

概述

DS18B20 数字温度传感器提供 9-Bit 到 12-Bit 的摄氏温度测量精度和一个用户可编程的非易失性且具有过温和低温触发报警的报警功能。DS18B20 采用的 1-Wire 通信即仅采用一个数据线（以及地）与微控制器进行通信。该传感器的温度检测范围为-55℃至+125℃，并且在温度范围超过-10℃至 85℃之外时还具有 +0.5℃的精度。此外，DS18B20 可以直接由数据线供电而不需要外部电源供电。

每片 DS18B20 都有一个独一无二的 64 位序列号，所以一个 1-Wire 总线上可连接多个 DS18B20 设备。因此，在一个分布式的大环境里用一个微控制器控制多个 DS18B20 是非常简单的。这些特征使得其在 HVAC 环境控制，在建筑、设备及机械的温度监控系统，以及温度过程控制系统中有着很大的优势。

特性

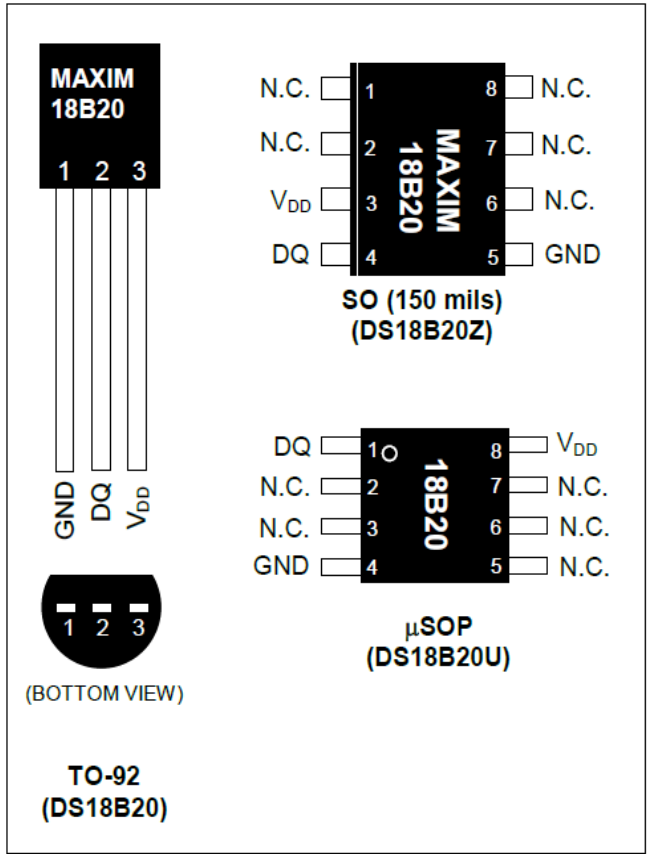
- 独特的 1-Wire 总线接口仅需要一个管脚来通信。
- 每个设备的内部 ROM 上都烧写了一个独一无二的 64 位序列号。
- 多路采集能力使得分布式温度采集应用更加简单。
- 无需外围元件。
- 能够采用数据线供电；供电范围为 3.0V 至 5.5V。
- 温度可测量范围为：-55℃至+125℃（-67°F至+257°F）。
- 温度范围超过-10℃至 85℃之外时具有 +0.5℃的精度。
- 内部温度采集精度可以由用户自定义为 9-Bits 至 12-Bits。

DS18B20 分辨率可编程

1-Wire 数字温度传感器

- 温度转换时间在转换精度为 12-Bits 时达到最大值 750ms。
- 用户自定义非易失性的温度报警设置。
- 定义了温度报警搜索命令和当温度超过用户自定义的设定值时。
- 可选择的 8-Pin SO (150 mils), 8-Pin μ SOP, 及 3-Pin TO-92 封装。
- 与 DS1822 程序兼容。
- 应用于温度控制系统，工业系统，民用产品，温度传感器，或者任何温度检测系统中。

管脚定义图



订购信息

零件	温度范围	引脚数-封装	顶部标号
DS18B20	-55℃至+125℃	3 TO-92	18B20
DS18B20+	-55℃至+125℃	3 TO-92	18B20
DS18B20/T&R	-55℃至+125℃	3 TO-92（2000 片）	18B20
DS18B20+T&	-55℃至+125℃	3 TO-92（2000 片）	18B20
DS18B20-SL/T	-55℃至+125℃	3 TO-92（2000 片）*	18B20
DS18B20-SL+	-55℃至+125℃	3 TO-92（2000 片）*	18B20
DS18B20U	-55℃至+125℃	8 uSOP	18B20
DS18B20U+	-55℃至+125℃	8 uSOP	18B20
DS18B20U/T&	-55℃至+125℃	8 uSOP（3000 片）	18B20
DS18B20+T&	-55℃至+125℃	8 uSOP（3000 片）	18B20
DS18B20Z	-55℃至+125℃	8 SO	DS18B20
DS18B20Z+	-55℃至+125℃	8 SO	DS18B20
DS18B20Z/T&	-55℃至+125℃	8 SO（2500 片）	DS18B20
DS18B20Z+T	-55℃至+125℃	8 SO（2500 片）	DS18B20

“+”号表示的是无铅封装。“+”出现在无铅封装的顶部标号处。

T&R=卷带包装。

*TO-92 封装

管脚描述

管脚			管脚名	功能描述
SO	uSOP	TO-92		
1、2、 6、7、 8	2、3、5、6、 7	—	N.C	置空
3	8	3	VDD	VDD 引脚。VDD 必须连接到地当采用“寄生电源”供电时。
4	1	2	DQ	数据输入/输出。1-Wire 漏极开路接口引脚。当采用“寄生电源”供电方式时，同时向设备提供电源。（详见“DS18B20 的供电”章节）
5	4	1	GND	地

综述

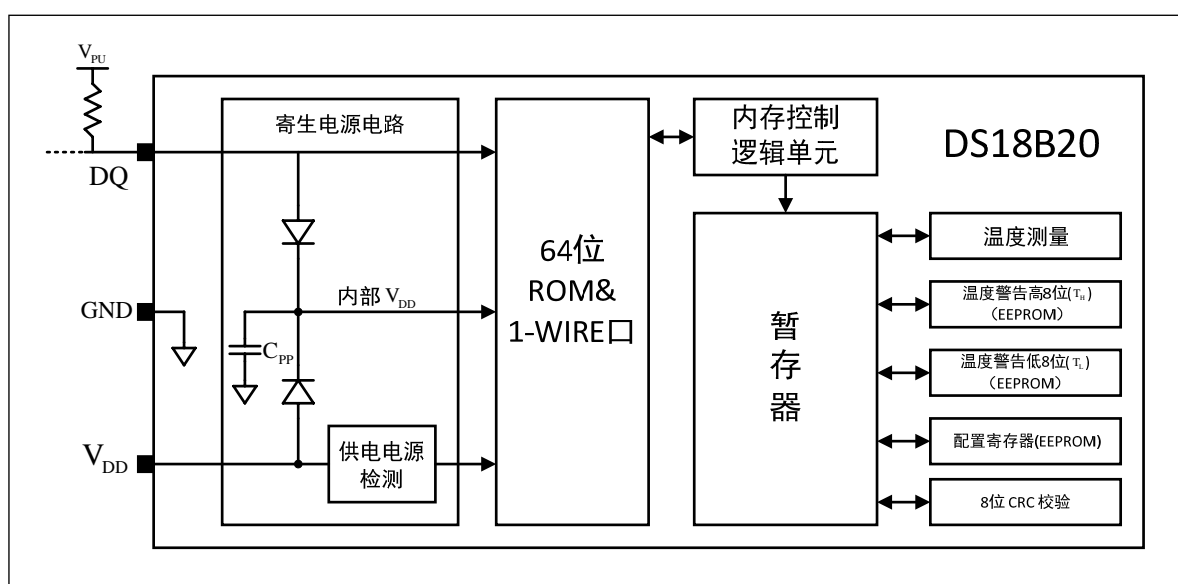
图 1 为 DS18B20 的内部框图。内部的 64 位的 ROM 存储其独一无二的序列号。暂存存储器（The scratchpad memory）包含了存储有数字温度结果的 2 个字节宽度的温度寄存器。另外，暂存存储器还提供了一个字节的过温和低温（TH 和 TL）温度报警寄存器和一个字节的配置寄存器。

配置寄存器允许用户自定义温度转换为 9、10、11、12 位精度。过温和低温（TH 和 TL）温度报警寄存器是非易失性的（EEPROM），所以其可以在设备断电的情况下保存。

DS18B20 采用的 Maxim 公司专有的 1-Wire 总线协议，该总线协议仅需要一个控制信号进行通信。该控制信号线需要一个唤醒的上拉电阻以防止连接在该总线上的口是 3 态或者高阻态（DQ 信号线是在 DS18B20 上）。在该总线系统中，微控制器（主设备）通过每个设备的 64 为序列号来识别该总线上的设备。因为每个设备都有一个独一无二的序列号，挂在一个总线上的设备理论上是可以无限个的。在下面的“1-Wire 总线系统”章节中包含有 1-Wire 总线协议详细的命令和时序关系。

DS18B20 的另外一个特性就是可以无需外部电源供电。当数据线 DQ 为高的时候由其为设备供电。总线拉高的时候为内部电容（C_{pp}）充电，当总线拉低时由该电容向设备供电。这种由 1-Wire 总线为设备供电的方式称为“寄生电源”。此外，DS18B20 也可以由外部电源通过 VDD 供电。

图 1 DS18B20 内部方框图



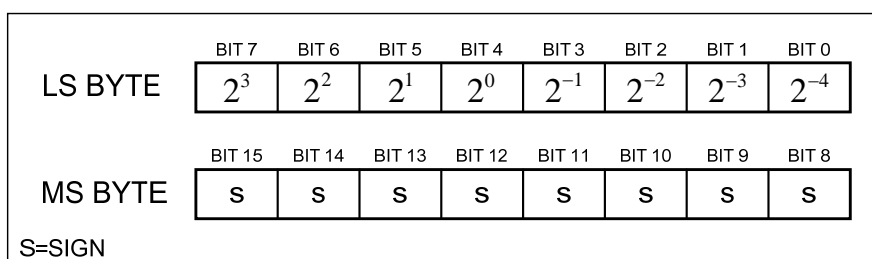
说明-温度测量

DS18B20 的核心功能是直接温度-数字测量。其温度转换可由用户自定义为 9、10、11、12 位精度分别为 0.5℃、0.25℃、0.125℃、0.0625℃分辨率。值得注意的是，上电默认为 12 位转换精度。DS18B20 上电后工作在低功耗闲置状态下。主设备必须向 DS18B20 发送温度转换命令[44h]才能开始温度转换。温度转换后，温度转换的值将会保存在暂存存储器的温度寄存器中，并且 DS18B20 将会恢复到闲置状态。如果 DS18B20 是由外部供电，当发送完温度转换命令[44h]后，主设备可以执行“读数据时序”（请参阅“1-Wire 总线系统”章节），若此时温度转换正在进行 DS18B20 将会响应“0”，若温度转换完成则会响应“1”。如果 DS18B20 是由“寄生电源”供电，该响应的技术将不能使用，因为在整个温度转换期间，总线必须强制拉高。该总线的“寄生电源”供电方式将会在“DS18B20 的供电”章节中详细讲解。

DS18B20 的温度输出数据时在摄氏度下校准的；若是在华氏度下应用的话，可以用查表法或者常规的数据换算。温度数据以一个 16 位标志扩展二进制补码数的形式存储在温度寄存器中（详

见图 2)。符号标志位 (S) 温度的正负极性：正数则 S=0，负数则 S=1。如果 DS18B20 被定义为 12 位的转换精度，温度寄存器中的所有位都将包含有效数据。若为 11 位转换精度，则 bit 0 为未定义的。若为 10 位转换精度，则 bit 1 和 bit 0 为未定义的。若为 9 位转换精度，则 bit 2、bit 1 和 bit 0 为未定义的。表格 1 为在 12 位转换精度下温度输出数据与相对应温度之间的关系表。

图 2 温度寄存器格式



表格 1 温度/数据对应关系

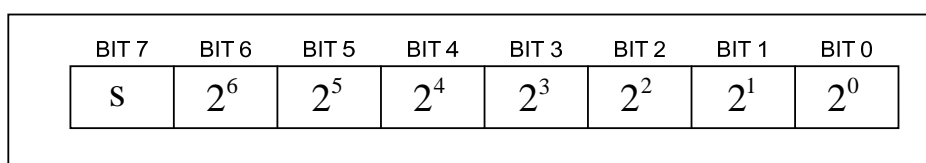
温度 (°C)	数据输出 (二进制)	数据输出 (十六进制)
+125	0000 0111 1101 0000	07D0h
+85*	0000 0101 0101 0000	0550h
+26.0625	0000 0001 1001 0001	0191h
+10.125	0000 0000 1010 0010	00A2h
+0.5	0000 0000 0000 1000	0008h
0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1111 1000	FFF8h
-10.125	1111 1111 0101 1110	FF5Eh
-25.0525	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

*上电复位时温度寄存器中的值为+85°C。

说明-温度报警

当 DS18B20 完成一次温度转换后，该温度转换值将会与用户定义的温度报警 TH 和 TL 寄存器（详见图 3）中的值进行比较。符号标志位 (S) 温度的正负极性：正数则 S=0，负数则 S=1。过温和低温 (TH 和 TL) 温度报警寄存器是非易失性的 (EEPROM)，所以其可以在设备断电的情况下保存。过温和低温 (TH 和 TL) 温度报警寄存器在“寄存器”章节中可以解释为暂存寄存器的第 2、3 个字节。

图 3 过温和低温 (TH 和 TL) 温度报警寄存器



因为过温和低温（TH 和 TL）温度报警寄存器是一个 8 位的寄存器，所以在于其比较时温度寄存器的 4 位至 11 位才是有效的数据。如果温度转换数据小于或等于 TL 及大于或等于 TH，DS18B20 内部的报警标志位将会被置位。该标志位在每次温度转换之后都会更新，因此，如报警控制消失，该标志位在温度转换之后将会关闭。

主设备可以通过报警查询命令[Che]查询该总线上的 DS18B20 设备的报警标志位。任何一个报警标志位已经置位的 DS18B20 设备都会响应该命令，因此，主设备可以确定到底哪个 DS18B20 设备存在温度报警。如果温度报警存在，并且过温和低温（TH 和 TL）温度报警寄存器已经被改变，则下一个温度转换值必须验证其温度报警标志位。

DS18B20 的供电

DS18B20 可以通过 DVD 引脚由外部供电，或者可以由“寄生电源”供电，这使得 DS18B20 可以不采用当地的外部电源供电而实现其功能。“寄生电源”供电方式在远程温度检测或空间比较有限的地方有很大的应用。图 1 展示的就是 DS18B20 的“寄生电源”控制电路，其由 DQ 口拉高时向其供电。总线拉高的时候为内部电容（C_{pp}）充电，当总线拉低是由该电容向设备供电。当 DS18B20 为“寄生电源”供电模式时，该 VDD 引脚必须连接到地。

在“寄生电源”供电模式下，只要工作在指定的时序下，则该 1-Wire 总线和 C_{pp} 可以提供给 DS18B20 足够的电流来完成各种工作以及满足供电电压（详见“交/直流电气特性”）。然而，当 DS18B20 正在进行温度转换或正将暂存寄存器中的值拷贝至 EEPROM 时，其工作电流将会高至 1.5mA。通过 1-Wire 总线上的上拉电阻提供的电流将会引起不可接受的电压跌落，同时将会有很大部分电流由 C_{pp} 提供。为了保证 DS18B20 有足够的电流供应，有必要在 1-Wire 总线上提供一个强有力的上拉，不管此时在进行温度转换还是正将暂存寄存器中的值拷贝至 EEPROM 中。图 4 中所示的由一个 MOSFET 直接将总线拉至高电平能够很好的实现。值得注意的是，1-Wire 总线必须在温度转换命令[44h]或暂存寄存器拷贝命令[48h]下达 10uS 后提供一个强有力的上拉，同时在整个温度转换期间（T_{conv}）或数据传送（T_{wr}=10ms）期间总线必须一直强制拉高。当强制拉高时该 1-Wire 总线上不允许有任何其他动作。

当然，DS18B20 也可以采用常规的通过外部电源连接至 VDD 引脚的供电方式，如图 5 所示。这种供电方式具有不需要上拉的 MOSFET、该 1-Wire 总线在温度转换期间可执行其他动作的优点。

“寄生电源”供电方式在温度超过+100℃时不推荐使用，因为在超过该温度下时将会有很大的漏电流导致不能进行正常的通信。实际应用中，在类似的温度状态下强烈推荐该 DS18B20 由外部供电电源供电。

在某些情况下，总线上的主设备可能不知道连接到该总线上的 DS18B20 是由“寄生电源”供电还是由外部电源供电。此时该主设备就需要得到一些信息来决定在温度转换期间是否要强制拉高。为了得到这些信息，主设备可以在发送一个跳过 ROM 命令[CCh]之后再发送一个读取供电方式命令[B4h]再紧跟一个“读取数据时序”。在读取数据时序中，“寄生电源”供电方式的 DS18B20 将会将总线拉低，但是，由外部供电方式的 DS18B20 将会让该总线继续保持高。所以，如果总线被拉低，主设备就必须要在温度转换期间将总线强制拉高。

图 4 “寄生电源” 供电方式

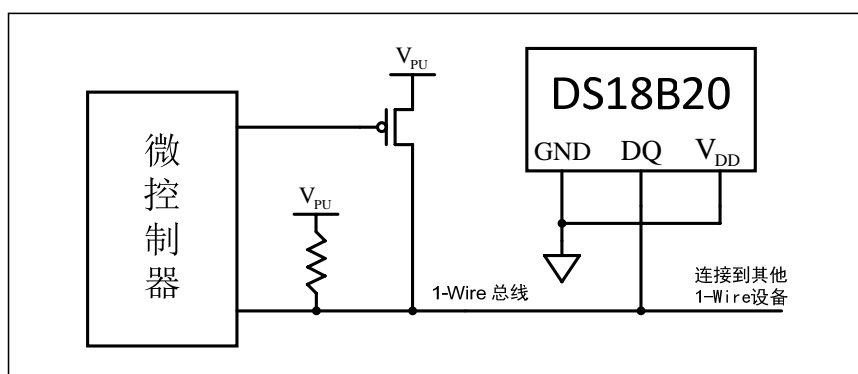
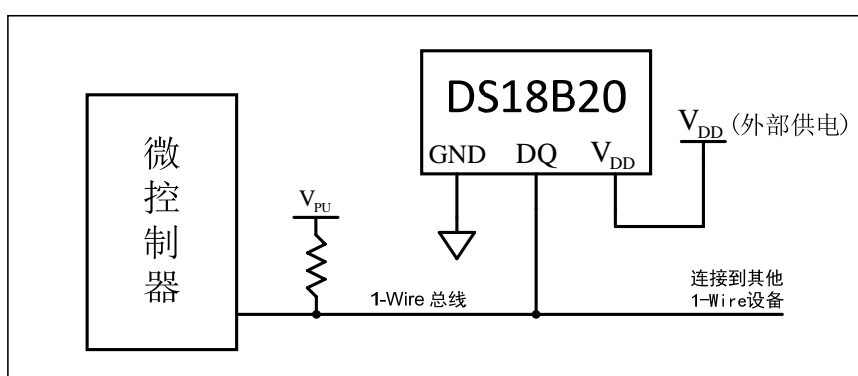


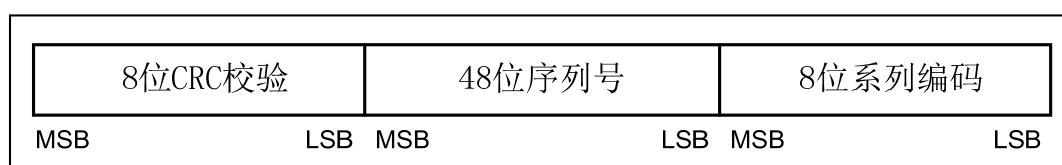
图 5 外部电源供电方式



64 位光刻 ROM 编码

每片 DS18B20 的片内 ROM 中都存有一个独一无二的 64 位的编码。在该 ROM 编码的低 8 位保存有 DS18B20 的分类编码：28h。中间的 48 位保存有独一无二的序列号。最高 8 位保存片内 ROM 中前 56 位的循环冗余校验（CRC）值。更加详细的在“1-Wire 总线系统”章节 该 64 位 ROM 编码及相关的 ROM 功能控制逻辑允许 DS18B20 作为 1-Wire 总线协议上的设备。

图 6 64 位光刻 ROM 编码



存储器

DS18B20 的存储器组织结构如图 7 所示。该存储器包含了 SRAM 暂存寄存器和存储着过温和低温（TH 和 TL）温度报警寄存器及配置寄存器的非易失性 EEPROM。值得注意的是当 DS18B20 的温度报警功能没有用到的时候，过温和低温（TH 和 TL）温度报警寄存器可以当做通用功能的存储单元。所有的存储命令在“DS18B20 功能命令”章节有详细描述。

暂存寄存器中的 Byte 0 和 Byte 1 分别作为温度寄存器的低字节和高字节。同时这两个字节是只读的。Byte 2 和 Byte 3 作为过温和低温（TH 和 TL）温度报警寄存器。Byte 4 保存着配置寄存

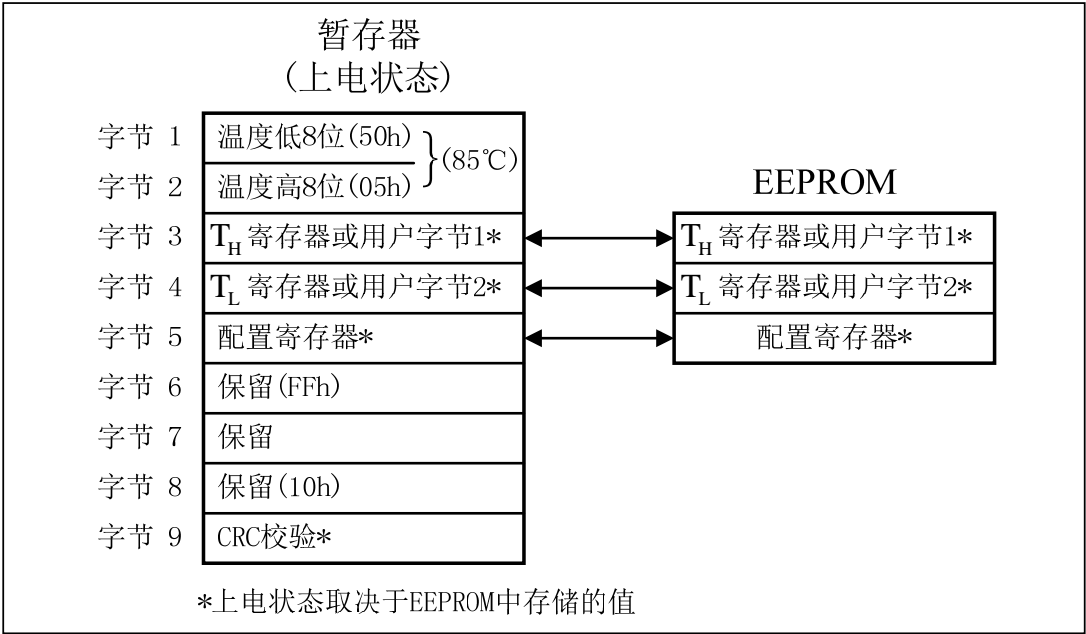
器的数据，详见“配置寄存器”章节。Byte 5、6、7 作为内部使用的字节而保留使用，不可被写入。

暂存寄存器的 Byte 8 为只读字节，其中存储着该暂存寄存器中 Byte 0 至 Byte 7 的循环冗余校验（CRC）值。DS18B20 计算该循环冗余校验（CRC）值的方法在“循环冗余校验（CRC）计算”章节中有详细描述。

使用写暂存寄存器命令[4Eh]才能将数据写入 Byte 2、3、4 中；这些写入 DS18B20 中的数据必须从 Byte 2 中最低位开始。为了验证写入数据的完整性，该暂存寄存器可以在写入后再读出来（采用读暂存寄存器命令[BEh]）。当从暂存寄存器中读数据时，从 1-Wire 总线传送的数据是以 Byte 0 的最低位开始的。为了将暂存寄存器中的过温和低温（TH 和 TL）温度报警值及配置寄存器数据转移至 EEPROM 中，主设备必须采用拷贝暂存寄存器命令[48h]。

在 EEPROM 寄存器中的数据在设备断电后是不会丢失的；在设备上电后 EEPROM 的值将会重新装载至相对应的暂存寄存器中。当然，在任何其他时刻 EEPROM 寄存器中的数据也可以通过重新装载 EEPROM 命令[B8h]将数据装载至暂存寄存器中。主设备可以在产生读时序后，紧跟着发送重新装载 EEPROM 命令，则如果 DS18B20 正在进行重新装载将会响应 0 电平，若重新装载已经完成则会响应 1 电平。

图 7 DS18B20 存储器图



配置寄存器

暂存寄存器中的 Byte 4 包含着配置寄存器；如图 8 所示。用户通过改变表 2 中 R0 和 R1 的值来配置 DS18B20 的分辨率。上电默认为 R0=1 及 R1=1（12 位分辨率）。需要注意的是，转换时间与分辨率之间是有制约关系的。Bit 7 和 Bit 0 至 Bit 4 作为内部使用而保留使用，不可被写入。

图 8 配置寄存器

BITS 7:0	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
	0	R1	R0	1	1	1	1	1

表 2 温度分辨率配置

R1	R0	转换位数	最大转换时间	
0	0	9	93.75ms	($t_{\text{CONV}}/8$)
0	1	10	187.5ms	($t_{\text{CONV}}/4$)
1	0	11	375ms	($t_{\text{CONV}}/2$)
1	1	12	750ms	(t_{CONV})

循环冗余校验（CRC）计算

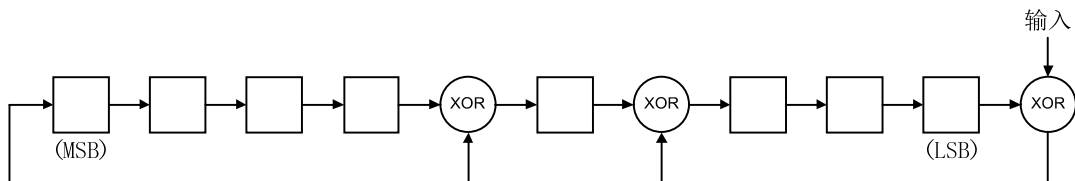
DS18B20 的 64 位 ROM 编码的一部分和暂存寄存器的第 9 字节都为循环冗余校验（CRC）计算字节。ROM 编码的循环冗余校验（CRC）值是由 ROM 编码的低 56 位计算而来，并且该 CRC 计算值存储在 ROM 编码的最高 8 位。暂存寄存器的 CRC 值是由存储在暂存寄存器中的值计算而来，故当暂存寄存器中的值发生改变后，该 CRC 值也会随之发生改变。当总线上的主设备从 DS18B20 中读取数据时循环冗余校验（CRC）值给主设备提供一个数据验证码。为了验证读取到的数据是正确的，主设备必须根据读取到的数据重新进行 CRC 计算，计算得到的值再与 ROM 编码 CRC（从 64 位 ROM 中读取到的）或者暂存寄存器 CRC（从暂存寄存器中读取到的）作比较。如果主设备计算得到的 CRC 值与读取到的 CRC 值相匹配，则读取到的数据为正确的。CRC 计算值与读取值的比较以及是否执行下一个动作都是由总线上的主设备决定的。如果主设备计算的 CRC 值与从 DS18B20 中（ROM 或暂存寄存器）读取的 CRC 值不匹配，DS18B20 内部没有任何电路能够阻止从主设备发送过来的命令。

CRC 校验（ROM 或暂存寄存器）的多项式等效公式为：

$$CRC = X^8 + X^5 + X^4 + 1$$

主设备可以根据图 9 中的多项式重新计算 CRC 值与从 DS18B20 中读取得到的值进行比较。该电路包括有左移寄存器和异或门（XOR），并且左移寄存器初始状态为 0。从 ROM 编码的最低位或暂存寄存器的 Byte 0 字节的最低位开始，每一步都必须有一位左移进入左移寄存器中。当 ROM 编码中的第 56 位或暂存寄存器中 Byte 7 字节的最高位左移后，该多项式计算式将会保存 CRC 校验值。下一步，将从 DS18B20 中的 ROM 编码中或暂存寄存器中读取到的 CRC 校验值左移进入该计算式。这些都完成后，如果重新计算的 CRC 值为正确的，则该计算式中的所有左移寄存器都会是 0。关于 Maxim 公司的 1-Wire 总线 CRC 校验的更多信息请参阅：Application Note 27: Understanding and Using Cyclic Redundancy Checks with Maxim iButton Products.

CRC 计算式



1-Wire 总线系统

1-Wire 总线系统即一个总线主设备控制一个或多个从设备。**DS18B20** 始终是一个从设备。当总线上只有一个从设备时，此系统被称为“单节点”系统；当总线上有多个从设备连接时，此系统被称之为“多节点”系统。

1-Wire 总线上所有的命令或者数据的发送都是遵循低位先发送的原则。

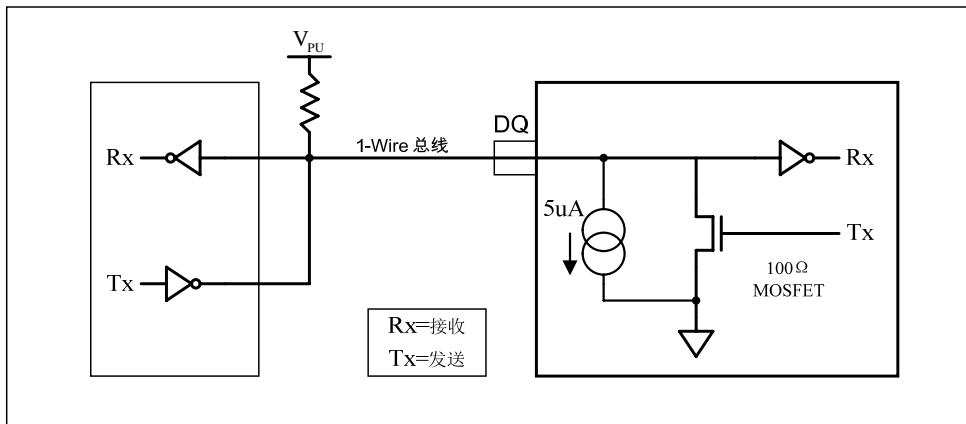
接下来关于 1-Wire 总线系统的描述将会分成三个部分：硬件配置，事件序列和 1-Wire 总线信号（信号定义和时序）。

硬件配置

1-Wire 总线被定义为仅有一根数据线。每个设备（主设备或从设备）通过一个漏极开路或 3 态门引脚连接至数据线上。这就允许每个设备“释放”数据线，当设备没有传递数据的时其他设备可以有效地使用数据线。DS18B20 的 1-Wire 总线接口（DQ 引脚）是其内部电路组成的漏极开路（如图 10 所示）。

1-Wire 总线需要一个 $5k\ \Omega$ 左右的外部上拉电阻；因此，1-Wire 总线在闲置情况下是高电平。如果因为任何原因一个事件需要被取消，且该事件要重新开始则该总线必须先进入闲置状态。Infinite recovery time can occur between bits so long as the 1-Wire bus is in the inactive (high) state during the recovery period 如果总线被拉低超过 $480\mu S$ ，则该总线上的所有设备都会复位。

图 10 硬件配置



事件序列

访问 DS18B20 的事件序列如下所示:

第一步：初始化

第二步：ROM 命令（紧跟任何数据交换请求）

第三步：DS18B20 功能命令（紧跟任何数据交换请求）

每次对 DS18B20 的访问都必须遵循这样的步骤来进行，如果这些步骤中的任何一个丢失或者没有执行，则 DS18B20 将不会响应。除了 ROM 搜索命令[F0h]和报警搜索命令[ECh]之外。当执行完这些 ROM 命令之后，主设备必须回到上述步骤中的第一步。

初始化

1-Wire 总线上的所有事件都必须以初始化为开始。初始化序列由总线上的主设备发出的复位脉冲以及紧跟着从设备回应的存在脉冲构成。该回应脉冲让总线上的主设备知道在该总线上有从设备（例如 DS18B20），并且已经准备好进行操作。复位及存在脉冲时序详见“1-Wire 信号”章节。

ROM 命令

当总线上的主设备检测到了存在脉冲后，就可以执行 ROM 命令。这些命令是对每个设备独一无二的 64 位 ROM 编码进行操作的，当总线上连接有多个设备时，可以通过这些命令识别各个设备。这些命令同时也可以使主设备确定该总线上有多少个什么类型的设备或者有温度报警信号的设备。总共包含有 5 种 ROM 命令，每个命令的长度都是 8 Bit。主设备在执行 DS18B20 功能命令之前必须先执行一个适当的 ROM 命令。ROM 命令的执行流程图如图 11 所示。

搜索 ROM[F0h]

当系统上电初始化后，主设备必须识别该总线上所有的从设备的 ROM 编码，这样就可以使得主设备确定总线上的从设备的类型及数量。主设备学习 ROM 编码是一个清除的过程，则主设备要根据需要循环地发送搜索 ROM[F0h]命令（搜索 ROM 命令跟随着数据交换）来确定总线上所有的从设备。如果仅有一个从设备在该总线上，更加简单的读取 ROM 命令（下一段落有详解）可以代替搜索 ROM 的过程。

读取 ROM[33h]

该命令在总线上仅有一个从设备时才能使用。该命令使得总线上的主设备不需要搜索 ROM 命令过程就可以读取从设备的 64 位 ROM 编码。当总线上有超过一个从设备时，若再发送该命令，则当所有从设备都会回应时，将会引起数据冲突。

匹配 ROM[55h]

该匹配 ROM 命令之后跟随发送 64 位的 ROM 编码使得总线上的主设备能够匹配特定的从设备。只有完全匹配该 64 位 ROM 编码的从设备才会响应总线上的主设备发出的功能命令；总线上的其他从设备将会等待下下一个复位脉冲。

跳过 ROM[CCh]

主设备可以使用该命令来同时向总线上的所有从设备发送不要发送任何的 ROM 编码命令。例如，主设备通过向总线上所有的 DS18B20 发送跳过 ROM 命令后再发送温度转换[44h]命令，则所有设备将会同时执行温度转。

需要注意的是，当总线上仅有一个从设备时，读取暂存寄存器[BEh]命令后面可以跟随跳过 ROM 命令。在这种情况下，主设备可以读取从设备中的数据而不发送 64 位 ROM 编码。当总线上有多个从设备时，若在跳过 ROM 命令后再发送读取暂存寄存器命令，则所有的从设备将会同时开始传送数据而导致总线上的数据冲突。

警报搜索[ECh]

该命令的操作与跳过 ROM 命令基本相同，但是不同的是只有警报标志置位的从设备才会响应。该命令使得主设备确定在最近一次温度转换期间是否有 DS18B20 有温度报警。当所有的报警搜索命令循环执行后，总线上的主设备必须回到事件序列中的第一步（初始化）。详见“操作报警信号”章节。

DS18B20 功能命令

当总线上的主设备通过 ROM 命令确定了哪个 DS18B20 能够进行通信时，主设备可以向其中一个 DS18B20 发送功能命令。这些命令使得主设备可以向 DS18B20 的暂存寄存器写入或者读出数据，初始化温度转换及定义供电模式。DS18B20 的功能命令在下面详细描述，在表 3 中总结及图 12 中的流程图。

温度转换[44h]

该命令为初始化单次温度转换。温度转换完后，温度转换的数据存储在暂存寄存器的 2 个字节长度的温度寄存器中，之后 DS18B20 恢复到低功耗的闲置状态。如果该设备是采用的“寄生电源”供电模式，在该命令执行 10 μ s（最大）后主设备在温度转换期间必须强制拉高数据线（“DS18B20 的供电”章节所描述）。如果该设备是采用的外部供电模式，主设备在温度转换命令之后可以执行读取数据时序，若 DS18B20 正在进行温度转换则会响应 0 电平，温度转换完成则响应 1 电平。在“寄生电源”供电模式下，因为在整个温度转换期间总线都是强制拉高的状态，故不会有上述响应。

写入暂存寄存器[4Eh]

该命令使得主设备向 DS18B20 的暂存寄存器写入 3 个字节的数据。第一个字节的数据写入 TH 寄存器（暂存寄存器的 Byte 2），第二个字节的数据写入 TL 寄存器（Byte 3），第三个字节的数据写入配置寄存器（Byte 4）。所有的数据必须是以低位先发的原则。所有的三个字节的数据在写入之前主设备必须先对从设备复位，否则数据将会损坏。

读取暂存寄存器[BEh]

该命令使得主设备可以读取暂存寄存器中存储的值。数据从 Byte 0 的低位开始传送直到第 9 个字节（Byte 8 - CRC）读取完毕。主设备若只需要暂存寄存器中的部分数据，则可以在读取数据中通过复位来终止。

拷贝暂存寄存器[48h]

该命令为将暂存寄存器中的 TH、TL 及配置寄存器（Byte 2，Byte 3 和 Byte 4）的值拷贝至 EEPROM 中。如果该设备采用的“寄生电源”供电模式，在该命令发送后 10us（最大）内主设备必须强制拉高 1-Wire 总线超过 10ms。如“DS18B20 的供电”章节中详述。

召回 EEPROM[B8h]

该命令将温度报警触发值（TH 和 TL）及配置寄存器的数据从 EEPROM 中召回至暂存寄存器中的 Byte 2，Byte 3 和 Byte4 中。主设备可以在召回 EEPROM 命令之后执行读取数据时序，若 DS18B20 正在进行召回 EEPROM 则会响应 0 电平，召回 EEPROM 完成则响应 1 电平。召回数据操作在上电初始化后会自动执行一次，所以设备在上电期间暂存寄存器中一直会有有效的数据。

读取供电模式[B4h]

主设备通过执行该命令之后再执行读取数据时序来确定总线上的 DS18B20 是否是由“寄生电源”供电。在读取数据时序中，“寄生电源”供电的 DS18B20 将会拉低总线，外部电源独立供电模式的 DS18B20 则会释放总线让其保持在高电平。更多详细请参阅“DS18B20 的供电”章节

表 3 DS18B20 的功能命令

命令	描述	规定数据	命令执行后总线下一步活动	注释
温度转换命令				
转换温度	初始化温度转换	44h	DS18B20 发送转换标志到控制器(对寄生电源供电的 DS18B20 不适用)	1
存储器命令				
读取暂存寄存器	读取暂存寄存器中全部的数据, 包括 CRC 字节。	BEh	DS18B20 发送 9 字节数据到控制器	2
写入暂存寄存器	向暂存寄存器中的字节 2、3 和 4 (TH, TL 及配置寄存器) 写入数据	4Eh	控制器发送 3 字节数据到 DS18B20	3
拷贝暂存寄存器	将暂存寄存器中的 TH, TL 及配置寄存器中的数据拷贝到 EEPROM 中	48h	无	1
召回 EEPROM	将 EEPROM 中的 TH, TL 及配置寄存器数据召回到暂存寄存器中	B8h	DS18B20 发送召回标志到控制器	
读取供电模式	读取 DS18B20 的供电模式到控制器中	B4h	DS18B20 发送供电标志到控制器	

注释 1: 对于寄生电源供电的 DS18B20, 在温度转换及将数据从暂存寄存器中拷贝到 EEPROM 期间控制器必须将 1-Wire 总线强制拉高, 在这段时间内没有其他总线活动。

注释 2: 控制器可以在任何时候通过执行复位来中断数据的传送

注释 3: 所有 3 个字节数据必须在复位执行之前写入

图 11 ROM 命令流程图

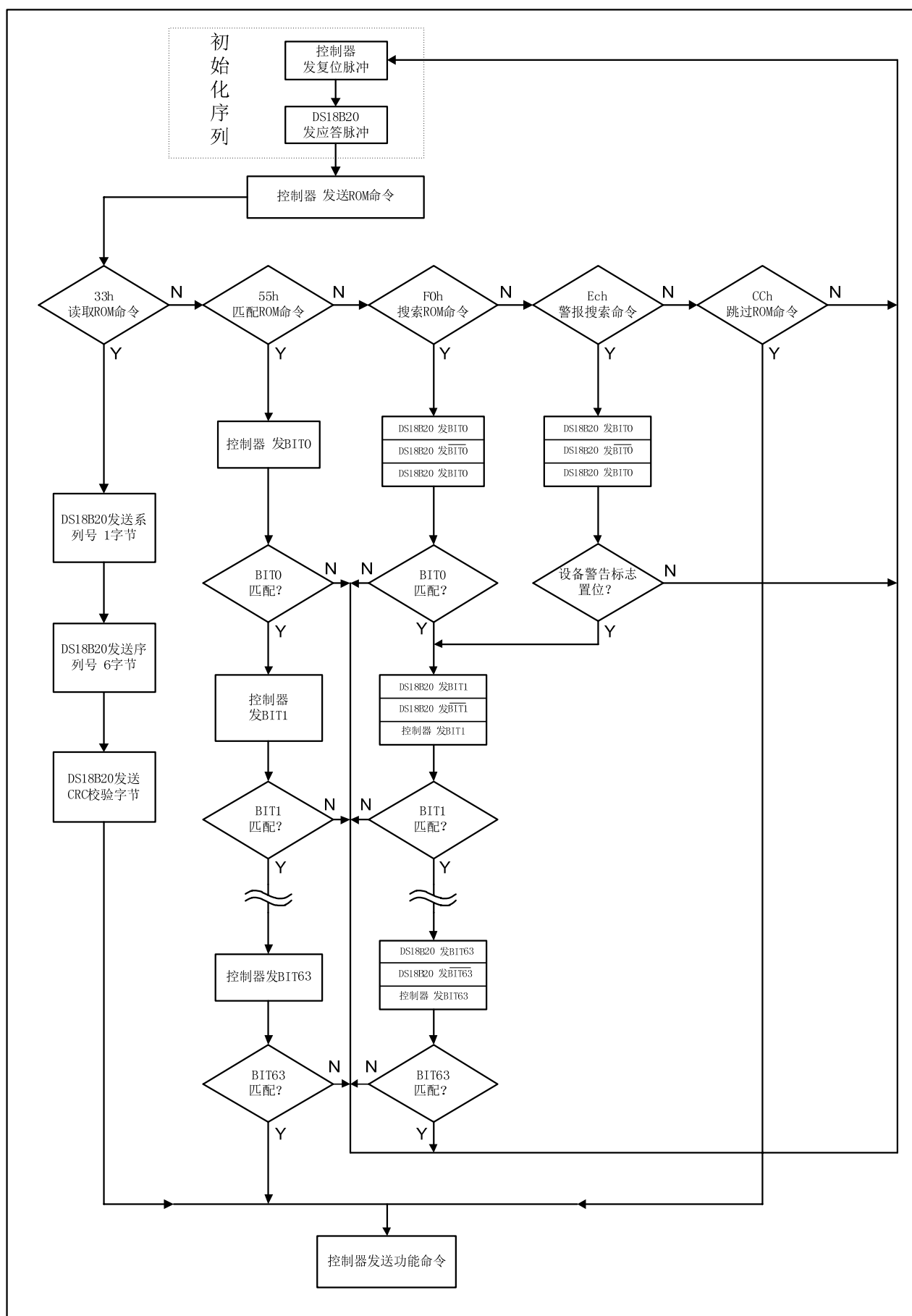
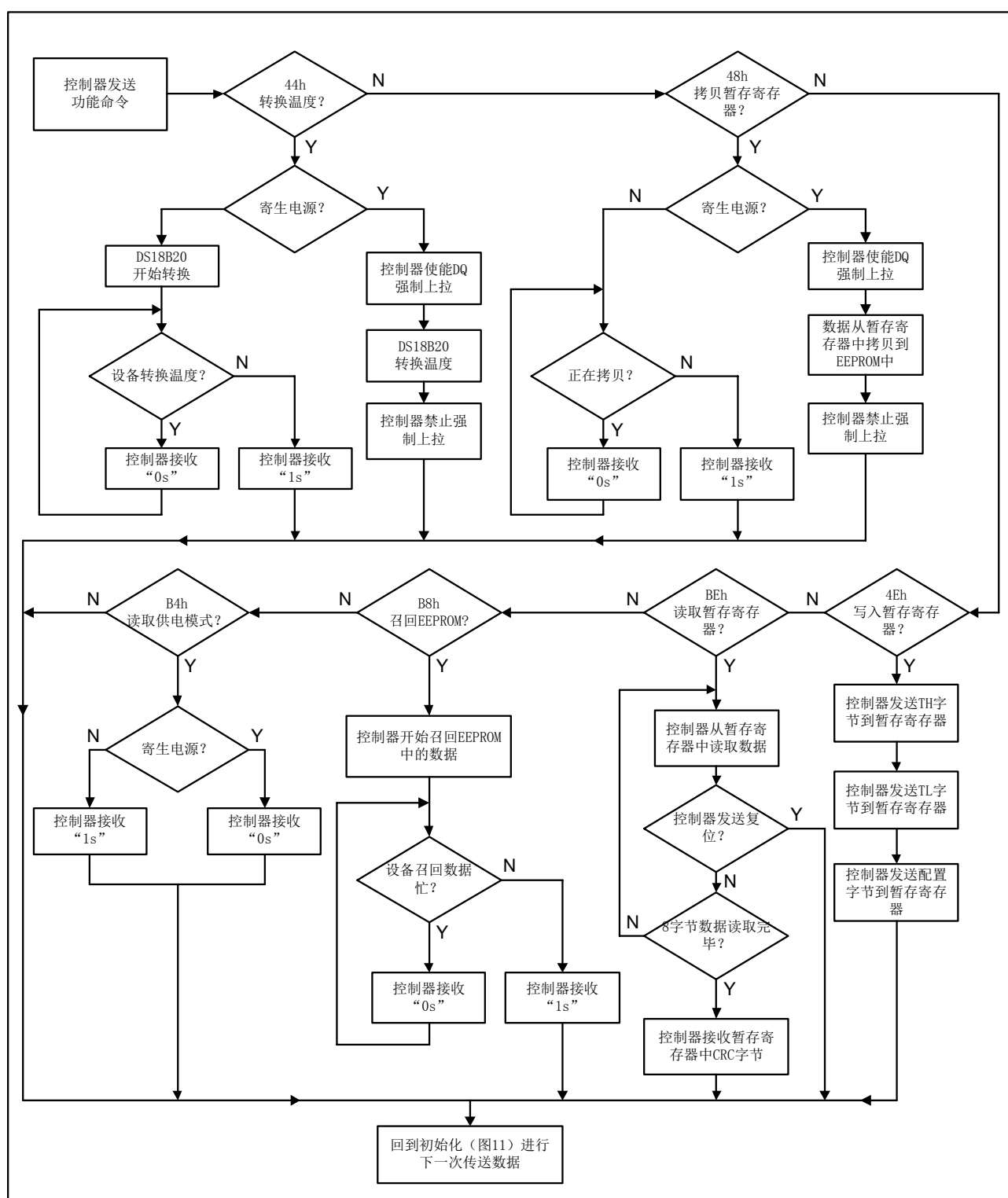


图 12 DS18B20 功能命令流程图



1-Wire 总线信号

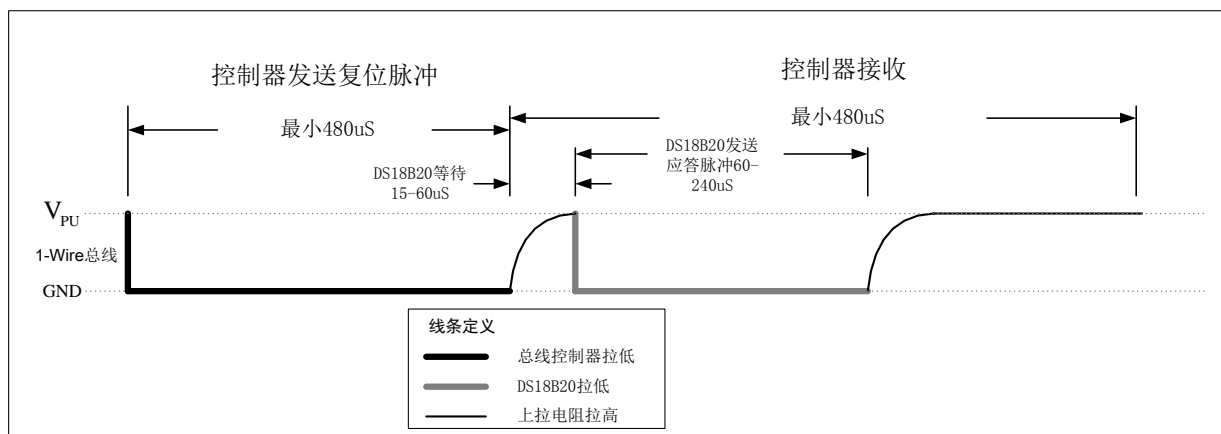
DS18B20 采用严谨的 1-Wire 总线通信协议来保证数据的完整性。该协议定义多个信号形式：复位脉冲，存在脉冲，写 0，写 1，读 0，读 1。主设备执行除了存在脉冲外的所有其他信号。

初始化程序—复位和存在脉冲

与 DS18B20 所有的通信都是由初始化序列开始的，该序列包括从主设备发出的复位脉冲及从 DS18B20 响应的存在脉冲组成。如图 13 所示。当 DS18B20 响应复位信号的存在脉冲后，则其向主设备表明其在该总线上，并且已经做好操作命令。

在初始化序列期间，总线上的主设备通过拉低 1-Wire 总线超过 480us 来发送 (TX) 复位脉冲。之后主设备释放总线而进入接收模式 (RX)。当总线释放后，5k Ω 左右的上拉电阻将 1-Wire 总线拉至高电平。当 DS18B20 检测到该上升边沿信号后，其等待 15us 至 60us 后通过将 1-Wire 总线拉低 60us 至 240us 来实现发送一个存在脉冲。

图 13 初始化时序



读/写时段

主设备通过写时段向 DS18B20 中写入数据，通过读时段从 DS18B20 中读取数据。1-Wire 总线上每一个读写时段只能传送一个位的数据。

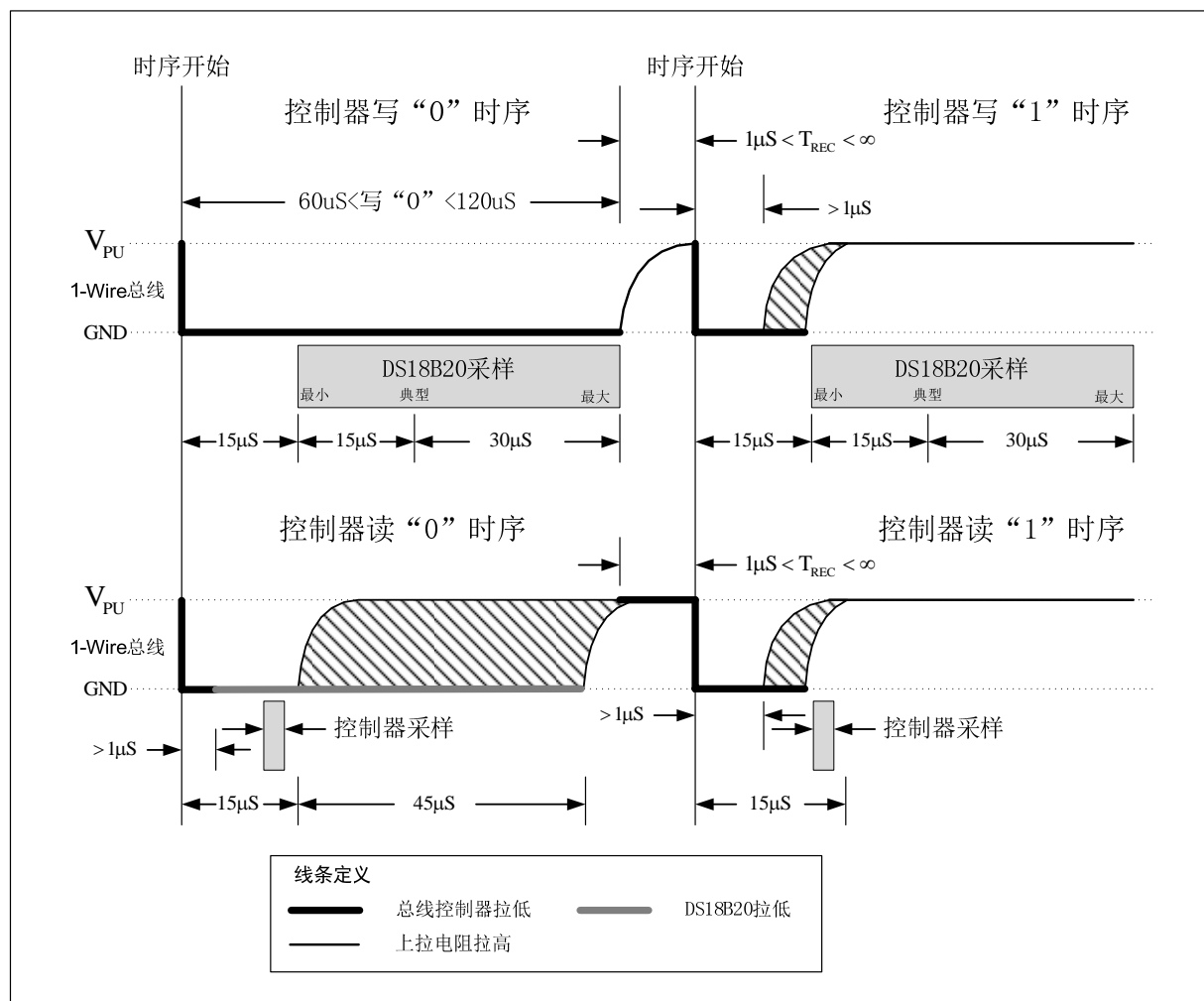
写时段

写时段有两种情况：“写 1”时段和“写 0”时段。主设备通过写 1 时段来向 DS18B20 中写入逻辑 1 以及通过写 0 时段来向 DS18B20 中写入逻辑 0。每个写时段最小必须有 60us 的持续时间且独立的写时段间至少有 1us 的恢复时间。两个写时段都是由主设备通过将 1-Wire 总线拉低来进行初始化（详见图 14）。

为了形成写 1 时段，在将 1-Wire 总线拉低后，主设备必须在 15us 之内释放总线。当总线释放后，5k Ω 的上拉电阻将总线拉至高。为了形成写 0 时段，在将 1-Wire 总线拉低后，在整个时段期间主设备必须一直拉低总线（至少 60us）。

在主设备初始化写时段后，DS18B20 将会在 15us 至 60us 的时间窗口内对总线进行采样。如果总线在采样窗口期间是高电平，则逻辑 1 被写入 DS18B20；若总线是低电平，则逻辑 0 被写入 DS18B20。

图 14 读/写时段时序图



读时段

仅在读时段期间 DS18B20 才能向主设备传送数据。因此，主设备在执行完读暂存寄存器[B5h]或读取供电模式[B4h]后，必须及时地生成读时段，这样 DS18B20 才能提供所需的数据。此外，主设备可以在执行完转换温度[44h]或拷贝 EEPROM[B8h]命令后生成读时段，以便获得在“DS18B20 功能命令”章节中提到的操作信息。

每个读时段最小必须有 60us 的持续时间且独立的写时段间至少有 1us 的恢复时间。读时段通过主设备将总线拉低超过 1us 再释放总线来实现初始化（见图 14）。当主设备初始化完读时段后，DS18B20 将会向总线发送 0 或者 1。DS18B20 通过将总线拉至高来发送逻辑 1，将总线拉至低来发送逻辑 0。当发送完 0 后，DS18B20 将会释放总线，则通过上拉电阻该总线将会恢复到高电平的闲置状态。从 DS18B20 中输出的数据在初始化读时序后仅有 15us 的有效时间。因此，主设备在开始改读时段后的 15us 之内必须释放总线，并且对总线进行采样。

图 15 读时段中的 T_{int} ， T_{rc} 及 T_{sample} 必须小于 15us。

图 16 显示系统的时序正确保证即是保持 T_{init} 和 T_{rc} 尽可能的短，同时主设备必须在读时段开始 $15\mu\text{s}$ 时间内采样。

图 15 详细的控制器读 1 时序

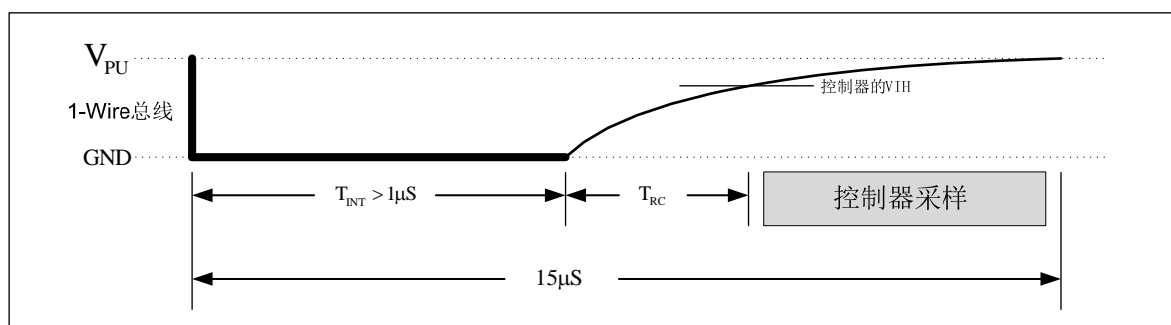
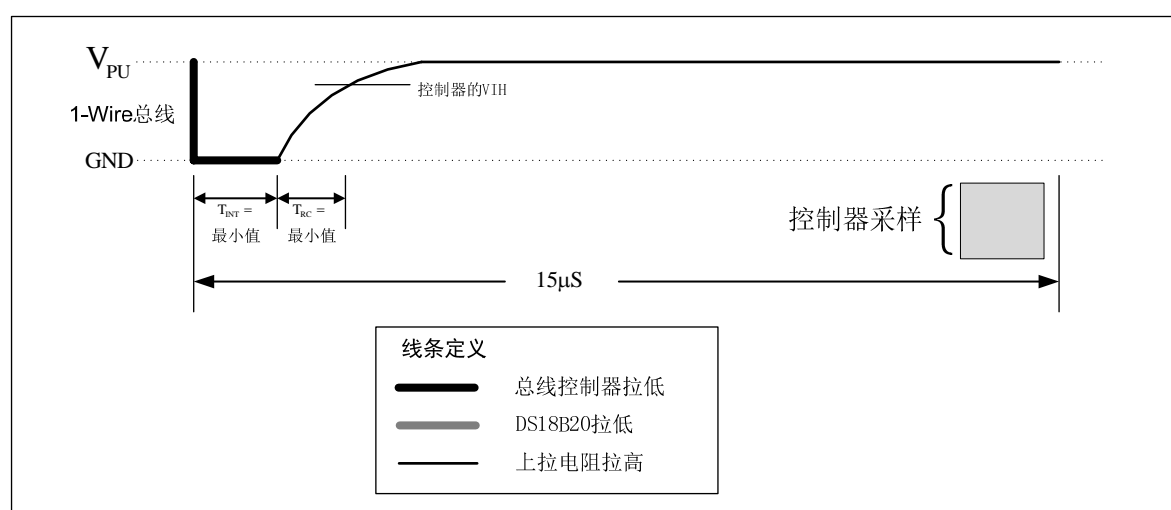


图 16 建议的控制器读 1 时序



相关应用笔记

The following application notes can be applied to the DS18B20 and are available on our website at www.maxim-ic.com.

Application Note 27: Understanding and Using Cyclic Redundancy Checks with Maxim iButton Products

Application Note 122: Using Dallas' 1-Wire ICs in 1-Cell Li-Ion Battery Packs with Low-Side N-Channel Safety FETs Master

Application Note 126: 1-Wire Communication Through Software

Application Note 162: Interfacing the DS18x20/DS1822 1-Wire Temperature Sensor in a Microcontroller Environment

Application Note 208: Curve Fitting the Error of a Bandgap-Based Digital Temperature Sensor

Application Note 2420: 1-Wire Communication with a Microchip PICmicro Microcontroller

Application Note 3754: Single-Wire Serial Bus Carries Isolated Power and Data

Sample 1-Wire subroutines that can be used in conjunction with *Application Note 74: Reading and Writing iButtons via Serial Interfaces* can be downloaded from the Maxim website.

DS18B20 操作实例 1

在该例中，有多个 DS18B20 挂在该总线上，同时它们都采用的“寄生电源”供电。主设备初始化一个 DS18B20 的温度转换后再读取它的暂存寄存器中的数据之后进行 CRC 校验来验证数据。

控制器模式	数据（低位先发）	注释
发送	复位	控制器执行复位脉冲
接收	存在脉冲	DS18B20 响应存在脉冲
发送	55h	控制器执行匹配 ROM 命令
发送	64 位 ROM 编码	控制器发送 DS18B20 ROM 编码
发送	44h	控制器执行转换温度命令
发送	DQ 被强制上拉	控制器在温度转换时间内强制上来 DQ
发送	复位	控制器执行复位脉冲
接收	存在脉冲	DS18B20 响应存在脉冲
发送	55h	控制器执行匹配 ROM 命令
发送	64 位 ROM 编码	控制器发送 DS18B20 ROM 编码
发送	BEh	控制器执行读取暂存寄存器命令
接收	9 字节数据	控制器读取暂存寄存器中的 CRC 检验值。控制器将由暂存寄存器中读取的前八位数据的 CRC 值与读取出来的 CRC 值比较。如果该两个值相等，则控制器继续，若不符，则再次读取暂存寄存器。

DS18B20 操作实例 2

在该实例中，仅有一个 DS18B20 连接到总线上，并且其采用的“寄生电源”供电。主设备向 DS18B20 中的暂存寄存器的 TH，TL 和配置寄存器中写入数据，之后读出暂存寄存器中的数据，并对数据进行 CRC 校验。最后主设备将暂存寄存器中的值拷贝至 EEPROM 中。

控制器模式	数据（低位先发）	注释
发送	复位	控制器执行复位脉冲
接收	存在脉冲	DS18B20 响应存在脉冲
发送	CCh	控制器执行跳过 ROM 命令
发送	4Eh	控制器执行写取暂存寄存器命令
发送	3 字节数据	控制器发送 3 个字节数据到暂存寄存器（TH，TL，配置寄存器）。
发送	复位	控制器执行复位脉冲
接收	存在脉冲	DS18B20 响应存在脉冲
发送	CCh	控制器执行跳过 ROM 命令
发送	BEh	控制器执行读取暂存寄存器命令
接收	9 字节数据	控制器读取暂存寄存器中的 CRC 检验值。控制器将由暂存寄存器中读取的前八位数据的 CRC 值与读取出来的 CRC 值比较。如果该两个值相等，则控制器继续，若不符，则再次读取暂存寄存器。
发送	复位	控制器执行复位脉冲
接收	存在脉冲	DS18B20 响应存在脉冲
发送	CCh	控制器执行跳过 ROM 命令
发送	48h	控制器执行拷贝暂存寄存器命令
发送	DQ 被强制上拉	在拷贝命令执行期间控制器强制拉高 DQ 至少 10ms。

后记

本 DS18B20 中文数据手册为陈兵先生于 2013 年 7 月翻译于实验楼 226。

七月初时，我在全老师实验室的新生培训中负责讲解 DS18B20 的编程，虽然刚开始学习单片机的时候有用过 DS18B20，但是真正系统的去讲解这个器件还是第一次。为了能够系统并且专业地讲解该芯片的编程方法，我开始在网络上寻找一些文档来参考，再找来找去都没有找到较好的中文文档后，我去官方网站只找到了一个英文的文档；分析了一下英文原版文档，并且与网上流传的各种版本的中文说明对比后，发现错误很多，很不专业。故此，萌发了自己动手翻译该篇文档的想法。一则能够锻炼自己的英文水平，二则能够在自己学生生涯里为我大中华的工程师贡献自己的一份力量。

本文档翻译仓促，有不当的地方还请见谅，仅做参考。版权归 MAXIM 公司所有。