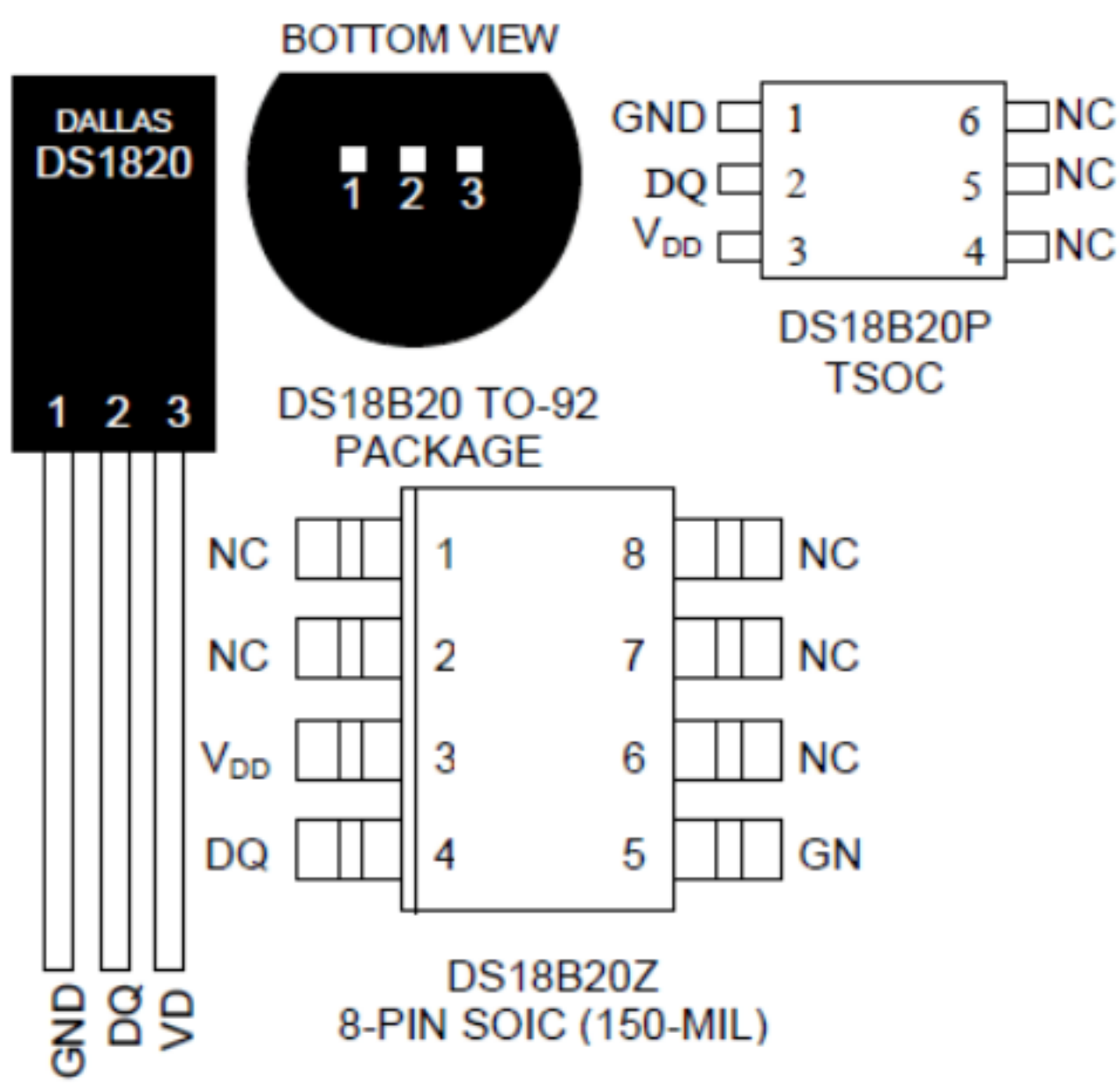
DS18B20

可编程分辨率的单总线　．数字温度计

特征

独特的单线接口仅需一个端口引脚进行通讯
每个器件有唯一的 64 位的序列号存储在内部存储器中
简单的多点分布式测温应用
无需外部器件
可通过数据线供电。供电范围为　3.0V 到5.5V
测温范围为 -55～＋125℃（－67～＋257℉）
在－10～＋85℃范围内精确度为＋　5℃
温度计分辨率可以被使用者选择为　9～12位
最多在 750ms　内将温度转换为 12　位数字
用户可定义的非易失性温度报警设置
报警搜索命令识别并标志超过程序限定温度（温度报警条件）的器件
与DS1822 兼容的软件
应用包括温度控制、工业系统、消费品、温度计或任何热感测系统

引脚排列



说明

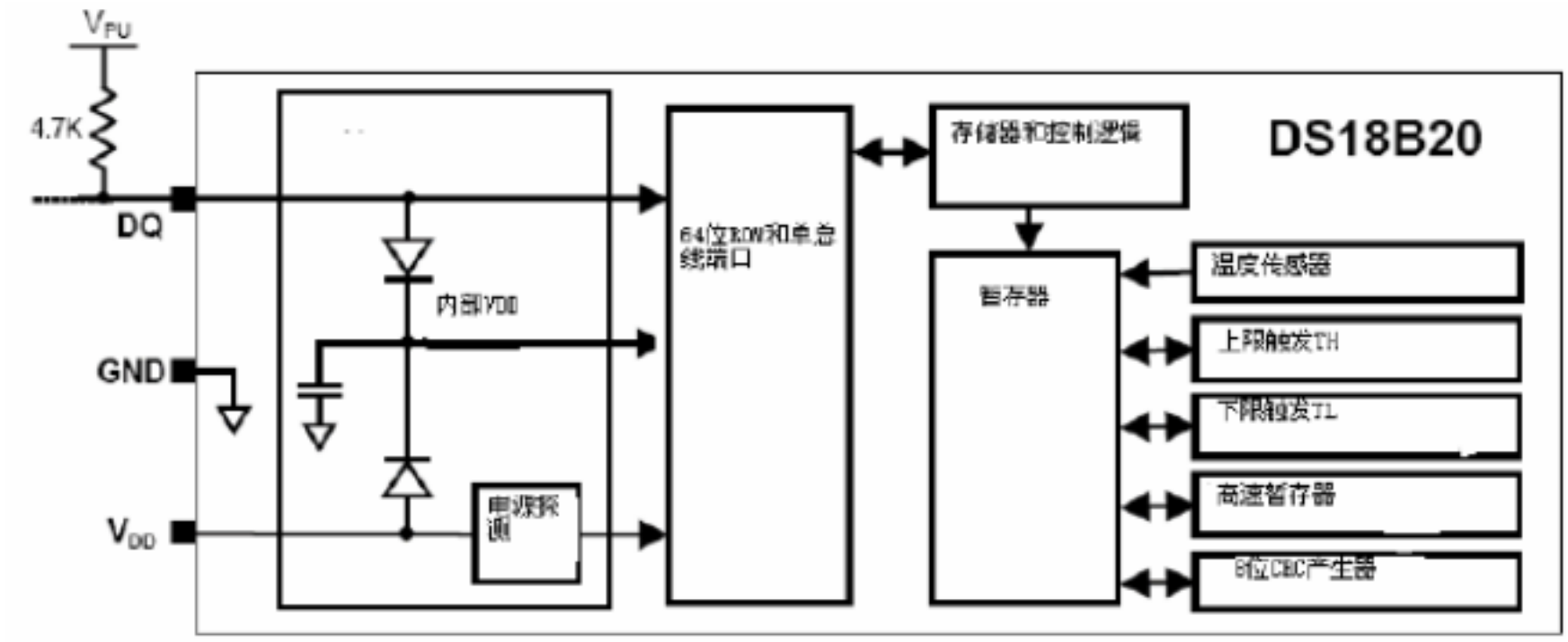
DS18B20 数字温度计提供 9至12位的摄氏温度测量， 并具有易失性的用户可编程触发点的上限和下限报警功能。　DS18B20 单总线通信按定义只需要一条数据线

（和地线）与中央微处理器进行通信。它具有工作温度范围 55 ℃ + 125 ℃和准确 ±0.5 ℃范围 - 10 ℃ + 85 ℃. 此外， DS18B20 可以导出功率直接从数据线（ “寄生虫 ”），消除了外部电源需要。
每个 DS18B20 都有一个唯一的 64位串行码，它允许多个 DS18B20 在同一根总线功能。因此，它是用一个简单的微处理器控制，大面积分布的许多 DS18B20s 。应用程序可以受益于这个功能包括空调环境控制， 建筑物内的温度监控系统， 设备，或机械，和过程监控控制系统。
详细的引脚说明 表1

8 引脚SOIC 封装*	TO-9 封装	符号	说明
5	1	GND	接地。
4	2	DQ	数据输入 /输出引脚。 对于单线操作： 漏极开路。当工作在寄生电源模式时 用来提供电源（建“ 寄生电源 ”节）。
3	3	VDD	可选的 VDD 引脚。 工作与寄生电源模 式时 VDD 必须接地。

概览

图 1 是表示 DS18B20 的方框图， 表1 已经给出了引脚说明。 64 位只读存储器储存器件的唯一片序列号。 高速暂存器含有两个字节的温度寄存器， 这两个寄存器用来存储温度传感器输出的数据。 除此之外， 高速暂存器提供一个直接的温度报警值寄存器（ TH和 TL），和一个字节的的配置寄存器。配置寄存器允许用户将温度的精度设定为 9， 10， 11 或12 位。 TH,TL 和配置寄存器是非易失性的可擦除程序寄存器（ EEPROM ），所以存储的数据在器件掉电时不会消失。
DS18B20 通过达拉斯公司独有的单总线协议依靠一个单线端口通讯。当全部器件经由一个 3态端口或者漏极开路端口（ DQ引脚在 DS18B20 上的情况下)与总线连接的时候，控制线需要连接一个弱上拉电阻。 在这个总线系统中， 微控制器（主器件）依靠每个器件独有的 64 位片序列号辨认总线上的器件和记录总线上的器件地址。由于每个装置有一个独特的片序列码 ,总线可以连接的器件数目事实上是无限的。单总线协议，包括指令的详细解释和 “ 时序 ” 见 单总线系统 节。
DS18B20 的另一个功能是可以在没有外部电源供电的情况下工作。当总线处于高电平状态， DQ与上拉电阻连接通过单总线对器件供电。同时处于高电平状态的总线信号对内部电容（ Cpp）充电，在总线处于低电平状态时，该电容提供能量给器件。 这种提供能量的形式被称为 “ 寄生电源 ”。作为替代选择， DS18B20 同样可以通过 VDD引脚连接外部电源供电。
DS18B20 方框图 图1



测温操作

DS18B20 的核心功能是它的直接读数字的温度传感器。温度传感器的精度为用户可编程的 9，10，11或12 位，分别以 0.5℃，0.25℃，0.125℃和0.0625 ℃增量递增。在上电状态下默认的精度为 12位。DS18B20 启动后保持低功耗等待状态；当需要执行温度测量和 AD转换时，总线控制器必须发出 [44h] 命令。在那之后，产生的温度数据以两个字节的形式被存储到高速暂存器的温度寄存器中，

DS18B20 继续保持等待状态。当 DS18B20 由外部电源供电时，总线控制器在温度转换指令之后发起“读时序”（ 见单总线系统节 ），DS18B20 正在温度转换中返回 0,转换结束返回 1。如果 DS18B20 由寄生电源供电，除非在进入温度转换时总线被一个强上拉拉高，否则将不会由返回值。寄生电源的总线要求在 DS18B20 供电 节详细解释。

温度寄存器格式 图 2

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	2^6	2^5	2^4

温度/数据关系 表 2

温度 ℃	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
+25.0625	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
-25.0625	1111 1110 0110 1111	FE6Eh
-55	1111 1100 1001 0000	FC90h

*上电复位时温度寄存器默认值为+85℃

报警操作信号

DS18B20 完成一次温度转换后， 就拿温度值与和存储在 TH和TL中一个字节的用户自定义的报警预置值进行比较。标志位（ S）指出温度值的正负：正数 S=0，负数 S=1。TH和TL寄存器是非易失性的，所以它们在掉电时仍然保存数据。在存储器 节将解释 TH和TL是怎么存入高速暂存器的第 2和第3个字节的。

TH 和TL 寄存器格式 图3

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
S	2^6	2^5	2^5	2^5	2^2	2^1	2^0

当TH和TL为8位寄存器时， 4位温度寄存器中的 11个位用来和 TH、TL进行比较。如果测得的温度高于 TH 或低于TL，报警条件成立， DS18B20 内部就会置位一个报警标识。 每进行一次测温就对这个标识进行一次更新； 因此，如果报警条件不成立了，在下一次温度转换后报警标识将被移去。

总线控制器通过发出报警搜索命令 [ECh] 检测总线上所有的 DS18B20 报警标识。任何置位报警标识的 DS18B20 将响应这条命令，所以总线控制器能精确定位每一个满足报警条件的 DS18B20 。如果报警条件成立，而 TH或TL的设置已经改变，另一个温度转换将重新确认报警条件。

DS18B20 供电

DS18B20 可以通过从 VDD引脚接入一个外部电源供电，或者可以工作于寄生电源模式，该模式允许 DS18B20 工作于无外部电源需求状态。寄生电源在进行远距离测温时是非常有用的。寄生电源的控制回路见图 1，当总线为高电平时，寄生电源由单总线通过 VDD引脚。这个电路会在总线处于高电平时偷能量，部分汲取的能量存储在寄生电源储能电容（ Cpp）内，在总线处于低电平时释放能量以提供给器件能量。当 DS18B20 处于寄生电源模式时， VDD引脚必须接地。

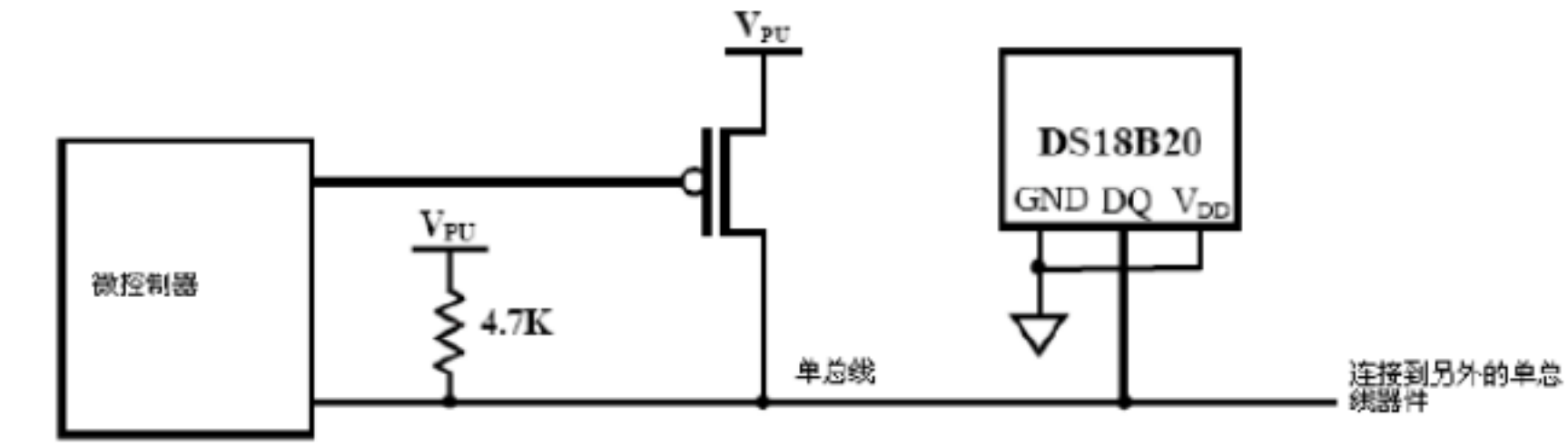
寄生电源模式下， 单总线和 Cpp 在大部分操作中能提供充分的满足规定时序和电压的电流（见 直流电特性 和交流电特性 节）给DS18B20 。然而，当 DS18B20 正在执行温度转换或从高速暂存器向 EPPROM 传送数据时，工作电流可能高达 1.5mA 。这个电流可能会引起连接单总线的弱上拉电阻的不可接受的压降， 这需要更大的电流，而此时 Cpp 无法提供。为了保证 DS18B20 由充足的供电，当进行温度转换或拷贝数据到 EEPROM 操作时，必须给单总线提供一个强上拉。用漏极开路把 I/O直接拉到电源上就可以实现，见图 4。在发出温度转换指令 [44h] 或拷贝暂存器指令 [48h] 之后，必须在至多 10us之内把单总线转换到强上拉，并且在温度转换时序 (tconv) 或拷贝数据时序 (ter=10 ms) 必须一直保持为强上拉状态。当强上拉状态保持时，不允许有其它的动作。

对DS18B20 供电的另一种传统办法是从 VDD引脚接入一个外部电源， 见图5。这样做的好处是单总线上不需要强上拉。 而且总线不用在温度转换期间总保持高电平。

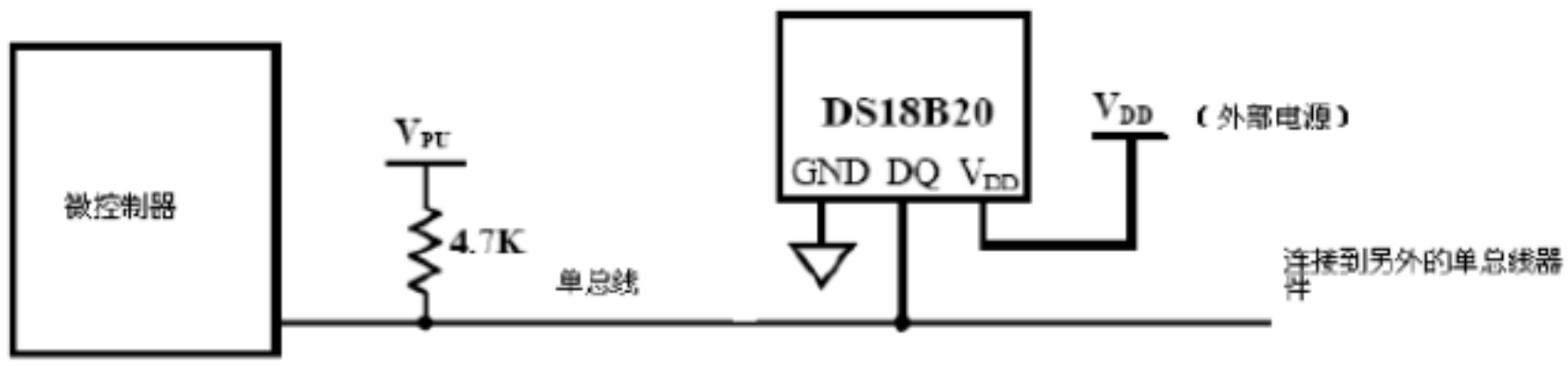
温度高于 100℃时，不推荐使用寄生电源， 因为 DS18B20 在这种温度下表现出的漏电流比较大，通讯可能无法进行。在类似这种温度的情况下，强烈推荐使用 DS18B20 的VDD引脚。

对于总线控制器不直到总线上的 DS18B20 是用寄生电源还是用外部电源的情况， DS18B20 预备了一种信号指示电源的使用意图。总线控制器发出一个 Skip ROM指令 [CCh] ，然后发出读电源指令 [B4h] ，这条指令发出后，控制器发出读时序，寄生电源会将总线拉低， 而外部电源会将总线保持为高。 如果总线被拉低，

总线控制器就会知道需要在温度转换期间对单总线提供强上拉。
DS18B20温度转换期间的强上拉供电　图4



外部电源给 DS18B20　供电 图5



64 位（激）光刻只读存储器

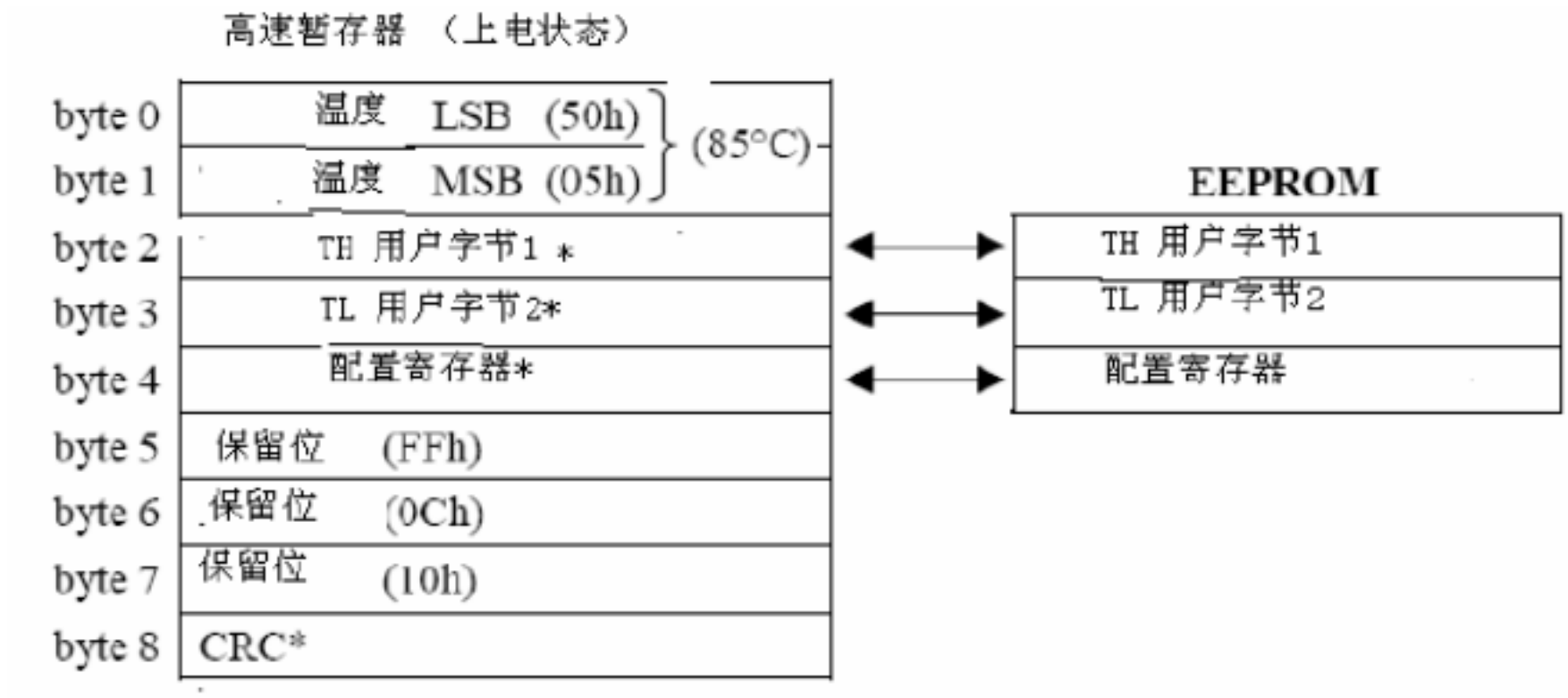
每只DS18B20　都有一个唯一存储在　ROM　中的64　位编码。最前面　8　位是单线系列编码：　28h。接着的 48　位是一个唯一的序列号。最后　8　位是以上 56　位的 CRC　编码。 CRC 的详细解释见 CRC　发生器 节。 64位ROM 和ROM 操作控制区允许 DS18B20 作为单总线器件并按照详述于　单总线系统 节的单总线协议工作。
64　位（激）光刻只读存储器　图6

8 位 CRC	48 位序列号	8 位系列码

存储器

DS18B20 的存储器结构示于图　7。存储器有一个暂存　SRAM 和一个存储高低报警触发值 TH　和TL　的非易失性电可擦除　EEPROM　组成。注意当报警功能不使用时，　TH和TL　寄存器可以被当作普通寄存器使用。所有的存储器指令被详述于 DS18B20 功能指令 节。
位0和位 1为测得温度信息的　LSB 和MSB。这两个字节是只读的。第　2和第 3字节是 TH和TL 的拷贝。位 4　包含配置寄存器数据，其被详述于　配置寄存器 节。位 5，6　和7被器件保留，禁止写入；这些数据在读回时全部表现为逻辑　　1。
高速暂存器的位　8是只读的，包含以上八个字节的　CRC码，　CRC的执行方式如 CRC发生器节 所述。
数据通过写暂存器指令　[4Eh] 写入高速暂存器的　2，　3和4位；数据必须以位　2为最低有效位开始传送。　为了完整的验证数据，　高速暂存器能够在数据写入后被读取（使用读暂存器指令　[BEh]）。在读暂存器时，数据以位　0为最低有效位从单总线移出。总线控制器传递从暂存器到　EEPROMTH,TL　和配置数据必须发出拷贝暂存器指令 [48h]。
EEPROM　寄存器中的数据在器件掉电时仍然保存；　上电时，数据被载入暂存器。

数据也可以通过召回 EEPROM 命令从暂存器载入 EEPROM 。总线控制器在发出这条
命令后发出读时序， DS18B20 返回0表示正在召回中，返回 1表示操作结束。
DS18B20 存储器图 图7



*上电状态依赖于 EEPROM 中的值

配置寄存器

存储器的第 4位为配置寄存器，其组织见图 8。用户可以通过按表 3所示设置 R0和R1位来设定 DS18B20 的精度。上电默认设置： R0=1,R1=1 （12位精度）。注意：
精度和转换时间之间有直接的关系。暂存器的位 7和位0-4被器件保留，禁止写入；在读回数据时，它们全部表现为逻辑 1。
配置寄存器 图8

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	R1	R0	1	1	1	1	1

温度计精确度配置 表3

R1	R0	精度	最大转换时间	
0	0	9-bit	93.75 ms	(t_{CONV}/8)
0	1	10-bit	187.5 ms	(t_{CONV}/4)
1	0	11-bit	375 ms	(t_{CONV}/2)
1	1	12-bit	750 ms	(t_{CONV})

CRC 发生器

CRC字节作为 DS18B2064 位ROM的一部分存储在存储器中。 CRC码由ROM的前56位计算得到，被包含在 ROM的重要字节当中。 CRC由存储在存储器中的数据计算得到，因此当存储器中的数据发生改变时， CRC的值也随之改变。

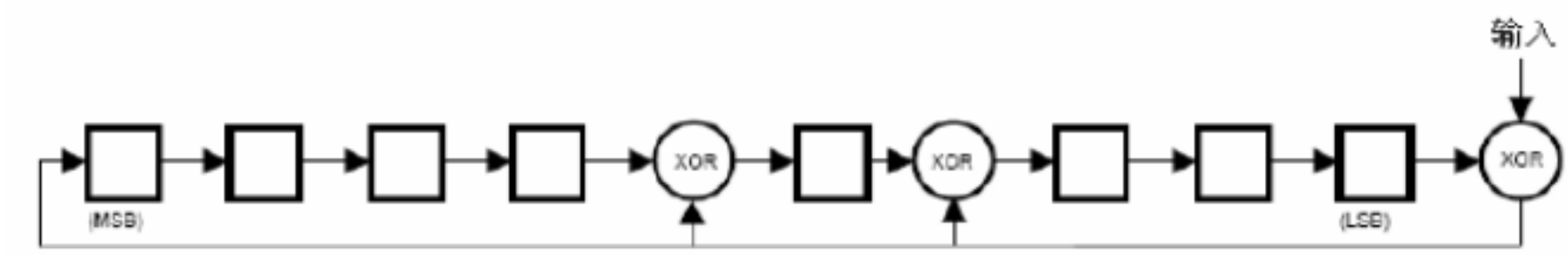
CRC能够在总线控制器读取 DS18B20 时进行数据校验。为校验数据是否被正确读取，总线控制器必须用接受到的数据计算出一个 CRC 值，和存储在 DS18B20 的64 位ROM 中的值（读ROM 时）或DS18B20 内部计算出的 8 位CRC 值（读存储器时）进行比较。如果计算得到的 CRC值和读取出来的 CRC值相吻合，数据被无错传输。 CRC 值的比较以及是否进行下一步操作完全由总线控制器决定。当在 DS18B20 中存储的或由其计算到 CRC 值和总线控制器计算的值不相符时， DS18B20 内部并没有一个能阻止命令序列进行的电路。

CRC的计算等式如下：

CRC = X8 + X5 + X4 + 1

单总线 CRC 可以由一个由移位寄存器和 XOR 门构成的多项式发生器来产生，见图9。这个回路包括一个移位寄存器和几个 XOR 门，移位寄存器的各位都被初始化为 0。从ROM 中的最低有效位或暂存器中的位 0开始，一次一位移入寄存器。在传输了 56位ROM 中的数据或移入了暂存器的位 7后，移位寄存器中就存储了 CRC 值。下一步， CRC 的值必须被循环移入。此时，如果计算得到的 CRC是正确的，移位寄存器将复 0。

CRC 发生器 图9



单总线系统

单总线系统包括一个总线控制器和一个或多个从机。 DS18B20 总是充当从机。当只有一只从机挂在总线上时，系统被称为“ 单点 ”系统；如果由多只从机挂在总线上，系统被称为“ 多点 ”。

所有的数据和指令的传递都是从最低有效位开始通过单总线。

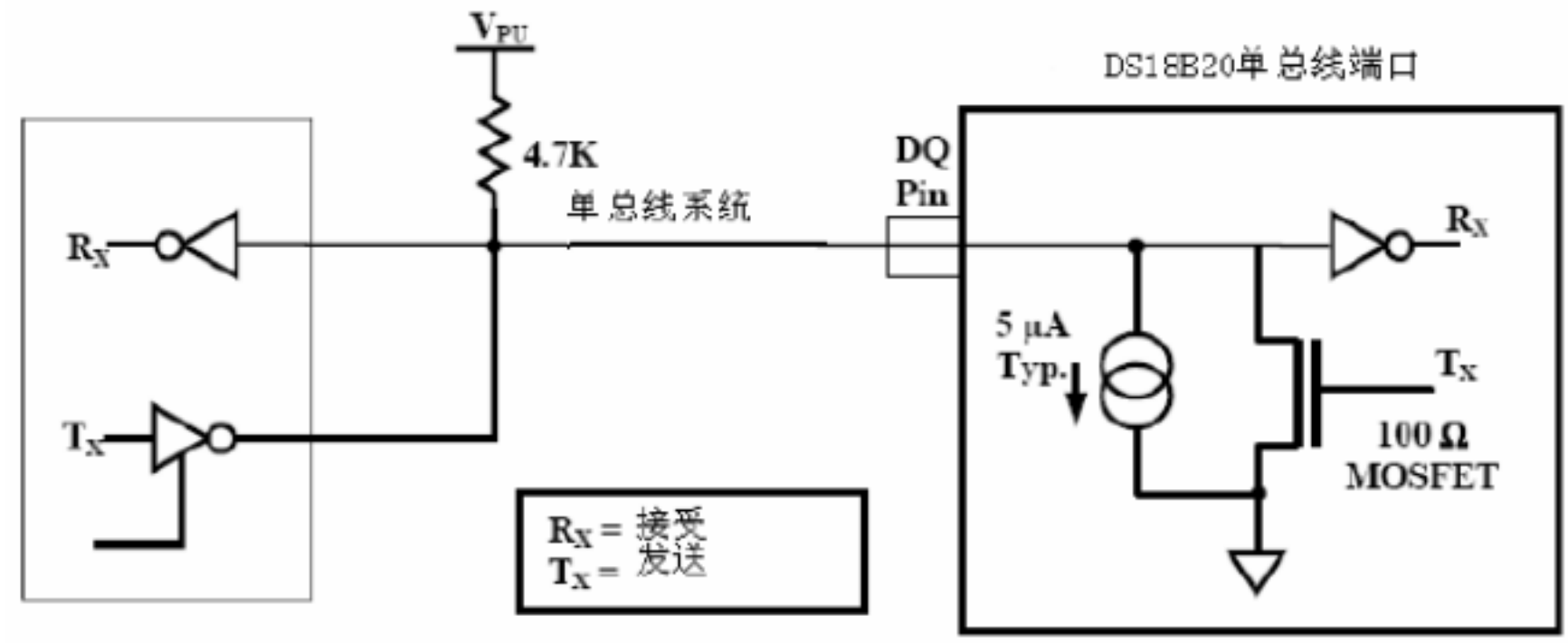
关于单总线系统分三个题目讨论： 硬件结构、 执行序列和单总线信号 （信号类型和时序）。

硬件结构

单总线系统只有一条定义的信号线。 每一个总线上的器件必须是漏极开路或三态输出。这样的系统允许每一个挂在总线上的区间都能在适当的时间驱动它。

DS18B20 的单总线端口（ DQ引脚）是漏极开路式的，内部等效电路见图 10 单总线需要一个约 5KΩ 的外部上拉电阻；单总线的空闲状态是高电平。无论任何理由需要暂停某一执行过程时， 如果还想恢复执行的画， 总线必须 停留在空闲状态。在恢复期间，如果单总线处于非活动（高电平）状态，位与位间的恢复时间可以无限长。如果总线停留在低电平超过 480us ， 总线上的所有器件都将被复位。

硬件结构 图10



执行序列

通过单线总线端口访问 DS18B20 的协议如下：

步骤 1. 初始化

步骤 2. ROM 操作指令

步骤 3. DS18B20 功能指令

每一次 DS18B20 的操作都必须满足以上步骤，若是缺少步骤或是顺序混乱，器件将不会返回值。例如这样的顺序：发起 ROM 搜索指令 [F0h] 和报警搜索指令 [ECh] 之后，总线控制器必须返回步骤 1。

初始化

通过单总线的所有执行操作处理都从一个初始化序列开始。 初始化序列包括一个由总线控制器发出的复位脉冲和其后由从机发出的存在脉冲。 存在脉冲让总线控制器知道 DS18B20 在总线上且已准备好操作，详见 单总线信号 节。

ROM 指令

一旦总线控制器探测到一个存在脉冲，它就发出一条 ROM 指令。如果总线上挂有多只 DS18B20，这些指令将基于器件独有的 64 位ROM 片序列码使得总线控制器选出特定要进行操作的器件。 这些指令同样也可以使总线控制器识别有多少只，什么型号的器件挂在总线上， 同样，它们也可以识别哪些器件已经符合报警条件。 ROM 指令有 5条，都是 8 位长度。总线控制器在发起一条 DS18B20 功能指令之前必须先发出一条 ROM 指令。 ROM 指令操作流程图见图 11。

Search ROM [F0h] (搜索 ROM 指令)

当系统上电初始化的时候，总线控制器必须通过识别总线上所有 ROM 片序列码去得到从机的数目和型号。总线控制器通过搜索 ROM 指令多次循环搜索 ROM 编码，以确认所有从机器件。 如果总线上只有一只从机， 那么可以用较为简单的读取ROM 指令（见下文）代替搜索 ROM 指令，关于 iButton Book of Standards 见 www.ibutton.com/ibuttons/standard.pdf。在每次搜索 ROM 指令之后，总线控制器必须返回步骤 1。

READ ROM [33h] （读取 ROM 指令）

只有在总线上存在单只 DS18B20 的时候才能使用这条命令。该命令允许总线控制器在不使用搜索 ROM 指令的情况下读取从机的 64 位片序列码。如果总线上有不止一只从机，当所有从机试图同时传送信号时就会发生数据冲突。

MATH ROM [55h] （匹配 ROM 指令）

匹配 ROM 指令，后跟 64 位 ROM 编码序列，让总线控制器在多点总线上定位

一只特定的 DS18B20 。只有和 64 位ROM 片序列码完全匹配的 DS18B20 才能响应随后的存储器操作指令；所有和 64 位ROM 片序列码不匹配的从机都将等待复位脉冲。

SKIP ROM [CCh] (忽略 ROM 指令)

这条指令允许总线控制器不用提供 64 位ROM 编码就使用功能指令。例如，总线控制器可以先发出一条忽略 ROM 指令，然后发出温度转换指令 [44h]，从而完成温度转换操作。注意：当只有一只从机在总线上时，无论如何，忽略 ROM 指令之后只能跟着发出一条读取暂存器指令 [BEh]。在单点总线情况下使用该命令，器件无需发回 64 位ROM 编码，从而节省了时间。 如果总线上有不止一只从机，若发出忽略 ROM 指令，由于多只从机同时传送信号，总线上就会发生数据冲突。

ALARM SEARCH [ECH] （报警搜索指令）

这条命令的流程和搜索 ROM 指令相同，然而，只有满足报警条件的从机才对该命令作出响应。只有在最近一次测温后遇到符合报警条件的情况， DS18B20 才会响应这条命令。在每次报警搜索指令周期之后，总线控制器必须返回步骤 1。关于报警操作流程见 报警信号操作 节。

DS18B20 功能指令

在总线控制器发给欲连接的 DS18B20 一条ROM 命令后，跟着可以发送一条 DS18B20 功能指令。这些命令允许总线控制器读写 DS18B20 的暂存器，发起温度转换和识别电源模式。 DS18B20 的功能指令详见下文，同时被概括于表 4，并用流程图示于图 12。

CONVERT T [44h] (温度转换指令)

这条命令用以启动一次温度转换。 温度转换指令被执行， 产生的温度转换结果数据以2个字节的形式被存储在高速暂存器中，而后 DS18B20 保持等待状态。如果寄生电源模式下发出该命令后，在温度转换期间（ tconv），必须在 10us(最多), 内给单总线一个强上拉，见 DS18B20 供电 节。如果 DS18B20 以外部电源供电，总线控制器在发出该命令后跟着发出读时序， DS18B20 如处于转换中，将在总线上返回 0，若温度转换完成，则返回 1。寄生电源模式下，总线被强上拉拉高前这样的通讯技术不会被使用。

WRITE SCRATCHPAD [4Eh] （写暂存器指令）

这条命令向 DS18B20 的暂存器写入数据，开始位置在 TH 寄存器（暂存器的第 2个字节），接下来写入 TL 寄存器（暂存器的第 3 个字节），最后写入配置寄存器（暂存器的第 4 个字节）。数据以最低有效位开始传送。上述三个字节的写入必须发生在总线控制器发出复位命令前，否则会中止写入。

READ SCRATCHPAD [BEh] （读暂存器指令）

这条命令读取暂存器的内容。读取将从字节 0 开始，一只进行下去，知道第 9字节（字节 8， CRC）读完，如果不想读完所有字节，控制器可以在任何时间发出复位命令来中止读取。

COPY SCRATCHPAD [48h] （拷贝暂存器指令）

这条命令把 TH,TL 和配置寄存器（第 2、3、4 字节）的内容拷贝到 EEPROM 中。如果使用寄生电源总线控制器必须在发出这条命令的 10us 内启动强上拉并最少保持 10ms，见 DS18B20 供电 节所述。

RECALL E2 [B8H] （召回 EEPROM 指令）

这条命令把报警触发器的值（ TH 和 TL）以及配置数据从 EEPROM 拷回暂存器。总线控制器在发出该命令后读时序， DS18B20 会输出拷回标识： 0标识正在拷回,

1标识拷回结束。这种拷回操作在 DS18B20 上电时自动执行，这样器件一上电暂存器里马上就存在有效的数据了。

READ POWER SUPPLY [B4h] (读电源模式指令)

总线控制器在这条命令发给 DS18B20 后发出读时序，若是寄生电源模式，DS18B20 将拉低总线，若是外部电源模式， DS18B20 将会把总线拉高。关于这条指令的用法信息详述于 DS18B20 供电 节。

DS18B20 功能指令表 表4

指令	说明	协议	单总线发出协议后	备注
温度转换命令	TEMPERATURE CONVERSION COMMANDS			
Convert T	开始温度转换	44h	读温度忙状态	1
存储器命令	MEMORY COMMANDS			
Read Scratchpad	读取暂存器和CRC字节	BEh	读数据直到第9字节	2
Write Scratchpad	把字节写入暂存器的肘子2，3和4（TH, TL和配置寄存器）	4Eh	写入3个字节到DS18B20	3
Copy Scratchpad	把TH, TL和配置寄存器拷贝到EEPROM中	48h	无	1
Recall E²	把非易失性存储器中的值召回暂存器（TH, TL和配置寄存器）	B8h	传送拷贝标识给总线控制器	
Read Power Supply	标识DS18B20的供电模式	B4h	返回供电状态	

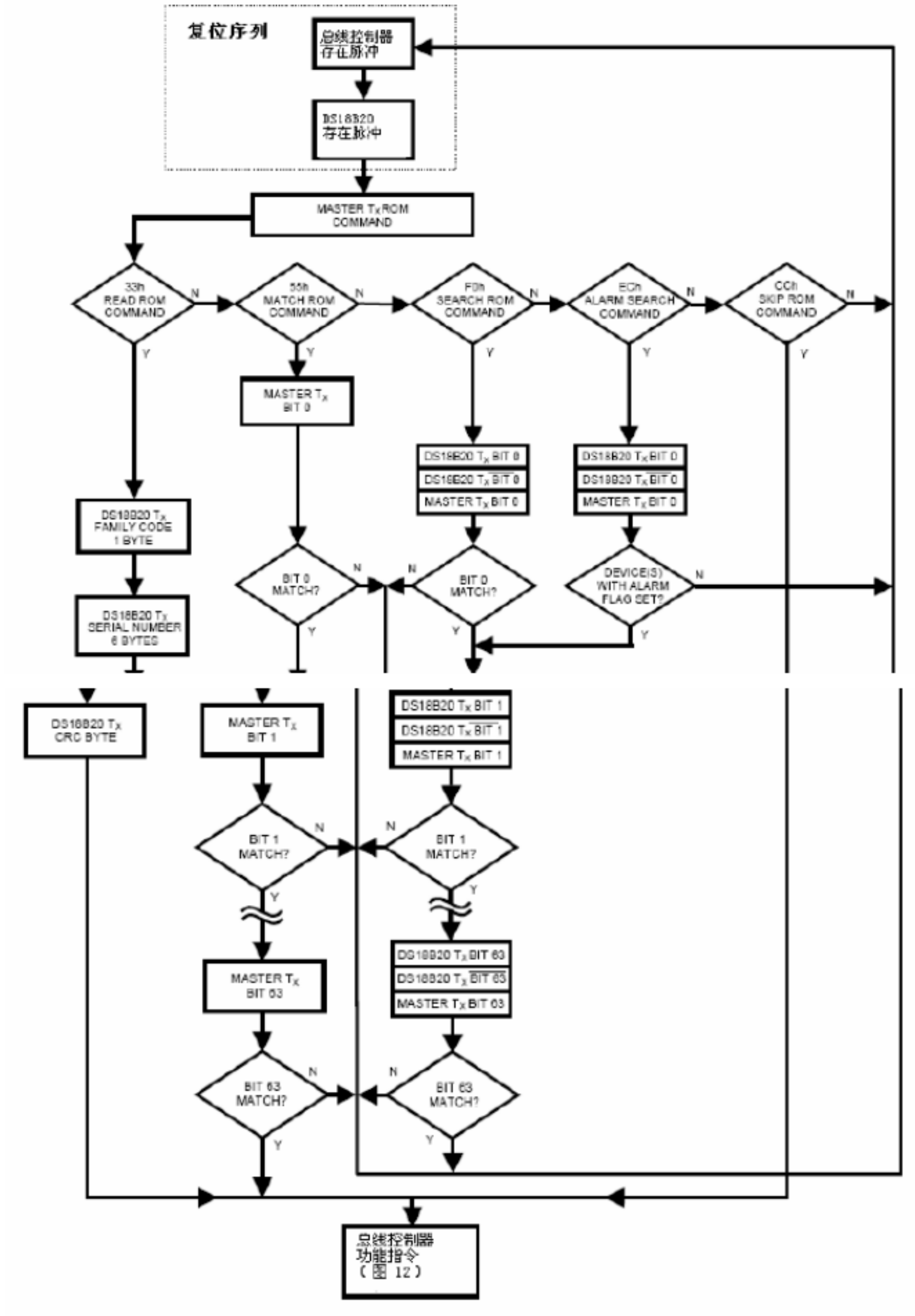
备注：

1. 对于寄生电源模式下的 DS18B20 ， 在温度转换和拷贝数据到 EEPROM 期间，必须给单总线一个强上拉。总线上在这段时间内不能有其它活动。

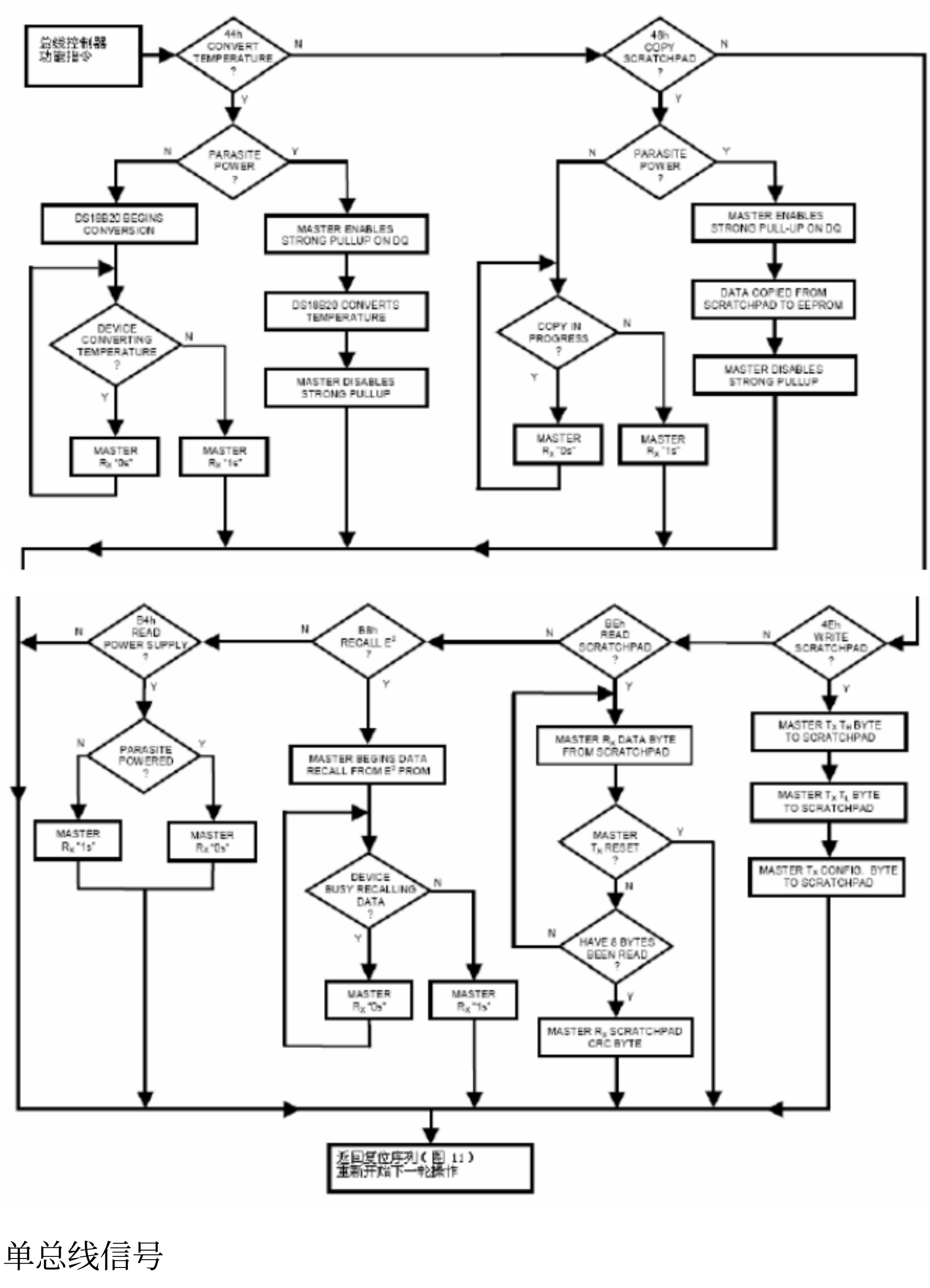
2. 总线控制器在任何时刻都可以通过发出复位信号中止数据传输。

3. TH,TL 和配置寄存器这 3个字节的写入必须在复位信号发起之前。

ROM 指令流程图 图11



DS18B20　功能指令流程图 图12



单总线信号

DS18B20 需要严格的单总线协议以确保数据的完整性。协议包括集中单总线信号类型：复位脉冲、存在脉冲、写 0、写1、读0 和读1。所有这些信号，除存在脉冲外，都是由总线控制器发出的。

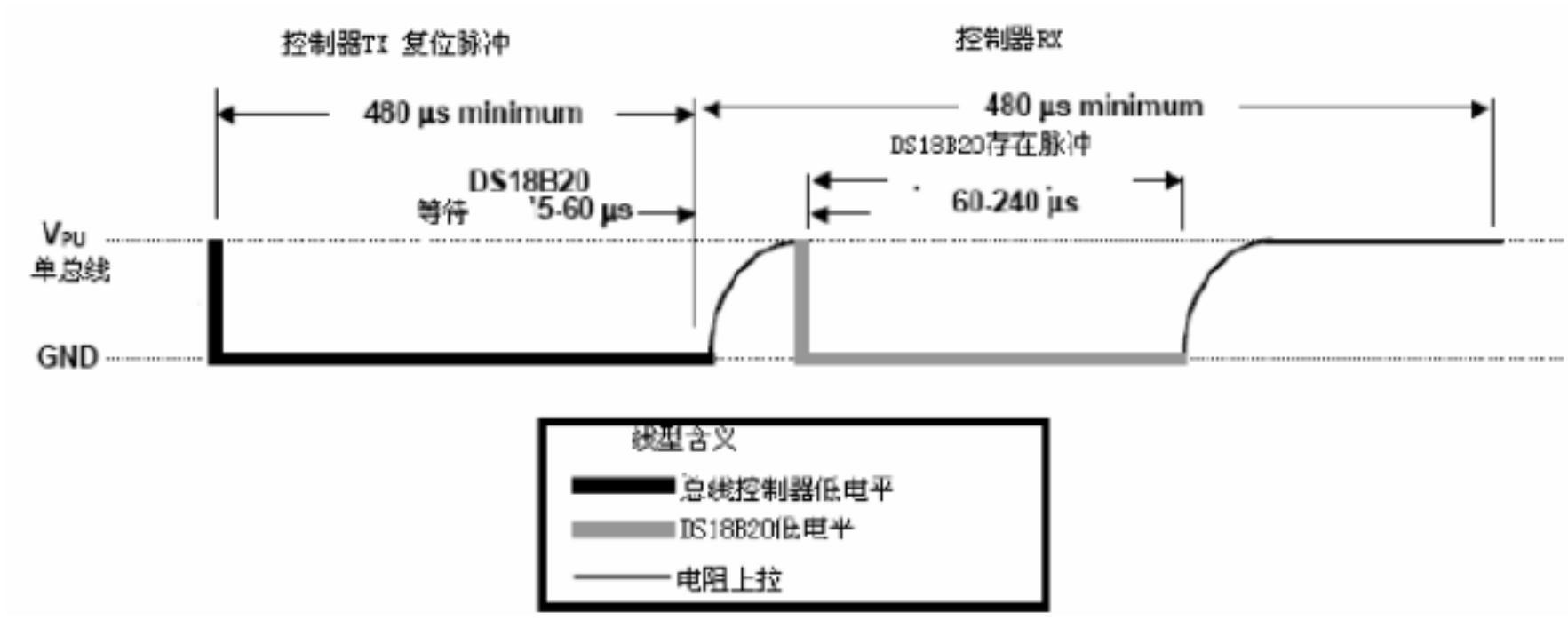
复位序列：复位和存在脉冲

和 DS18B20 间的任何通讯都需要以初始化序列开始，初始化序列见图 13。一个复位脉冲跟着一个存在脉冲表明 DS18B20 已经准备好发送和接收数据。

在初始化序列期间， 总线控制器拉低总线并保持 480us 以发出（TX）一个复位脉

冲，然后释放总线，进入接收状态（ RX）。单总线由 5K上拉电阻拉到高电平。当DS18B20 探测到 I/O引脚上的上升沿后，等待 15-60us, 然后发出一个由 60-240us 低电平信号构成的存在脉冲。

初始化时序 图13



读/写时序

DS18B20 的数据读写是通过时序处理位来确认信息交换的。

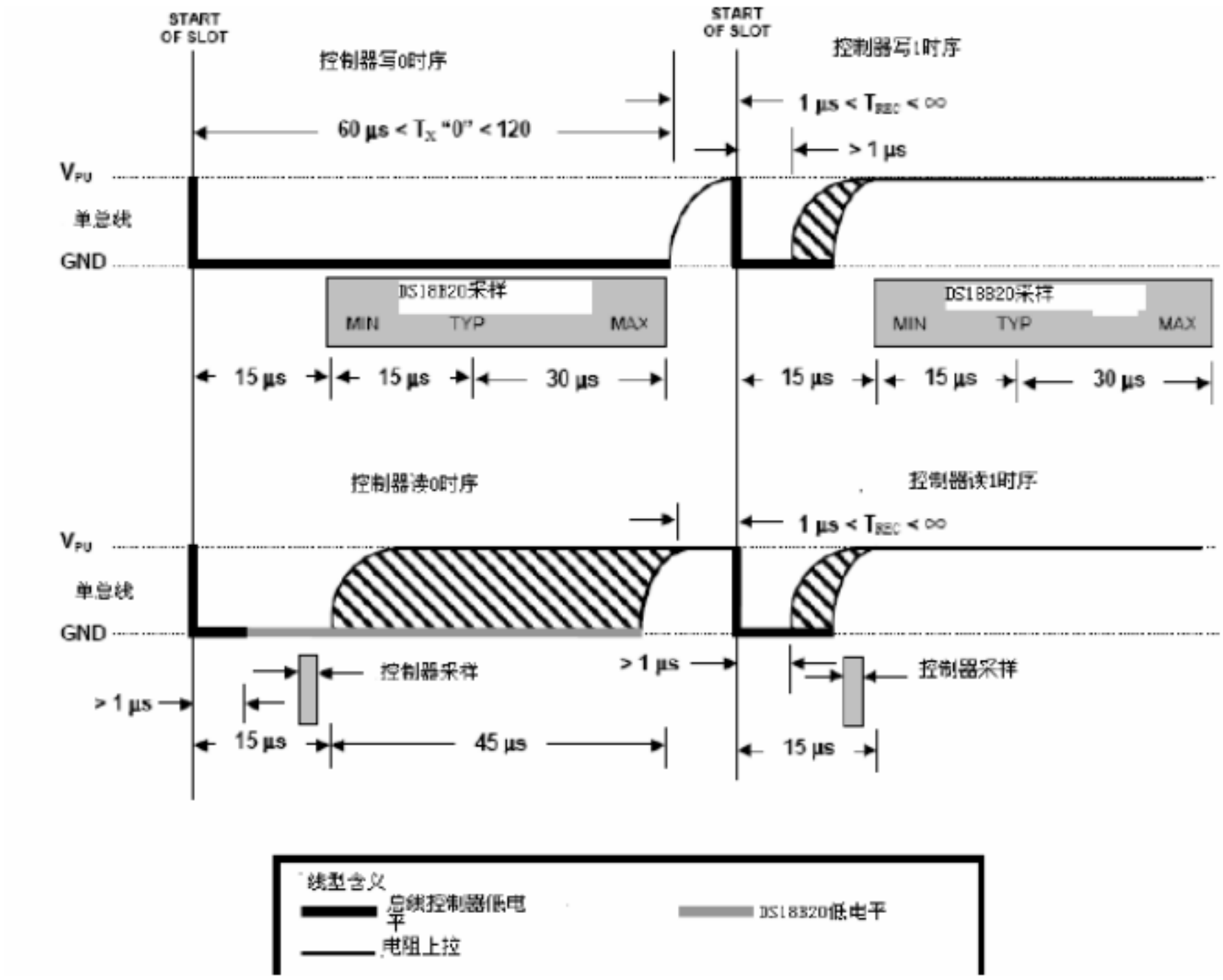
写时序

由两种写时序：写 1 时序和写 0 时序。总线控制器通过写 1 时序写逻辑 1 到 DS18B20，写0 时序写逻辑 0 到DS18B20 。所有写时序必须最少持续 60us，包括两个写周期之间至少 1us 的恢复时间。当总线控制器把数据线从逻辑高电平拉到低电平的时候，写时序开始（见图 14）。

总线控制器要生产一个写时序， 必须把数据线拉到低电平然后释放， 在写时序开始后的 15us 释放总线。当总线被释放的时候， 5K的上拉电阻将拉高总线。总控制器要生成一个写 0时序，必须把数据线拉到低电平并持续保持（至少 60us）。

总线控制器初始化写时序后， DS18B20 在一个 15us 到60us 的窗口内对 I/O线采样。如果线上是高电平，就是写 1。如果线上是低电平，就是写 0。

读/写时序图 图14

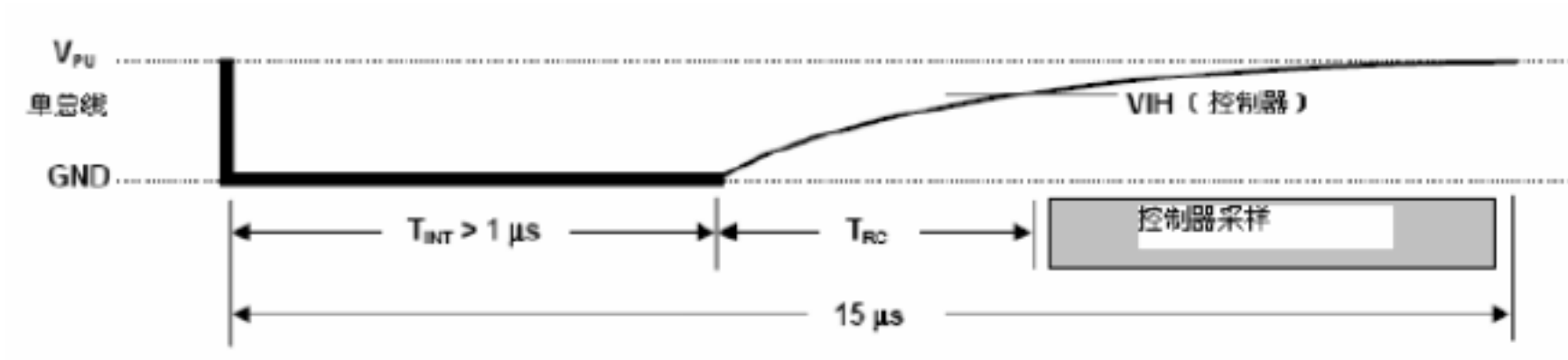


读时序

总线控制器发起读时序时， DS18B20 仅被用来传输数据给控制器。因此，总线控制器在发出读暂存器指令 [BEh] 或读电源模式指令 [B4H] 后必须立刻开始读时序， DS18B20 可以提供请求信息。除此之外，总线控制器在发出发送温度转换指令 [44h] 或召回 EEPROM 指令 [B8h] 之后读时序，详见 DS18B20 功能指令 节。所有读时序必须最少 60us, 包括两个读周期间至少 1us 的恢复时间。当总线控制器把数据线从高电平拉到低电平时，读时序开始，数据线必须至少保持 1us, 然后总线被释放（见图 14）。在总线控制器发出读时序后， DS18B20 通过拉高或拉低总线上来传输 1或0。当传输逻辑 0结束后，总线将被释放，通过上拉电阻回到上升沿状态。从 DS18B20 输出的数据在读时序的下降沿出现后 15us 内有效。因此，总线控制器在读时序开始后必须停止把 I/O脚驱动为低电平 15us, 以读取 I/O 脚状态。

图 15 标识 TINIT,TRC 和TSAMPLE 之和必须小于 15us。图16 指出，系统时间可以用下面办法达到最大： TINIT 和TRC 保持时间尽可能校；把控制器采样时间放到 15us 周期的最后。

控制器读 1 的详细时序 图15



DS18B20 Programmable Resolution 1-Wire Digital Thermometer

FEATURES

Unique 1-Wire ? Interface Requires Only One Port Pin for Communication

Each Device has a Unique 64-Bit Serial Code Stored in an On-Board ROM

Multidrop Capability Simplifies Distributed Temperature-Sensing Applications

Requires No External Components

Can Be Powered from Data Line; Power Supply Range is 3.0V to 5.5V

Measures Temperatures from -55°C to +125°C (-67°F to +257°F)

±0.5°C Accuracy from -10°C to +85°C

Thermometer Resolution is User Selectable from 9 to 12 Bits

Converts Temperature to 12-Bit Digital Word in 750ms (Max)

User-Definable Nonvolatile (NV) Alarm Settings

Alarm Search Command Identifies and Addresses Devices Whose Temperature is Outside Programmed Limits (Temperature Alarm Condition)

Software Compatible with the DS1822

Applications Include Thermostatic Controls, Industrial Systems, Consumer Products, Thermometers, or Any Thermally Sensitive System

PIN CONFIGURATIONS

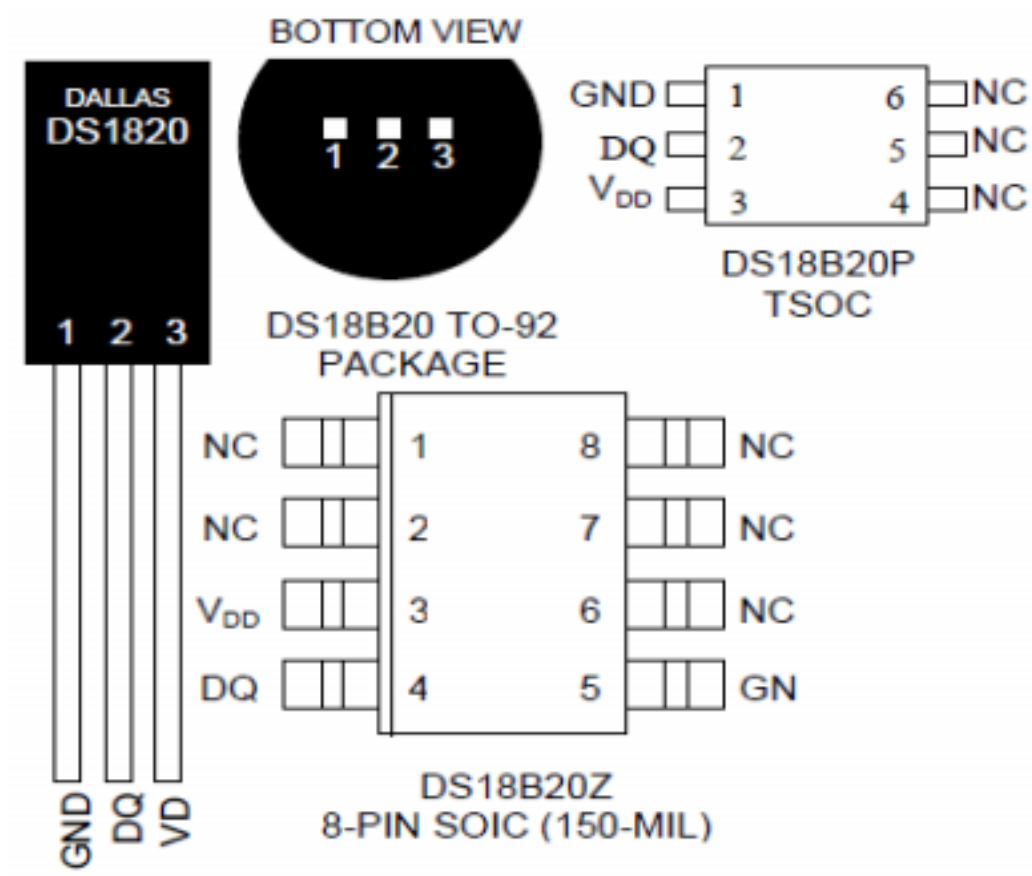


Figure 1

DESCRIPTION

The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. It has an operating temperature range of -55°C to +125 °C and is accurate to ±0.5 °C over the range of -10 °C to +85 °C. In addition, the DS18B20 can derive power directly from the data line (“parasite power”), eliminating the need for an external power supply.

Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s distributed over a large area. Applications that can benefit from this feature include HVAC environmental controls, temperature monitoring systems inside buildings, equipment, or machinery, and process monitoring and control systems

PIN DESCRIPTION

SOIC	TO-92	NAME	FUNCTION
5	1	GND	Ground
4	2	V_{DD}	Data Input/Output. Open-drain 1-Wire interface pin. Also provides power to the device when used in parasite power mode (see the *Powering the DS18B20* section.)
3	3	DQ	Optional V_{DD}. V_{DD} must be grounded for operation in parasite power mode.

Table 1

OVERVIEW

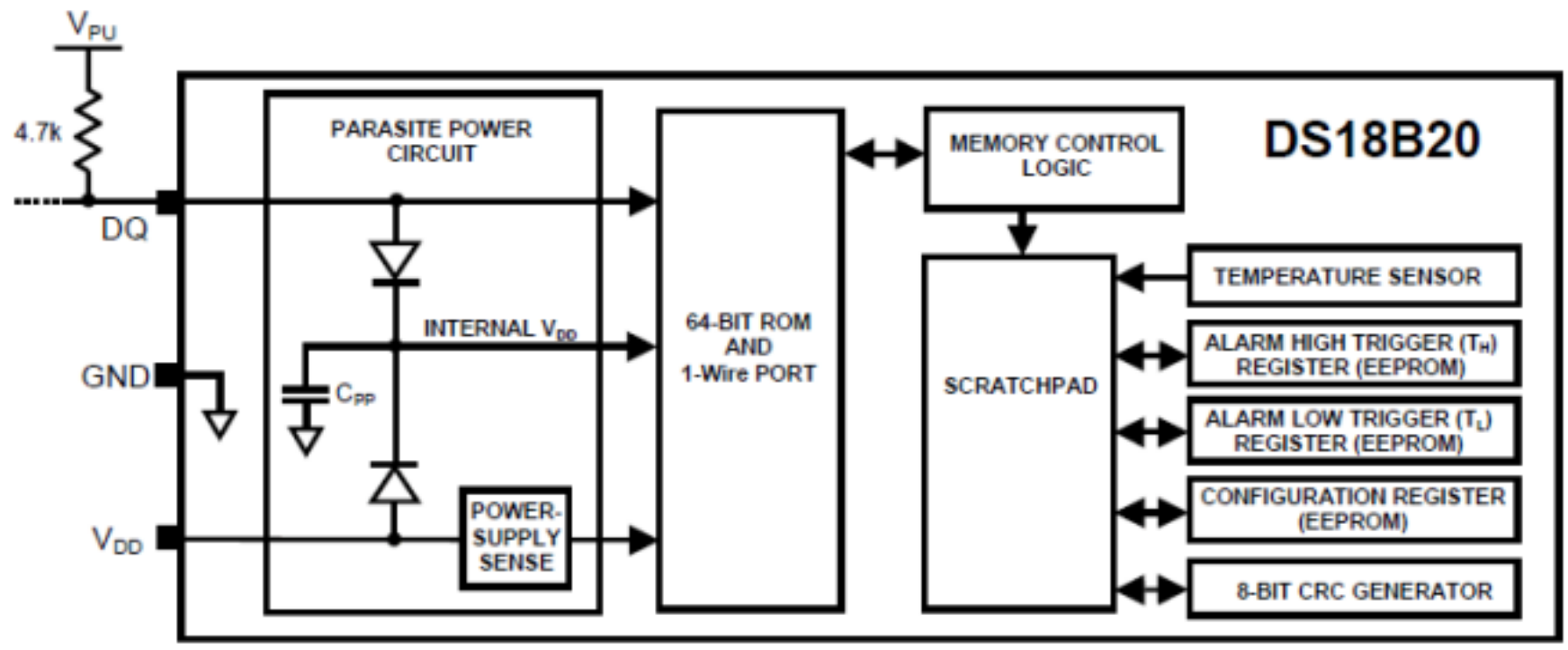
Figure 2 shows a block diagram of the DS18B20, and pin descriptions are given in the Pin Description table. The 64-bit ROM stores the device’s unique serial code. The scratchpad memory contains the 2-byte temperature register that stores the digital output from the temperature sensor. In addition, the scratchpad provides access to the 1-byte upper and lower alarm trigger registers (T_H and T_L) and the 1-byte configuration register. The configuration register allows the user to set the resolution of the temperature-to-digital conversion to 9, 10, 11, or 12 bits. The T_H, T_L, and configuration registers are nonvolatile (EEPROM), so they will retain data when the device is powered down.

The DS18B20 uses Maxim’s exclusive 1-Wire bus protocol that implements bus communication using one control signal. The control line requires a weak pullup resistor since all devices are linked to the bus via a 3-state or open-drain port (the DQ pin in the case of the DS18B20). In this bus system, the microprocessor (the master device) identifies and addresses devices on the bus using each device’s unique 64-bit code. Because each device has a unique code, the number of devices that can be addressed on one bus is virtually unlimited. The 1-Wire bus protocol, including

detailed explanations of the commands and time slots, is covered in the 1-Wire Bus System section.

Another feature of the DS18B20 is the ability to operate without an external power supply. Power is instead supplied through the 1-Wire pullup resistor via the DQ pin when the bus is high. The high bus signal also charges an internal capacitor (C_{PP}), which then supplies power to the device when the bus is low. This method of deriving power from the 1-Wire bus is referred to as parasite power. As an alternative, the DS18B20 may also be powered by an external supply on V_{DD}.

Figure 2. DS18B20 Block Diagram



OPERATION—MEASURING TEMPERATURE

The core functionality of the DS18B20 is its direct-to-digital temperature sensor. The resolution of the temperature sensor is user-configurable to 9, 10, 11, or 12 bits, corresponding to increments of 0.5° C, 0.25 ° C, 0.125 ° C, and 0.0625° C, respectively. The default resolution at power-up is 12-bit. The DS18B20 powers up in a low-power idle state. To initiate a temperature measurement and A-to-D conversion, the master must issue a Convert T [44h] command. Following the conversion, the resulting thermal data is stored in the 2-byte temperature register in the scratchpad memory and the DS18B20 returns to its idle state. If the DS18B20 is powered by an external supply, the master can issue read time slots (see the 1-Wire Bus System section) after the Convert T command and the DS18B20 will respond by transmitting 0 while the temperature conversion is in progress and 1 when the conversion is done. If the DS18B20 is powered with parasite power, this notification technique cannot be used since the bus must be pulled high by a strong pullup during the entire temperature conversion. The bus requirements for parasite power are explained in detail in the Powering the DS18B20section.

Figure 3. Temperature Register Format

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	BIT 15	**BIT 14**	**BIT 13**	**BIT 12**	**BIT 11**	**BIT 10**	**BIT 9**	**BIT 8**
MS BYTE	S	S	S	S	S	2^6	2^5	2^4

S = SIGN

Table 2. Temperature/Data Relationship

TEMPERATURE (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	0000 0111 1101 0000	07D0h
+85*	0000 0101 0101 0000	0550h
+25.0625	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
-25.0625	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

*The power-on reset value of the temperature register is +85 C. °

OPERATION —ALARM SIGNALING

After the DS18B20 performs a temperature conversion, the temperature value is compared to the userdefined two's complement alarm trigger values stored in the 1-byte T_H and T_L registers (see Figure 3). The sign bit (S) indicates if the value is positive or negative: for positive numbers S = 0 and for negative numbers S = 1. The T_H and T_L registers are nonvolatile (EEPROM) so they will retain data when the device is powered down. T_H and T_L can be accessed through bytes 2 and 3 of the scratchpad as explained in the Memory section.

Figure 4. TH and TL Register Format

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
S	2^6	2^5	2^5	2^5	2^2	2^1	2^0

Only bits 11 through 4 of the temperature register are used in the T_H and T_L comparison since T_H and T_L are 8-bit registers. If the measured temperature is lower than or equal to T_L or higher than or equal to T_H, an alarm condition exists and an alarm flag is set inside the DS18B20. This flag is updated after every temperature measurement; therefore, if the alarm condition goes away, the flag will be turned off after the next temperature conversion.

The master device can check the alarm flag status of all DS18B20s on the bus by issuing an Alarm Search [ECh] command. Any DS18B20s with a set alarm flag will respond to the command, so the master can determine exactly which DS18B20s have experienced an alarm condition. If an alarm condition exists and the T_H or T_L settings have changed, another temperature conversion should be done to validate the alarm condition.

64-BIT LASERED ROM CODE

Each DS18B20 contains a unique 64bit code (see Figure 6) stored in ROM. The least significant 8 bits of the ROM code contain the DS18B20 1-Wire's family code: 28h. The next 48 bits contain a unique serial number. The most significant 8 bits contain a cyclic redundancy check (CRC) byte that is calculated from the first 56 bits of the ROM code. A detailed explanation of the CRC bits is provided in the CRC Generation section. The 64-bit ROM code and associated ROM function control logic allow the DS18B20 to operate as a 1-Wire device using the protocol detailed in the 1-Wire Bus System section.

Figure 5. 64-Bit Lasered ROM Code

8-BIT CRC	48-BIT SERIAL NUMBER	8-BIT FAMILY CODE (28h)

MEMORY

The DS18B20's memory is organized as shown in Figure 6. The memory consists of an SRAM scratchpad with nonvolatile EEPROM storage for the high and low alarm trigger registers (T_H and T_L) and configuration register. Note that if the DS18B20 alarm function is not used, the T_H and T_L registers can serve as general-purpose memory. All memory commands are described in detail in the DS18B20 Function Commands section.

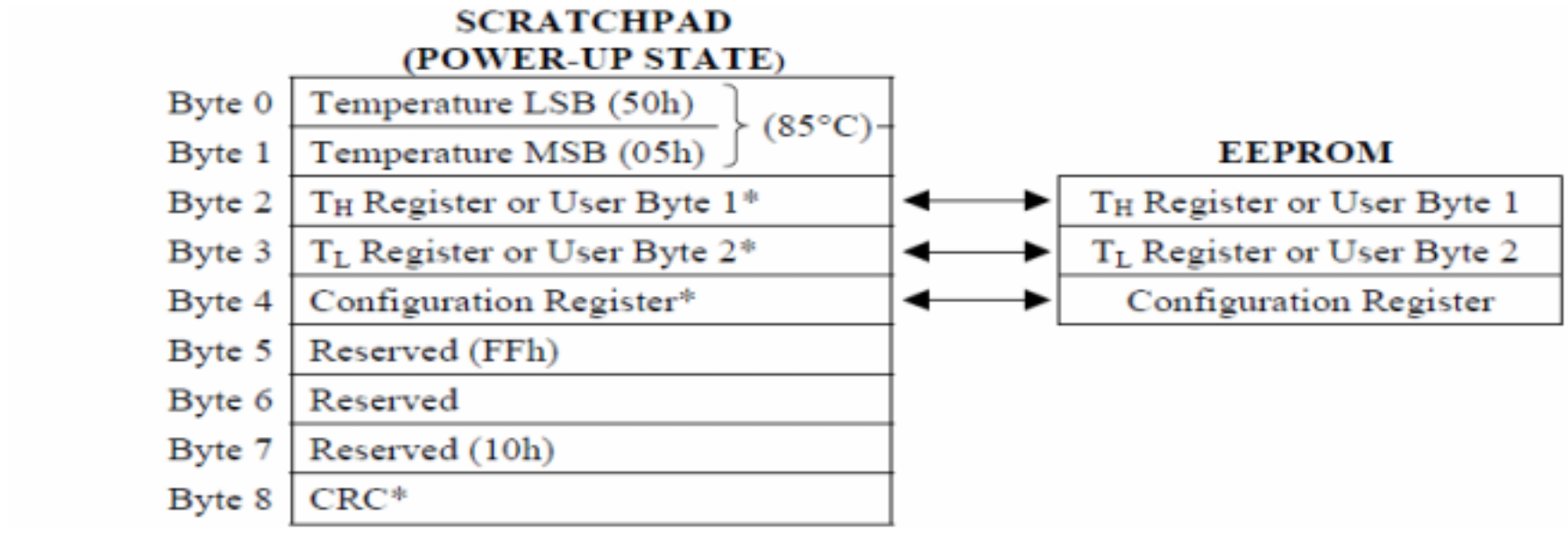
Byte 0 and byte 1 of the scratchpad contain the LSB and the MSB of the temperature register, respectively. These bytes are read-only. Bytes 2 and 3 provide access to T_H and T_L registers. Byte 4 contains the configuration register data, which is explained in detail in the Configuration Register section. Bytes 5, 6, and 7 are reserved for internal use by the device and cannot be overwritten.

Byte 8 of the scratchpad is read-only and contains the CRC code for bytes 0 through 7 of the scratchpad. The DS18B20 generates this CRC using the method described in the CRC Generation section.

Data is written to bytes 2, 3, and 4 of the scratchpad using the Write Scratchpad [4Eh] command; the data must be transmitted to the DS18B20 starting with the least significant bit of byte 2. To verify data integrity, the scratchpad can be read (using the Read Scratchpad [BEh] command) after the data is written. When reading the scratchpad, data is transferred over the 1-Wire bus starting with the least significant bit of byte 0. To transfer the T_H, T_L and configuration data from the scratchpad to EEPROM, the master must issue the Copy Scratchpad [48h] command.

Data in the EEPROM registers is retained when the device is powered down; at power-up the EEPROM data is reloaded into the corresponding scratchpad locations. Data can also be reloaded from EEPROM to the scratchpad at any time using the Recall E^2 [B8h] command. The master can issue read time slots following the Recall E^2 command and the DS18B20 will indicate the status of the recall by transmitting 0 while the recall is in progress and 1 when the recall is done.

Figure 6. DS18B20 Memory Map



* Power-up state depends on value(s) stored in EEPROM.

CONFIGURATION REGISTER

Byte 4 of the scratchpad memory contains the configuration register, which is organized as illustrated in Figure 8. The user can set the conversion resolution of the

DS18B20 using the R0 and R1 bits in this register as shown in Table 2. The power-up default of these bits is R0 = 1 and R1 = 1 (12-bit resolution). Note that there is a direct tradeoff between resolution and conversion time. Bit 7 and bits 0 to 4 in the configuration register are reserved for internal use by the device and cannot be overwritten.

Figure 7. Configuration Register

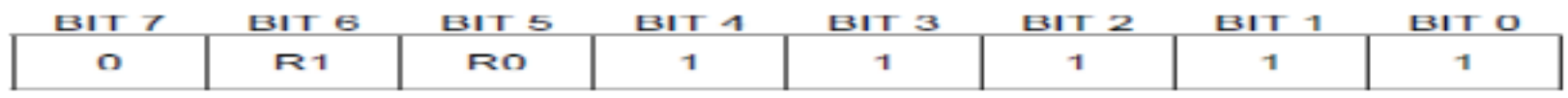


Table 3. Thermometer Resolution Configuration

R1	R0	RESOLUTION (BITS)	MAX CONVERSION TIME	
0	0	9	93.75ms	(t_{CONV}/8)
0	1	10	187.5ms	(t_{CONV}/4)
1	0	11	375ms	(t_{CONV}/2)
1	1	12	750ms	(t_{CONV})

CRC GENERATION

CRC bytes are provided as part of the DS18B20 64-bit ROM code and in the 9th byte of the scratchpad memory. The ROM code CRC is calculated from the first 56 bits of the ROM code and is contained in the most significant byte of the ROM. The scratchpad CRC is calculated from the data stored in the scratchpad, and therefore it changes when the data in the scratchpad changes. The CRCs provide the bus master with a method of data validation when data is read from the DS18B20. To verify that data has been read correctly, the bus master must re-calculate the CRC from the received data and then compare this value to either the ROM code CRC (for ROM reads) or to the scratchpad CRC (for scratchpad reads). If the calculated CRC matches the read CRC, the data has been received error free. The comparison of CRC values and the decision to continue with an operation are determined entirely by the bus master. There is no circuitry inside the DS18B20 that prevents a command sequence from proceeding if the DS18B20 CRC (ROM or scratchpad) does not match the value generated by the bus master.

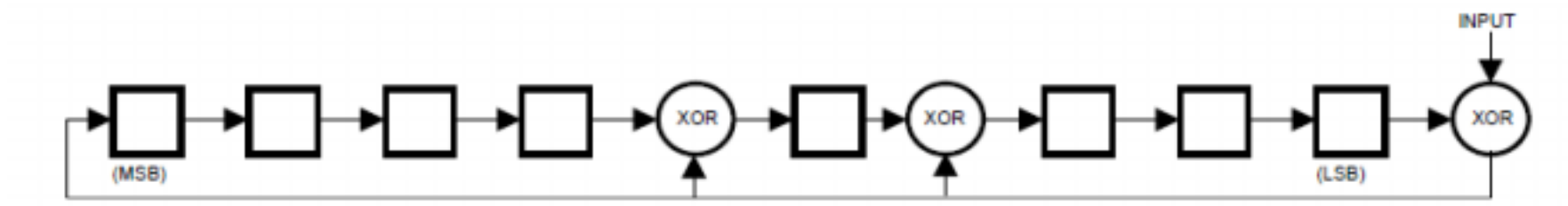
The equivalent polynomial function of the CRC (ROM or scratchpad) is:

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

The bus master can re-calculate the CRC and compare it to the CRC values from the DS18B20 using the polynomial generator shown in Figure8. This circuit consists of a shift register and XOR gates, and the shift register bits are initialized to 0. Starting with the least significant bit of the ROM code or the least significant bit of byte 0 in the scratchpad, one bit at a time should shifted into the shift register. After shifting in the 56th bit from the ROM or the most significant bit of byte 7 from the scratchpad, the polynomial generator will contain the re-calculated CRC. Next, the 8-bit ROM code or scratchpad CRC from the DS18B20 must be shifted into the circuit. At this point, if the re-calculated CRC was correct, the shift register will contain all 0s. Additional information about the Maxim 1-Wire cyclic redundancy check is available inApplication Note 27: Understanding and Using Cyclic

Redundancy Checks with Maxim iButton Products.

Figure 9. CRC Generator



1-WIRE BUS SYSTEM

The 1-Wire bus system uses a single bus master to control one or more slave devices. The DS18B20 is always a slave. When there is only one slave on the bus, the system is referred to as a "single-drop" system; the system is "multidrop" if there are multiple slaves on the bus.

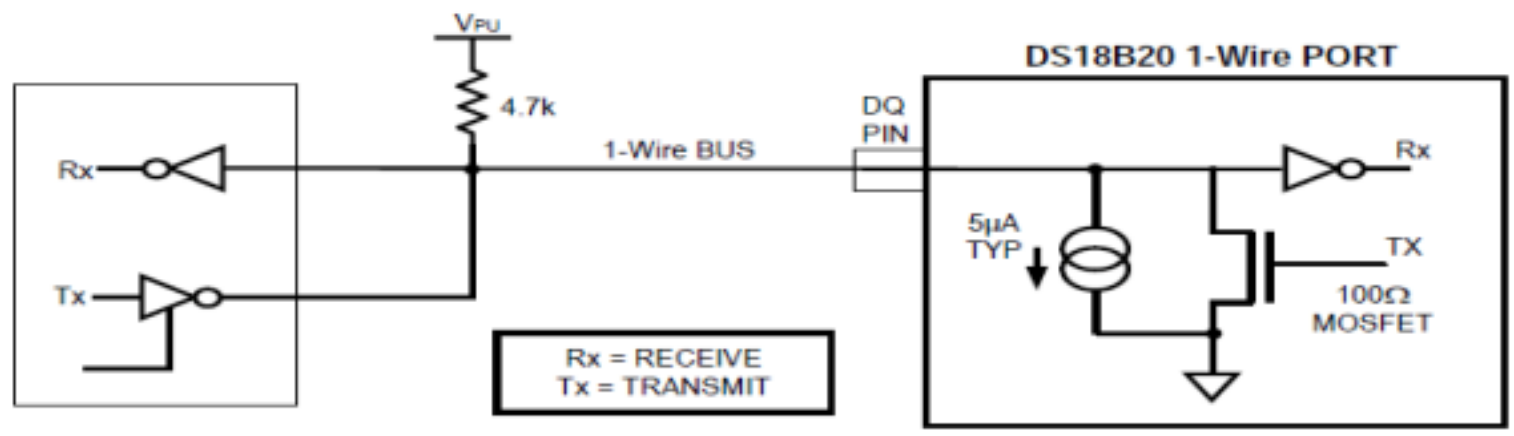
All data and commands are transmitted least significant bit first over the 1-Wire bus.The following discussion of the 1-Wire bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-Wire signaling (signal types and timing).

HARDWARE CONFIGURATION

The 1-Wire bus has by definition only a single data line. Each device (master or slave) interfaces to the data line via an open-drain or 3-state port. This allows each device to "release" the data line when the device is not transmitting data so the bus is available for use by another device. The 1-Wire port of the DS18B20 (the DQ pin) is open drain with an internal circuit equivalent to that shown in Figure 9.

The 1-Wire bus requires an external pullup resistor of approximately 5kΩ; thus, the idle state for the 1-Wire bus is high. If for any reason a transaction needs to be suspended, the bus MUST be left in the idle state if the transaction is to resume. Infinite recovery time can occur between bits so long as the 1-Wire bus is in the inactive (high) state during the recovery period. If the bus is held low for more than 480μs, all components on the bus will be reset.

Figure 9. Hardware Configuration



TRANSACTION SEQUENCE

The transaction sequence for accessing the DS18B20 is as follows:

Step 1. Initialization

Step 2. ROM Command (followed by any required data exchange)

Step 3. DS18B20 Function Command (followed by any required data exchange)

It is very important to follow this sequence every time the DS18B20 is accessed, as the DS18B20 will not respond if any steps in the sequence are missing or out of order. Exceptions to this rule are the Search ROM [F0h] and Alarm Search [ECh] commands. After issuing either of these ROM commands, the master must return to

Step 1 in the sequence.

INITIALIZATION

All transactions on the 1-Wire bus begin with an initialization sequence. The initialization sequence consists of a reset pulse transmitted by the bus master followed by presence pulse(s) transmitted by the slave(s). The presence pulse lets the bus master know that slave devices (such as the DS18B20) are on the bus and are ready to operate. Timing for the reset and presence pulses is detailed in the 1-Wire Signaling section.

ROM COMMANDS

After the bus master has detected a presence pulse, it can issue a ROM command. These commands operate on the unique 64-bit ROM codes of each slave device and allow the master to single out a specific device if many are present on the 1-Wire bus. These commands also allow the master to determine how many and what types of devices are present on the bus or if any device has experienced an alarm condition. There are five ROM commands, and each command is 8 bits long. The master device must issue an appropriate ROM command before issuing a DS18B20 function command. A flowchart for operation of the ROM commands is shown in Figure 11.

1-WIRE SIGNALING

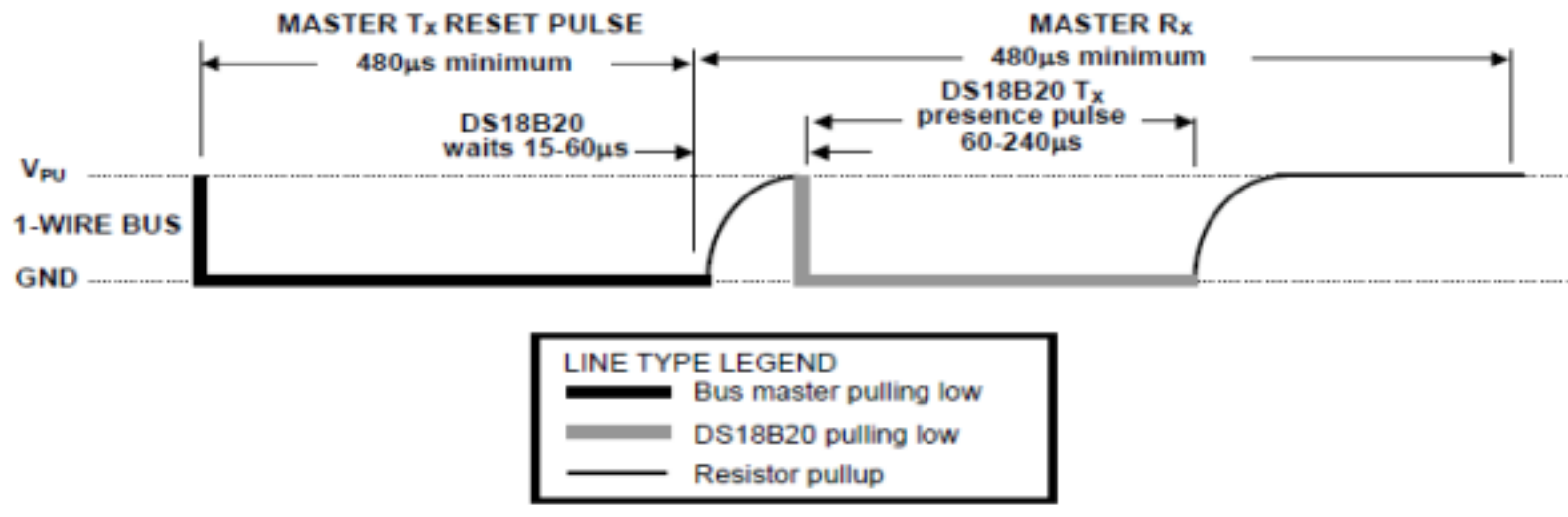
The DS18B20 uses a strict 1-Wire communication protocol to ensure data integrity. Several signal types are defined by this protocol: reset pulse, presence pulse, write 0, write 1, read 0, and read 1. The bus master initiates all these signals, with the exception of the presence pulse.

INITIALIZATION PROCEDURE —RESET AND PRESENCE PULSES

All communication with the DS18B20 begins with an initialization sequence that consists of a reset pulse from the master followed by a presence pulse from the DS18B20. This is illustrated in Figure 13. When the DS18B20 sends the presence pulse in response to the reset, it is indicating to the master that it is on the bus and ready to operate.

During the initialization sequence the bus master transmits (Tx) the reset pulse by pulling the 1-Wire bus low for a minimum of 480 μs. The bus master then releases the bus and goes into receive mode (Rx). When the bus is released, the 5kΩ pullup resistor pulls the 1-Wire bus high. When the DS18B20 detects this rising edge, it waits 15μs to 60μs and then transmits a presence pulse by pulling the 1-Wire bus low for 60 μs to 240μs.

Figure 10. Initialization Timing



READ/WRITE TIME SLOTS

The bus master writes data to the DS18B20 during write time slots and reads data from the DS18B20 during read time slots. One bit of data is transmitted over the 1-Wire bus per time slot.

WRITE TIME SLOTS

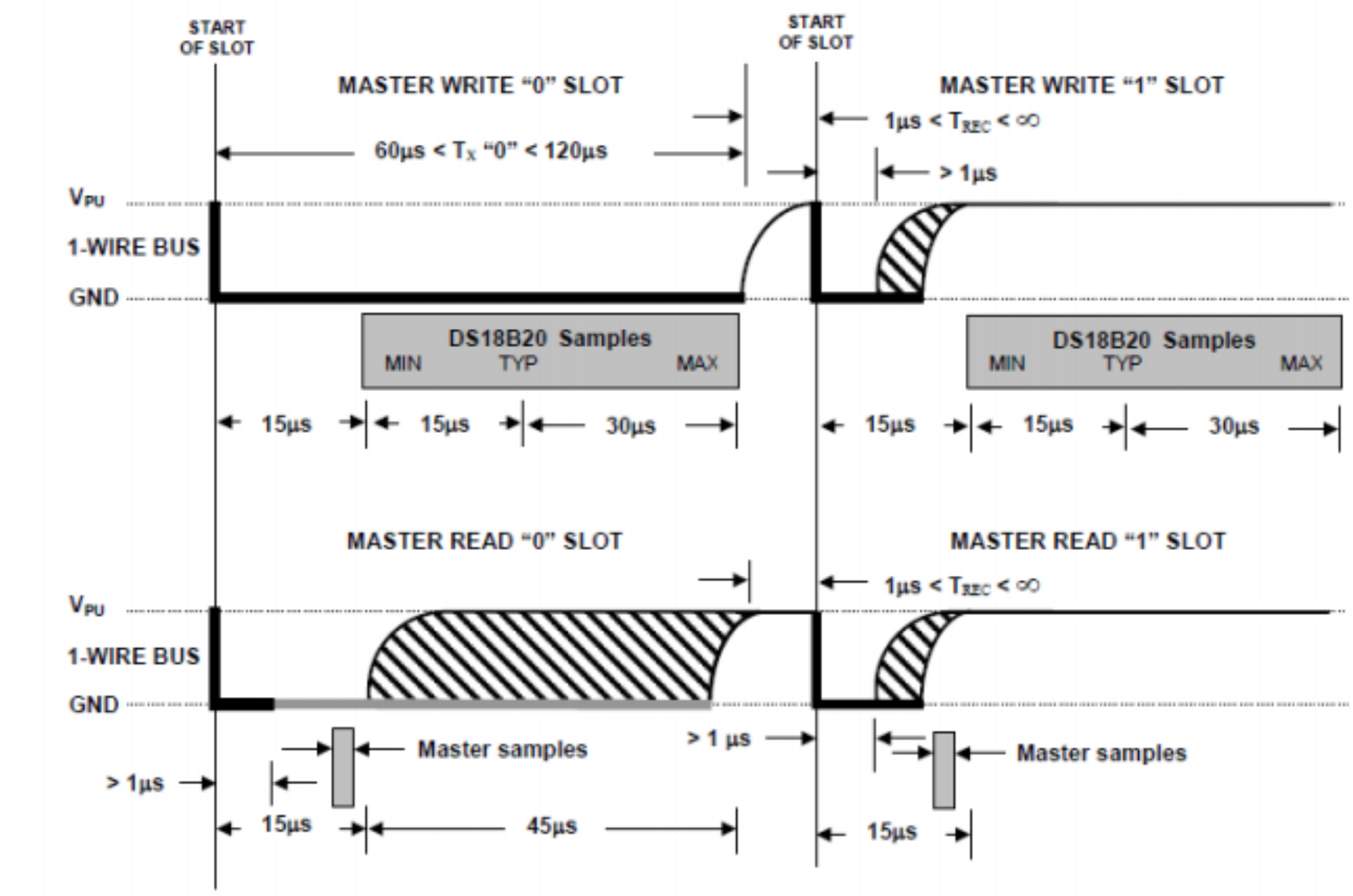
There are two types of write time slots: —Write 1 ‖ time slots and —Write 0 ‖ t

The bus master uses a Write 1 time slot to write a logic 1 to the DS18B20 and a Write 0 time slot to write a logic 0 to the DS18B20. All write time slots must be a minimum of 60 μ s in duration with a minimum of a 1 μ s recovery time between individual write slots. Both types of write time slots are initiated by the master pulling the 1-Wire bus low (see Figure 10).

To generate a Write 1 time slot, after pulling the 1-Wire bus low, the bus master must release the 1-Wire bus within 15μ s. When the bus is released, the 5kΩ pullup resistor will pull the bus high. To generate a Write 0 time slot, after pulling the 1-Wire bus low, the bus master must continue to hold the bus low for the duration of the time slot (at least 60μ s).

The DS18B20 samples the 1-Wire bus during a window that lasts from 15μ s to 60μ s after the master initiates the write time slot. If the bus is high during the sampling window, a 1 is written to the DS18B20. If the line is low, a 0 is written to the DS18B20.

Figure 11. Read/Write Time Slot Timing Diagram



LINE TYPE LEGEND

Bus master pulling low

DS18B20 pulling low

Resistor pullup

READ TIME SLOTS

The DS18B20 can only transmit data to the master when the master issues read time slots. Therefore, the master must generate read time slots immediately after issuing a Read Scratchpad [BEh] or Read Power Supply [B4h] command, so that the DS18B20 can provide the requested data. In addition, the master can generate read time slots after issuing Convert T [44h] or Recall E^2 [B8h] commands to find out the status

of the operation as explained in the DS18B20 Function Commands section.

All read time slots must be a minimum of 60µs in duration with a minimum of a 1µs recovery time between slots. A read time slot is initiated by the master device pulling the 1-Wire bus low for a minimum of 1 µs and then releasing the bus (see Figure 11). After the master initiates the read time slot, the DS18B20 will begin transmitting a 1 or 0 on bus. The DS18B20 transmits a 1 by leaving the bus high and transmits a 0 by pulling the bus low. When transmitting a 0, the DS18B20 will release the bus by the end of the time slot, and the bus will be pulled back to its high idle state by the pullup resister. Output data from the DS18B20 is valid for 15µs after the falling edge that initiated the read time slot. Therefore, the master must release the bus and then sample the bus state within 15µs from the start of the slot.

Figure 12 illustrates that the sum of T_{INIT}, T_{RC}, and T_{SAMPLE} must be less than 15 µs for a read time slot.

Figure 13shows that system timing margin is maximized by keeping T_{INIT} and T_{RC} as short as possible and by locating the master sample time during read time slots towards the end of the 15µs period.

Figure 12. Detailed Master Read 1 Timing

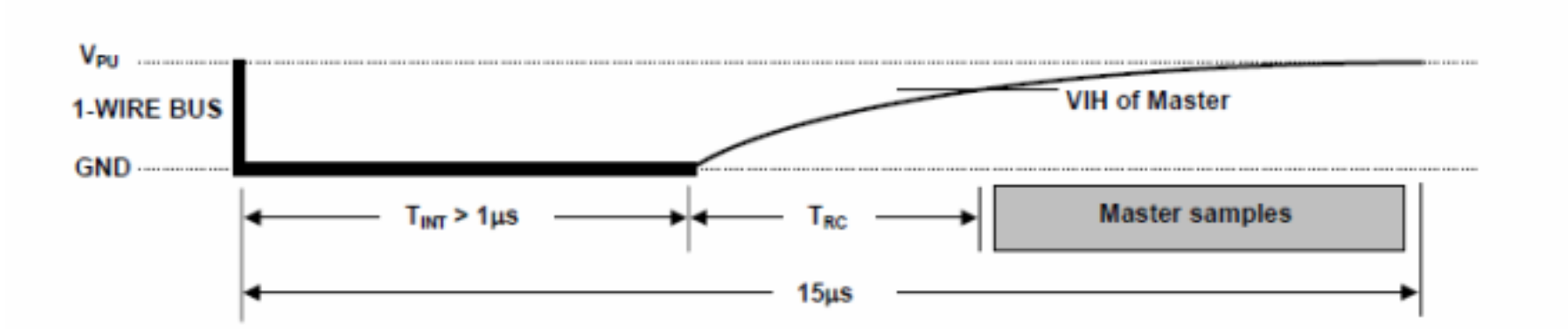


Figure 13. Recommended Master Read 1 Timing