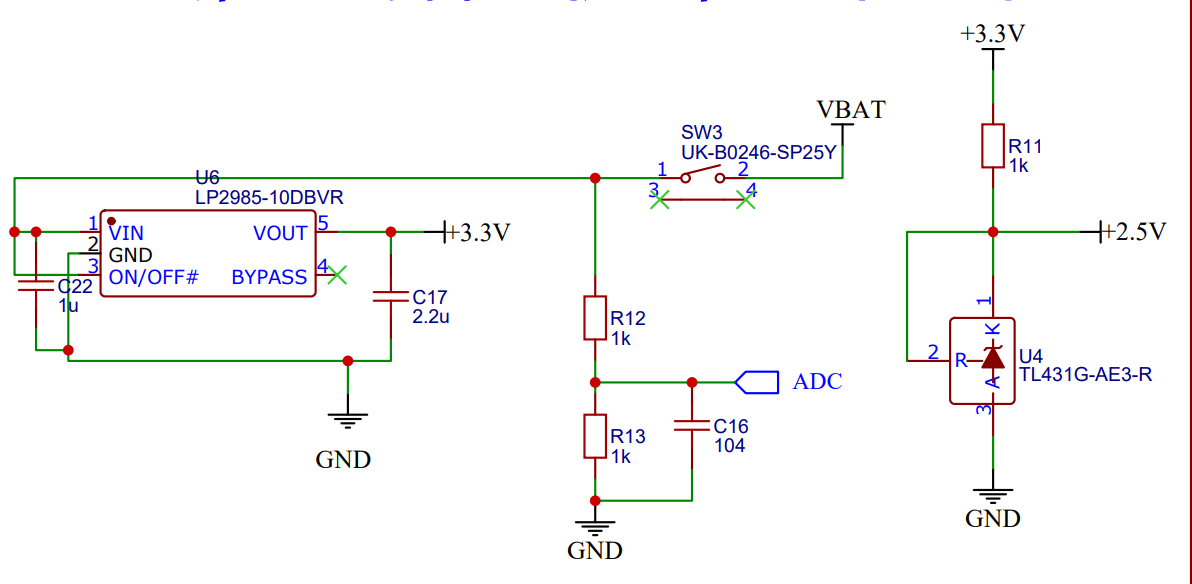
### 3.2.1系统电源硬件设计

本系统外部供电电源采用3.7V锂电池，而选用的主控芯片STM32L051K8U6所需的供电电压为3.3V，ST25DV16芯片所需供电电压为3.3V，计时芯片DS1302所需供电电压为3.3V,以及电子墨水屏驱动所需电压也为3.3V,温湿度传感器需要3.3V供电，所以整体电源进行管理设计时，只需要针对锂电池充电以及3.7V转3.3V降压设计即可，具体分析及电路设计如下：

①MCU供电电路：

根据电源芯片分类介绍，3.7V降到3.3V应该选取LDO电源芯片。在选取芯片之前，要对MCU所需及提供的电流大小进行分析，即MCU自身所需电流以及周围电路所需电流的总和的最大值。如果选取的电源芯片输出电流的大小小于MCU所需的电流大小，很可能导致系统无法正常工作。因此，正确的分析也为后期调试时排除了可能出现的硬件问题。

由于我们所选取的芯片时ST家族的超低功耗STM32L051K8，其运行电流低至2.816MA，由于其他芯片所需要的供电电压也为3.3V，所以我们将输出电压与其他芯片进行连接，这里我们选用的芯片为LP2985-33DBVR芯片，芯片是精度高、噪音低、采用CMOS生产工艺的 LDO电压调整器芯片，提供 1% 容差的精密输出电压。此系列芯片具有可特定选择输出电压的芯片，满足我们需求，具有低输出噪音，低输入输出压差和并且静态电流低，功耗小的特点。且其输出电流达到了150ma，满足我们的需求。但是在某些情况下，输出电压并不能保持一定为3.3V，所以我们提供了一个基准电路，作为某些功能的稳定参考电压。具体电路设计如图2-2所示：



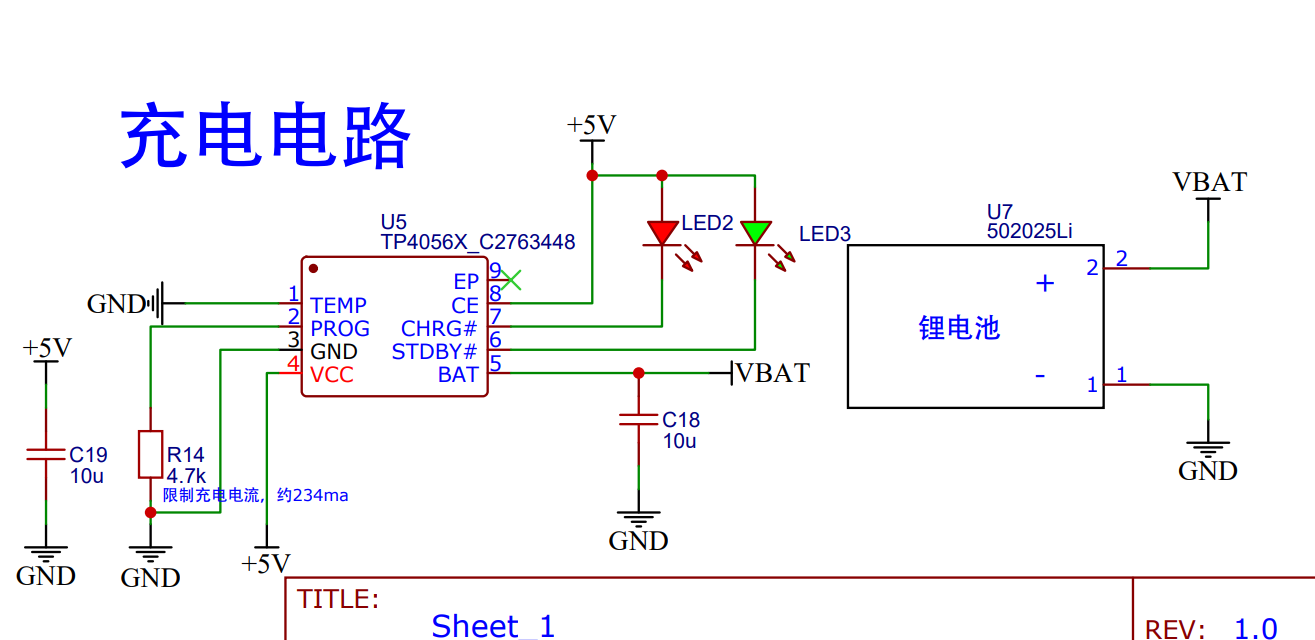
在这一部分电路中，我们将锂电池提供的3.7V电压通过LDO电路转为3.3V，并对其进行了电压基准，其中我们还对电池电压使用ADC进行可选择性电量监控，保证电池的长久使用。

②充电电路：

为了提高系统的可持续性，可以采用可充电型锂电池，并设计相应的充电电路。选用锂电池线性充电管理芯片 TP4056，该产品具有耐高输入电压、过温、过流保护、过压保护、内置充电流程控制等特点。其中该芯片带有充电状态指示功能，当电池充电完成时,STDBY被内部开关拉到低电平，绿灯亮起，表示充电完成。除此之外，STDBY管脚将处于高阻态。当充电器向电池充电时，CHRG引脚被内部开关拉到低电平，红灯亮起，表示充电正在进行；否则CHRG管脚处于高阻态。此芯片具备低功耗省电特性，如果电池电压低于2.9V，TP4056 用小电流对电池进行预充电。当电池电压超过2.9V时， 采用恒流模式对电池充电，充分的保护了电池。

芯片内部的高精度电压基准源、误差放大器和电阻分压网络确保电池端调制电压的精度在1% 以内，满足锂离子电池和锂聚合物电池的要求。当输入电压低于欠压锁定阈值电压或者输入电压低于电池电压时，充电器进入低功耗的睡眠模式，此时电池端消耗的电流小于2uA。我们在设计充电电路时，通过改变电阻R14的值来限制充电电流，防止锂电池产生过热，避免多余的能量消耗。

充电电路如下图所示：

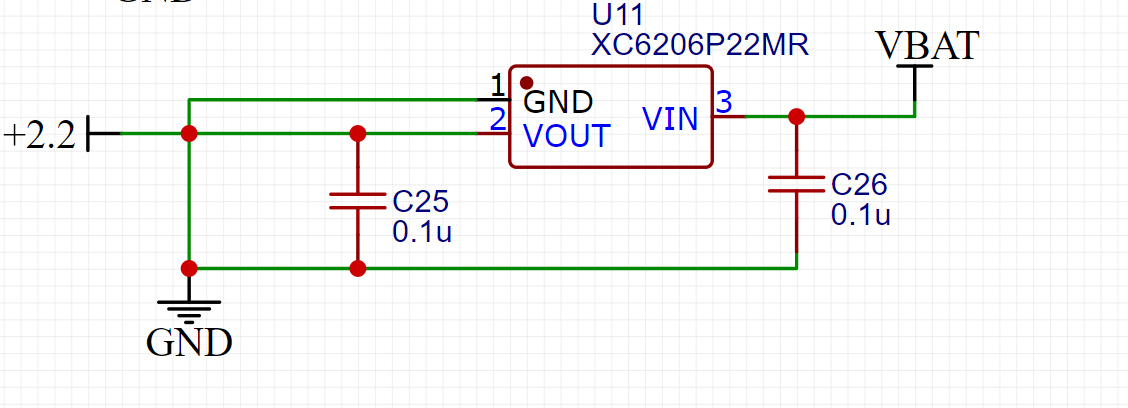


③传感器芯片电源管理

在本产品的实现中，我们发现其他传感器芯片的供电电压均可为3.3V，可直接由MCU供电电路提供，但是为了最小化低功耗，我们考虑到时钟芯片需要一直进行，其芯片工作在2.0V条件下使用电流小于300nA,所以在此我们选择3.7V-2.2VLDO电路单独，选用低功耗、高效率芯片：XC6206P22MR芯片。

XC6206P系列是高精度、低功耗、低压差，3端CMOS降压型电压稳压器。XC6206P系列在输入输出电压差极小的情况下提供200mA的输出电流，并且仍能保持良好的调整率。集成了过流保护和短路保护。

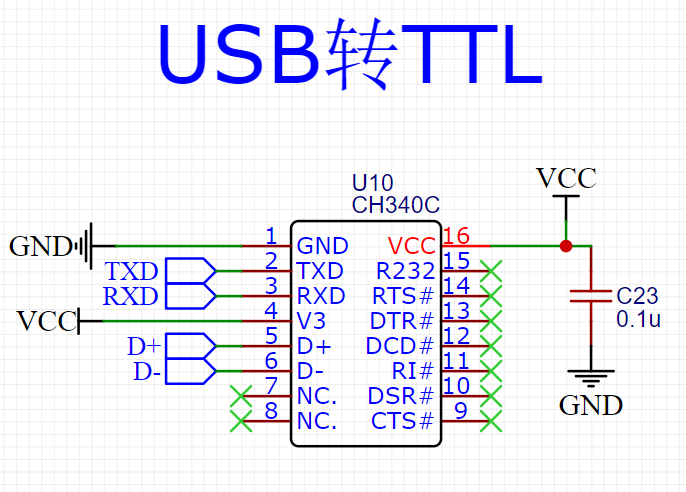
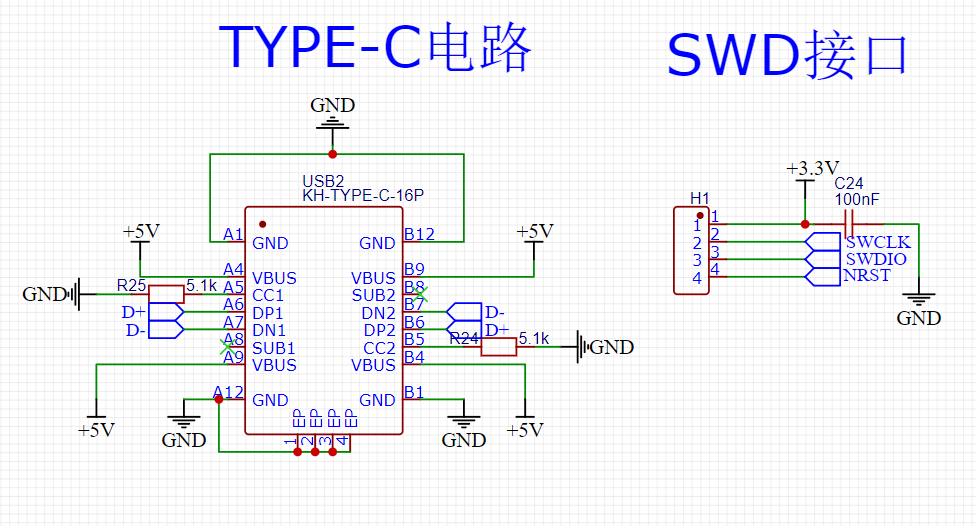
具体设计方案如下：



### 3.2.2 MCU程序下载电路设计

为方便用户操作，我们设计了两种程序下载电路方式，其通过按键BOOT0的切换可以进行选择，分别由SWD下载以及直接通过TYPE-C接口电路进行程序下载，为了实现这一功能，我们添加了USB转TTL芯片CH340C，更方便上位机对于产品的功能调试与交互。

具体实现电路如下：



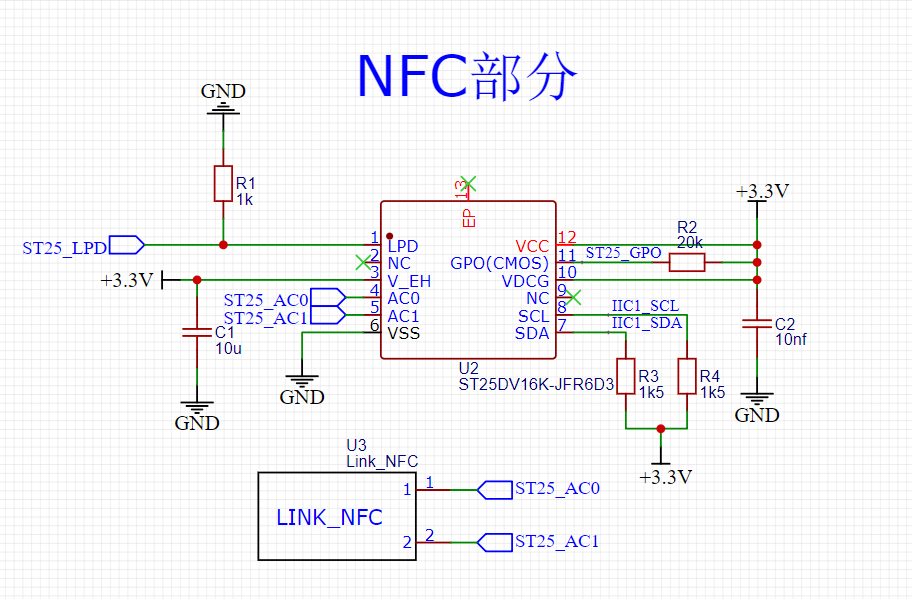
### 3.2.3传感器电路设计

①ST25DV16实现NFC数据传输

NFC数据传感器选用ST25DV16K芯片进行工作，此芯片具有超低功耗特性没并且可以根据周围RF场的变化进行能量捕获与数据传输，该芯片采用低功耗传输协议ISO/IEC 15693,NFC传输的数据保存在芯片内部信箱，通过SCL与SDA进行IIC协议与MCU通信实现数据传输，并且我们将捕获的能量用于MCU供电以节省电量消耗，为了实现软件上的功耗降低，我们将GPO中断接入MCU，实现特定功能下的功耗选择。

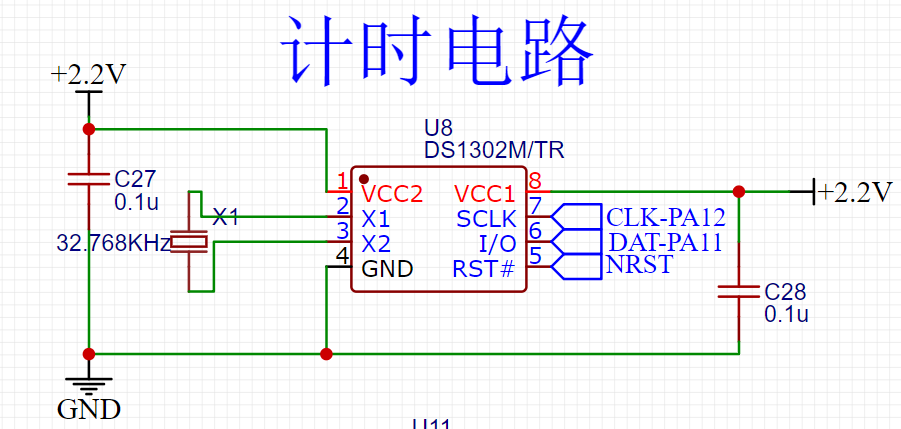
此外，为了能够实现加密卡的复制，将卡片的天线与ST25DV16天线引脚AC0,AC1共用，并且通过波轮通过天线的连接与否来确定卡片的选择。

具体电路设计如图2-8所示：



②DS1302传感器

DS1302是一款低功耗实时时钟芯片，支持多种定时器模式和闹钟模式，可以实现复杂的定时和闹钟功能，内置低功耗晶体振荡器（RTC Crystal），提供高精度的时钟信号，我们使用外接频率为32.768KHz的晶振作为时钟基准源，并且通过SPI协议与MCU进行数据传输，为了降低功耗，我们使用2.5V作为供电电源，使其能够长时间工作并且保证一定的精度要求。在进行电路设计时，为了避免由于电源和线路噪声所造成的可靠性和精度问题，我们接入滤波电容使得电路更加稳定可靠。



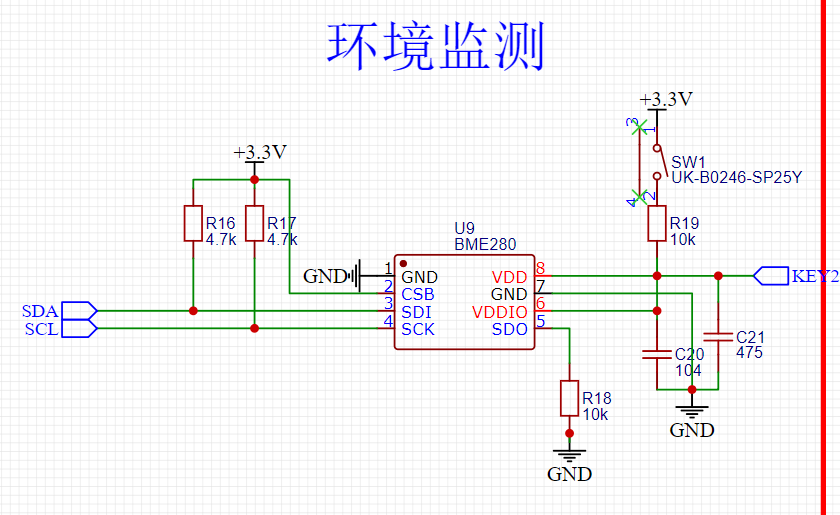
③温湿度传感器

BME280传感器芯片采用了Bosch Sensortec公司的先进技术，具有较高的灵敏度和准确性。该传感器模块内部集成了一个高精度的测温元件、一个感湿元件以及一个气压传感器，能够在同一时间对三种环境参数进行实时测量。传感器采用了高度集成化设计，体积小巧且功耗低，特别适合于便携式设备和物联网应用。

BME280的温度测量原理基于热电偶效应，传感器内部集成了一个微型热电偶元件，可以感知周围环境的温度变化。当温度发生变化时，热电偶元件会产生微小的电压信号，通过微处理器的数字信号处理电路进行放大、滤波和校准，最终输出高精度的数字温度值。

BME280的湿度测量原理基于电容式测量，传感器内部集成了一个微型电容元件和一个电容传感器，可以感知周围环境的湿度变化。当湿度发生变化时，电容元件会产生微小的电容变化，通过电容传感器的数字信号处理电路进行放大、滤波和校准，最终输出高精度的数字湿度值。当需要获得传感器数据时，我们通过中断唤醒MCU与BME280，通过IIC协议传输所需数据。

具体电路设计如下：

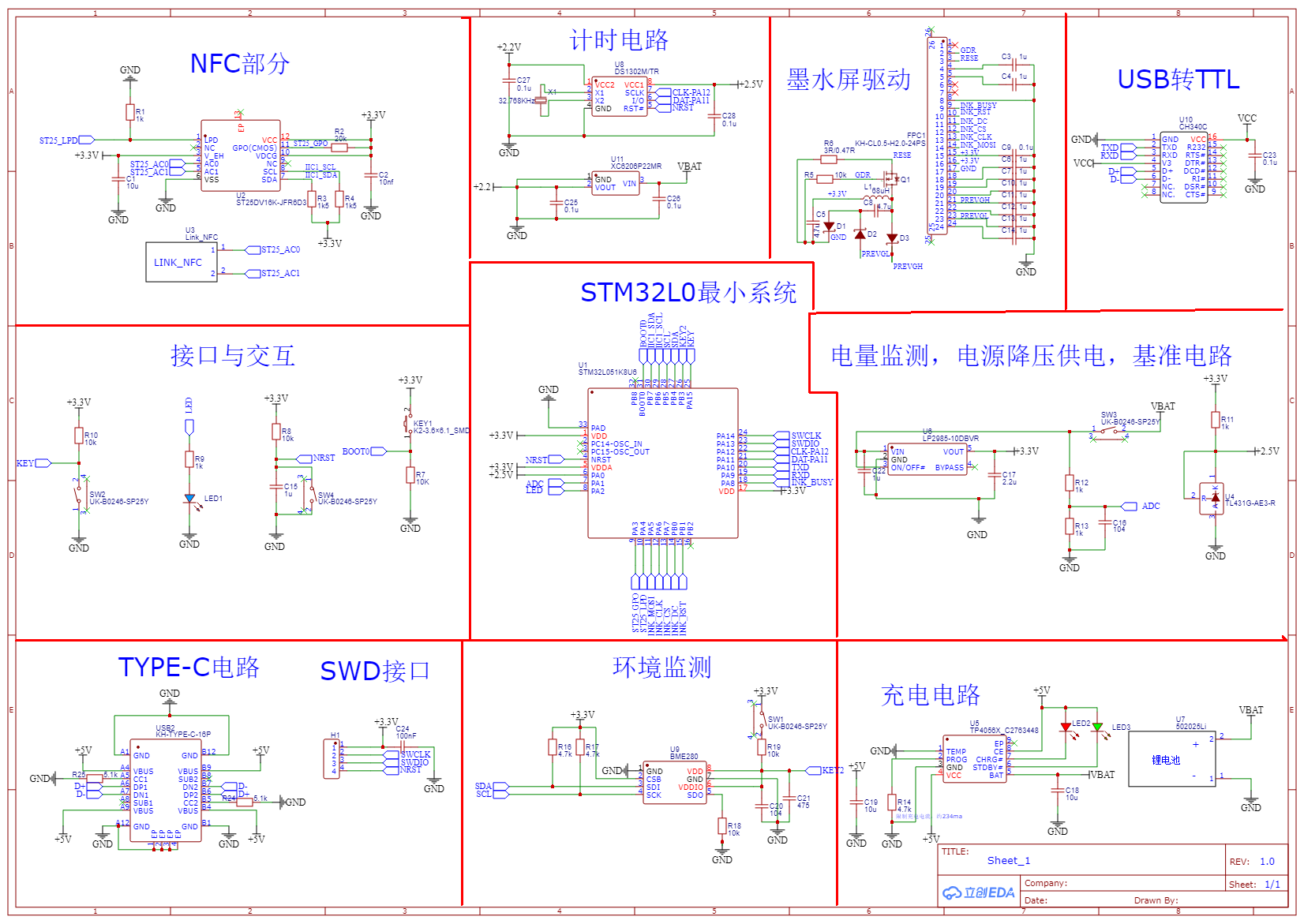


### 3.2.5系统完整电路图

本系统的实时时钟采用DS1302外接的32.768KHZ晶振，32.768KHz经过2的15次方分频后是1Hz，正好是1秒的方波脉冲，可用来计时。对于石英钟而言，其内部有15级二分频器，进行15 次分频后是1HZ 的秒信号。

系统复位电路采用低电平按键复位，在没有按下复位按键时，RST引脚时钟为高电平，但当按键按下时，RST引脚变为低电平，系统实现复位。

MCU 外围整体电路，包括3种传感器电路，电源电路，USB下载模块以及按键交互电路总原理图如图所示：

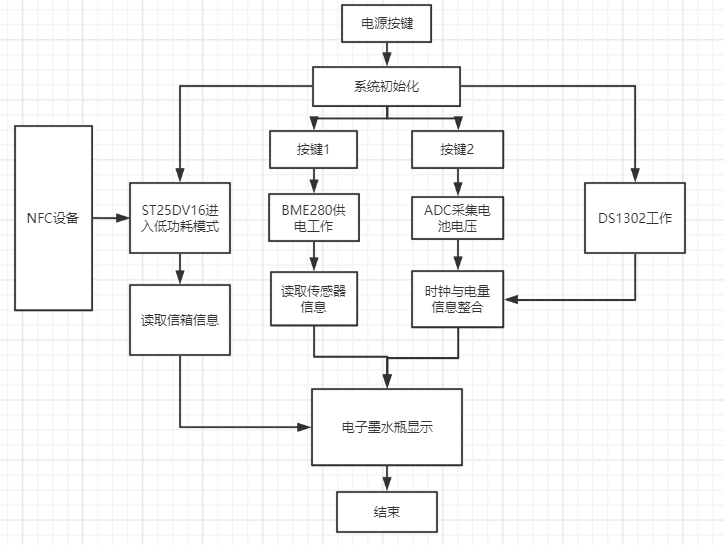


# 系统软件设计

## 4.1 系统整体程序设计思路

1. 我们发现在STM32L051芯片中具有多种功耗模式，其中 Standby 模式是最低功耗的模式，可以将系统功耗降至几微安以下。在 Standby 模式下，系统时钟和外设都会被关闭，只有 RTC 和 EXTI 引脚可以唤醒系统。但是我们需要对外设进行中断设置，所以在这使用功耗相对较低的STOP模式，在需要执行任务时通过 EXTI 中断唤醒系统。
2. 我们将会充分的利用定时器与中断功能，我们设置了三个功能按键，用于实现我们产品的中断，在不使用该模块时，我们的开关断开，并且将芯片都设置为低功耗模式，当按键按下时，使用中断来唤醒下位功能，以最大限度节省功耗。并且在不需要使用某个外设时，关闭外设的供电电源，减少不必要的功耗损耗。
3. 我们可以使用 STM32L 系列微控制器内置的 ADC 模块。ADC 模块可以将模拟信号转换为数字信号，并产生中断信号。我们可以将 ADC 的中断信号连接到 EXTI 引脚上，从而在需要读取电压数据时唤醒系统执行相应的任务。同时，我们可以使用定时器来控制 ADC 的采样频率，以保证系统功耗的最小化

系统整体流程图设计如下：



## 4.2 各传感器程序设计

### 4.2.1 ST25DV16传感器模块原理及程序设计

（1）ST25DV16与STM32L051通信

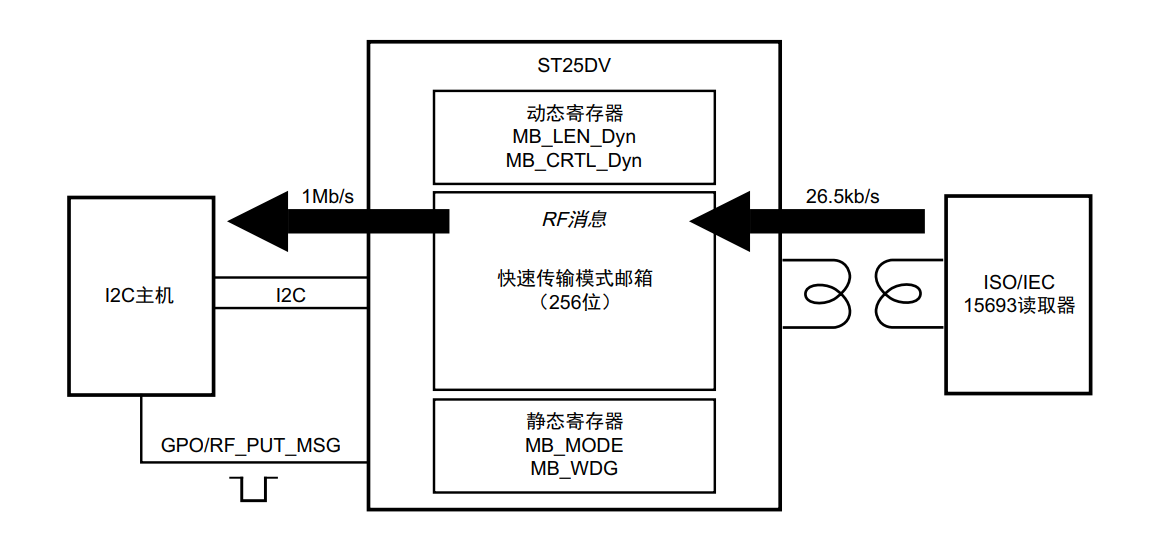
STM32L051和ST25DV16芯片之间可以通过I2C进行通信，当ST25DV16芯片中的信箱收到信息时，其中断引脚GPO变会拉低，我们将MCU的其中一个GPIO设置为中断引脚，使用IIC协议读取与发送所期望的数据与命令，并且在未有数据通信需求时，我们利用ST25DV16的LPD低功耗引脚使得芯片在低功耗模式下运行，只有检测到信箱中的数据变化，并且产生中断时才与MCU进行通信，最大化降低功耗。而后将得到的数据通过电子墨水屏显示出来。

其通信流程图如下：

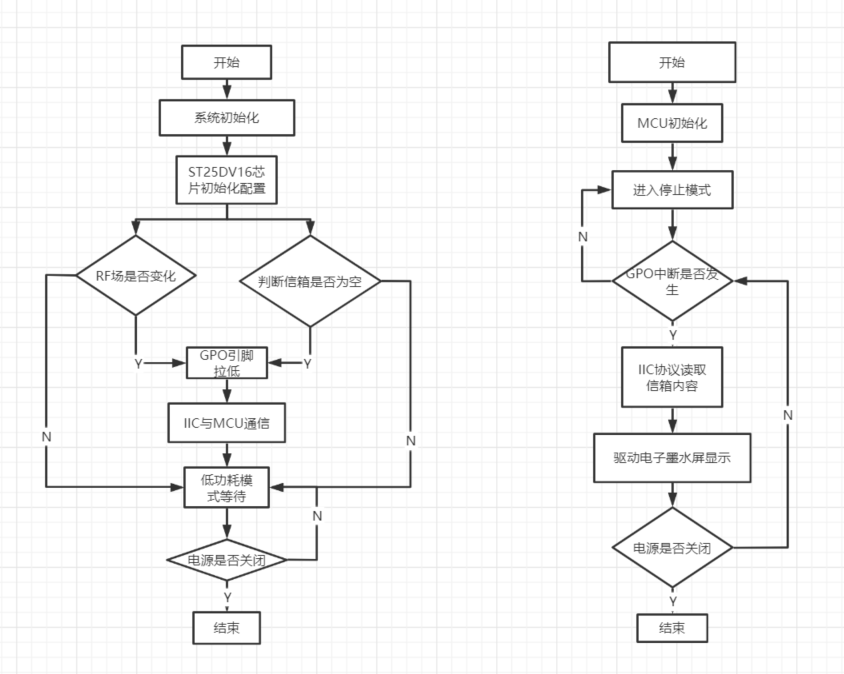
（2）ST25DV16与NFC设备通信

ST25DV16芯片与其他的NFC设备进行通信时，所采取的协议为ISO/IEC 15693,在外部RF场未变化时，ST25DV16芯片可以通过LPD引脚一直处于低功耗模式，当RF场发生变化时，ST25DV16芯片会自动唤醒，并通过ISO/IEC 15693协议读取的数据保存在芯片内部信箱中。

通信流程如下：



ST25DV16软件设计流程图如下：



### 4.2.2 DS1302时钟模块原理及程序设计

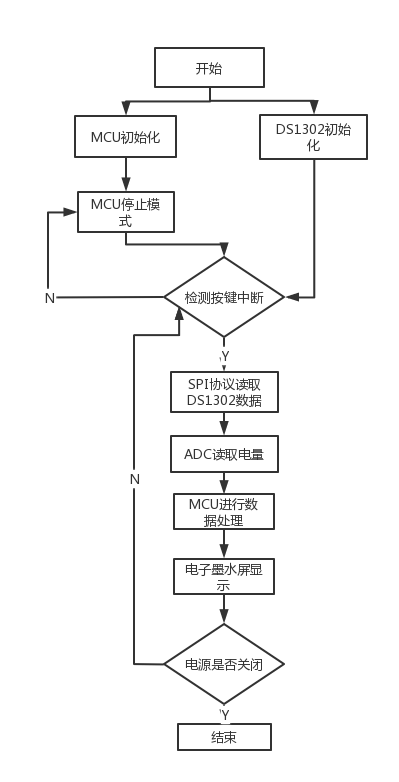
（1）原理

DS1302通过外接一个外部晶体振荡器提供时钟信号，使用32.768kHz的晶体振荡器，可以记录当前时间，包括年、月、日、时、分、秒等信息，同时还支持闰年判断和时钟校正等功能。并且其自身具备极低的功耗，使用SPI协议使得MCU与DS1302芯片进行通信，获取时钟数据显示在电子墨水屏上。

1. 程序设计

我们考虑到由于并不是一直显示时间在电子墨水屏上，所以我们在不需要显示时使得MCU一直处于停止模式，保留EXIT功能，其中DS1302一直处于低功耗运行模式，当按键按下时，由于按键所产生的中断将会唤醒MCU，并通过SPI协议读取DS1302中的时钟、日期数据，其中，我们将ADC显示电量的功能一同接入按键中断，按键按下时，ADC将会运行，读取电池电量，与时钟数据一同显示在电子墨水屏上。

DS1302芯片软件设计框图如下：

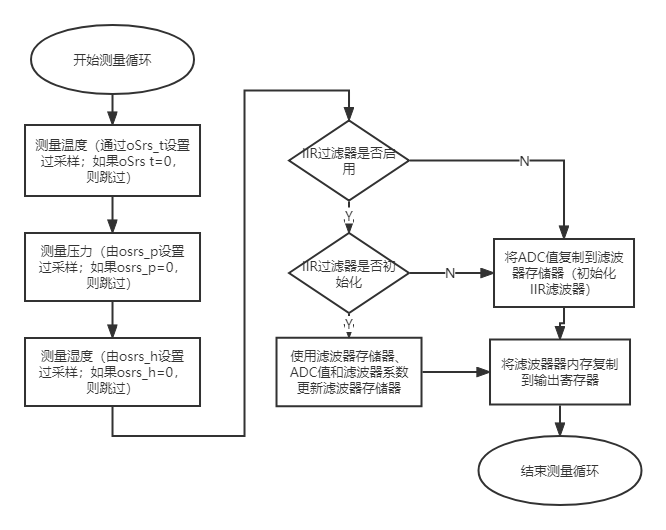


### 4.2.3 BM3290传感器原理及程序设计

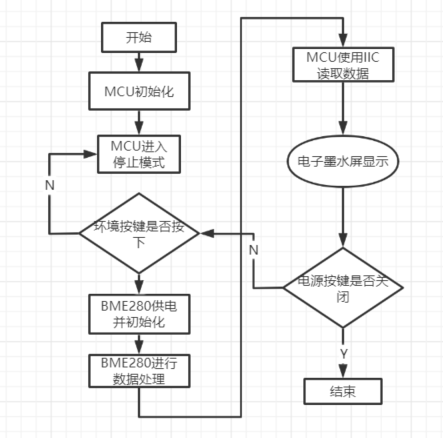
BME280传感器采用IIC数据传输方式，我们通过按键来控制BME280传感器的工作。当系统需要采集温湿度以及气压信息时，我们按下按键，电源会对BME280进行供电，同时MCU会通过IIC协议发送读取数据的命令给BME280，BME280接收到主机发送的开始信号后，再发送响应信号给MCU，并触发一次信号采集。当BME280处于从机模式时，BME280接收到MCU发出的开始信号后会触发一次数据采集。此传感器为低电流消耗 (3.6 μA @1Hz)、长期稳定性和高EMC 鲁棒性。湿度传感器具有极快的响应时间。

BME280采用补偿算法对温度数据进行处理，根据传感器内部的校准参数和补偿公式来计算出实际温度值。同时采用补偿算法对湿度数据进行处理。而后通过IIC协议将处理后的温湿度数据传输给MCU进行显示在电子墨水瓶上。

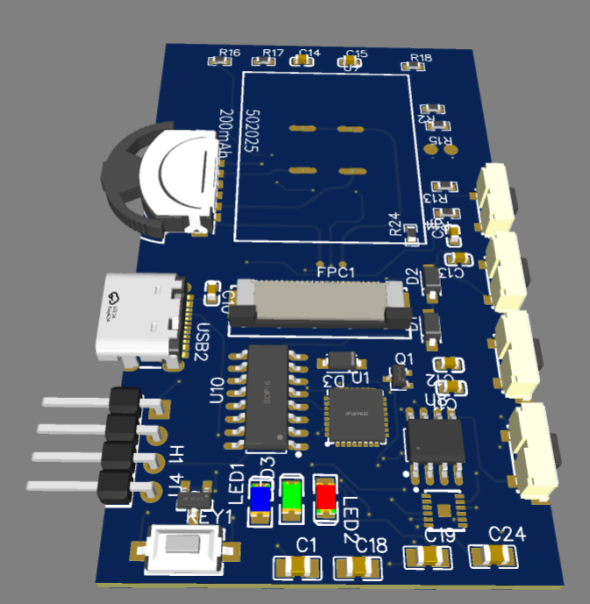
温湿度测量循环如下：



传感器软件设计流程图如下：



产品制作：



# 五、功耗预算

MCU不同工作模式下的电流如下表：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | 模式 | 条件 |  | 标准值 | 单位 |
|  | 运行模式 | 外设开启 | 32MHz | 2.8 | mA |
|  | 停止模式 | 部分外设开启 | 32MHz | 0.4 | uA |
|  | 待机模式 | 外设关闭 | 32MHz | 0.27 | uA |

表5-1 STM32L051K8工作模式

三个传感器工作模块正常工作的耗电电流为：DS1302为300nA,温湿度传感器BME280为2.7mA(最低功耗)，ST25DV16芯片在低功耗模式下其耗电电流为4uA,正常工作模式时为2.3mA,具有非常低的功耗，而当电子墨水瓶显示时所需要的电流消耗为26uA,考虑由于电压转换芯片一直再工作模式，我们所需要输出电流为5ma，所以可以得到其工作电流大概为75uA,,其工作时所需电流为由于我们所采取的程序设定为中断设定，所以需要根据不同功能下进行计算。

当需要MCU工作时，假设MCU一直处于工作状态，并且其外设一直处于工作模式，MCU耗电电流为2.8mA，外设总共耗电为2.8mA+2.7mA+26uA+4uA+2.3mA+300nA+75uA=7.909mA,假设使用170mAh 的可充电锂电池供电，假设本系统一直处于工作状态，可计算出本系统在时间显示下可连续工作21个小时左右。

如果我们根据低功耗方案进行设计，在中断未进行时，我们将MCU处于停止模式，当需要使用时再从停止模式中进行唤醒，其只需要5us唤醒时间。

当时间显示中断发生时，值得注意的ST25DV16一直处于低功耗模式，知道GPO中断发生，此时传感器所需要的总电流为2.8ma+300nA+26uA+4uA+75uA=2.8303mA。假设使用170mAh 的可充电锂电池供电，假设本系统一直处于工作状态，可计算出本系统在时间显示下可连续工作60 个小时左右。

当温度显示中断发生时，此时传感器所需要的总电流为2.8ma+2.7mA+26uA+300nA+4uA+75uA=5.8263mA。假设使用170mAh 的可充电锂电池供电，假设本系统一直处于工作状态，可计算出本系统在时间显示下可连续工作31 个小时左右。

当使用NFC芯片进行数据传输时，GPO中断的发生使得MCU与ST25DV16同时工作，此时传感器所需要的总电流为2.8ma+300nA+26uA+2.3mA+75uA=5.426mA。假设使用170mAh 的可充电锂电池供电，假设本系统一直处于工作状态，可计算出本系统在时间显示下可连续工作31 个小时左右。

当系统中断未发生时，系统设备均处于低功耗模式，MCU处于停止状态，此时所消耗的电流为：0.4uA+300nA+4uA+75uA=80.7uA，此时电池可连续使用2132小时

我们进行具体功耗估算如下（假设一个使用周期为100S）：

未进行低功耗处理时：7.909mA\*10s=790.9mAs=0.2196mAh

进行低功耗设计时，我们需要根据上述中断分开进行计算。

|  |  |  |  |
| --- | --- | --- | --- |
| 功能 | 时间 | 电流(总量) | 电量(mAS) |
| 停止模式 | 95s | 80.7uA | 7.6665 |
| 时钟显示 | 2s | 2.8303mA | 5.606 |
| 环境检测 | 2s | 5.8263mA | 11.6526 |
| ST25DV16传感器工作 | 1s | 5.426mA | 5.426 |
| Tatol | 100s | - | 30.3511 |

平均电流=0.030mA

假设以170maH的电池进行供电，具有5666h的寿命。