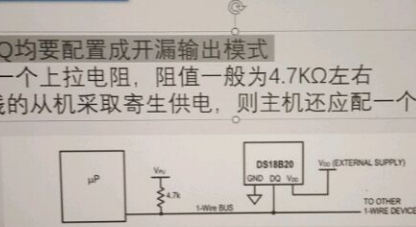


单总线和DS18B20温度传感器

单总线介绍

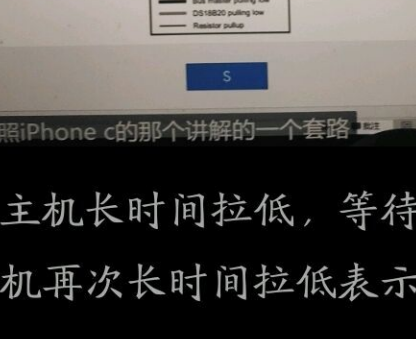
- 单总线 (1-Wire BUS) 是由Dallas公司开发的一种通用数据总线
- 一根通信线: DQ
- 单步、半双工
- 单总线只需要一根通信线即可实现数据的双向传输, 当采用寄生供电时, 还可以省去设备的VDD线路, 此时, 供电加通信只需要DQ和GND两根线



只需要一根通讯线

单总线电路规范

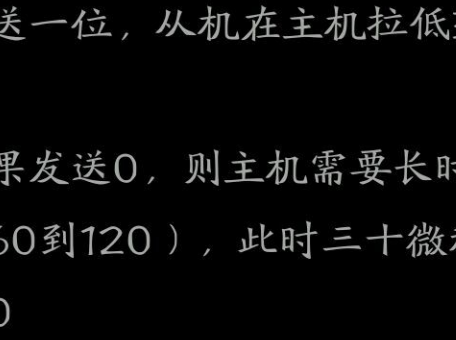
- 设备的DQ均要配置成开漏输出模式
- DQ添加一个上拉电阻, 阻值一般为4.7KΩ左右
- 若此总线的从机采取寄生供电, 则主机还应配一个强上拉输出电路



这一点呢我在他的手册里面没有找到啊

单总线时序结构

- 初始化: 主机将总线拉低至少480us, 然后释放总线, 等待15~60us后, 存在的从机会拉低总线60~240us以响应主机, 之后从机将释放总线

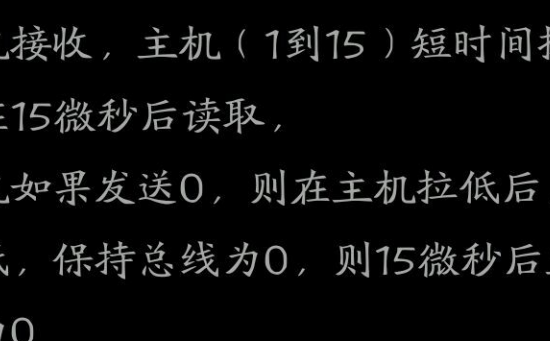


仿照iPhone c的那个讲解的一个套路

初始化, 主机长时间拉低, 等待一小段时间后, 从机再次长时间拉低表示接收到

单总线时序结构

- 发送一位: 主机将总线拉低60~120us, 然后释放总线, 表示发送0; 主机将总线拉低1~15us, 然后释放总线, 表示发送1。从机将在总线拉低30us后 (典型值) 读取电平, 整个时间片应大于60us



S: (BIT)

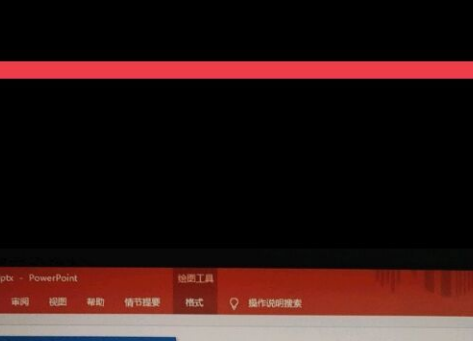
主机发送一位, 从机在主机拉低30微秒后读取,

主机如果发送0, 则主机需要长时间拉低, (60到120), 此时三十微秒后从机读取为0

主机如果发送1, 则主机需要短时间拉低 (1到15), 此时30微秒后从机读取为1

单总线时序结构

- 接收一位: 主机将总线拉低1~15us, 然后释放总线, 并在拉低后15us内读取总线电平 (尽量接近15us的末尾), 读取为低电平则为接收0, 读取为高电平则为接收1, 整个时间片应大于60us



R: (BIT)

接收移位呢其实跟发送移位也差不多

主机接收, 主机 (1到15) 短时间拉低, 并在15微秒后读取,

从机如果发送0, 则在主机拉低后, 跟着拉低, 保持总线为0, 则15微秒后主机读取为0

从机如果发送1, 则主机拉低后, 不作反应, 则15微秒后主机读取为1

为什么发送和接受不会混淆?

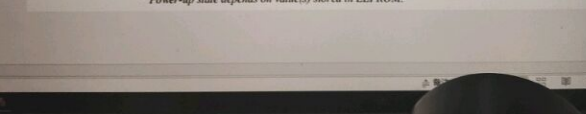
总线在一般情况下都由主机控制, 只有主机向从机发送需要读的指令后, 从机才能或者总线的控制权, 这样读和写就不会混淆

单总线时序结构

- 发送一个字节: 连续调用8次发送一位的时序, 依次发送一个字节的8位 (低位在前)

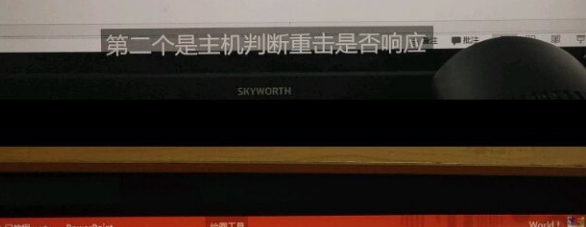


- 接收一个字节: 连续调用8次接收一位的时序, 依次接收一个字节的8位 (低位在前)



DS18B20介绍

- DS18B20是一种常见的数字温度传感器, 其控制命令和数据都是以数字信号的方式输入输出, 相比较于模拟温度传感器, 具有功能强大、硬件简单、易扩展、抗干扰性强等特点
- 测温范围: -55°C 到 +125°C
- 通信接口: 1-Wire (单总线)
- 其它特征: 可形成总线结构、内置温度报警功能、可寄生供电

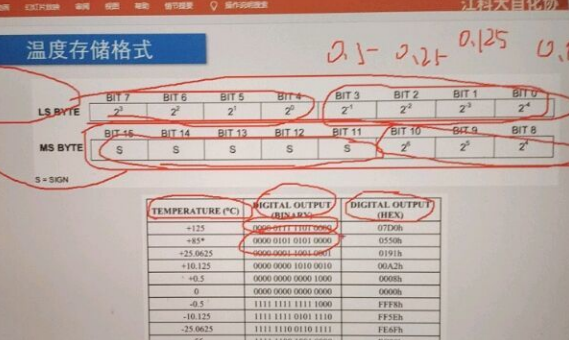


引脚及应用电路

引脚	功能
VDD	电源 (3.0V - 5.5V)
GND	电源地
DQ	单总线接口

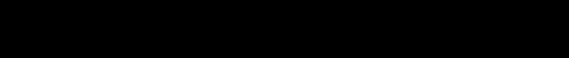


内部结构框图



- 64-BIT ROM: 作为器件地址, 用于总线通信的寻址
- SCRATCHPAD (暂存器): 用于总线的数据交互
- EEPROM: 用于保存温度触发阈值和配置参数

存储器结构



DS18B20操作流程

- 初始化: 从机复位, 主机判断从机是否响应
- ROM操作: ROM指令+本指令需要的读写操作
- 功能操作: 功能指令+本指令需要的读写操作

ROM指令	功能指令
SEARCH ROM [F0h]	CONVERT T [44h]
READ ROM [33h]	WRITE SCRATCHPAD [4Eh]
MATCH ROM [55h]	READ SCRATCHPAD [BEh]
SKIP ROM [CCh]	COPY SCRATCHPAD [48h]
ALARM SEARCH [ECh]	RECALL E2 [B8h]
	READ POWER SUPPLY [B4h]

第二个是主机判断是否响应

DS18B20数据帧

- 温度变换: 初始化→跳过ROM→开始温度变换

- 温度读取: 初始化→跳过ROM→读暂存器→连续的读操作

我们发生完这个字节之后

温度存储格式

S表示符号位

这16进制啊二进制16进制的一个