



GXCAS Technology

GX20MH01

可编程分辨率单总线高精度温度传感器

概述

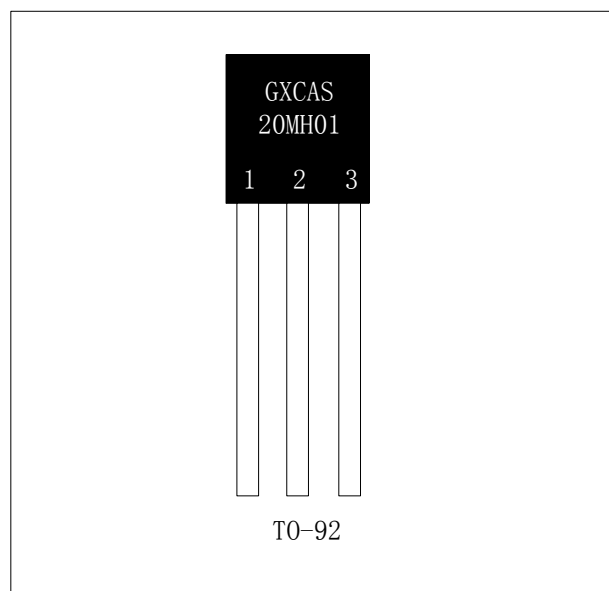
GX20MH01 数字温度计提供 14bit 分辨率的温度测量，可以通过可编程非易失性存储单元实现温度的下限和上限报警。GX20MH01 采用单总线协议与上位机进行通信，只需要一根信号线和一根地线。它的温度测量范围为 -55°C ~ $+125^{\circ}\text{C}$ 。在 0°C ~ $+50^{\circ}\text{C}$ 范围内的测试精度可以达到 $\pm 0.1^{\circ}\text{C}$ 。此外它还可以工作在寄生模式下，直接通过信号线对芯片供电，从而不需要额外的供电电源。每个 GX20MH01 都有一个全球唯一的 64 位序列号，可以将多个 GX20MH01 串联在同一跟单总线上进行组网，只需要一个处理器就可以控制分布在大面积区域中的多颗 GX20MH01。这种组网方式特别适合 HVAC 环境控制，建筑、设备、粮情测温 and 工业测温以及过程监测控制等应用领域。

特征

- 采用单总线接口仅需一个端口引脚进行通信
- 每颗芯片具有全球唯一的 64 位的序列号
- 具有多点分布式测温功能
- 无需外围元器件
- 可通过数据线供电；供电电压范围为 2.5V~5.5V

- 测度测量范围为 -55°C to $+125^{\circ}\text{C}$ (-67°F to $+257^{\circ}\text{F}$)
- 在 0°C ~ 50°C 范围内精确度为 $\pm 0.1^{\circ}\text{C}$
- 12 位精度下，温度转换速度小于 400ms
- 具有用户自定义的非易失性温度报警设置
- 报警搜索命令识别并标识超过程序设定温度的器件
- 超强静电保护能力：HBM>8000V
MM >800V
- 3 脚的 TO-92 封装
- 应用包括体温测量、药品储藏、医疗设备等领域。

PIN CONFIGURATIONS



管脚描述

管脚位置	名称	作用
TO-92		
3	V _{DD}	电源管脚；在寄生供电模式下 V _{DD} 管脚必须连接到地
2	DQ	数据输入输出管脚；当寄生供电模式下，该管脚给芯片供电（请见 GX20MH01 的电源说明部分）
1	GND	接地管脚

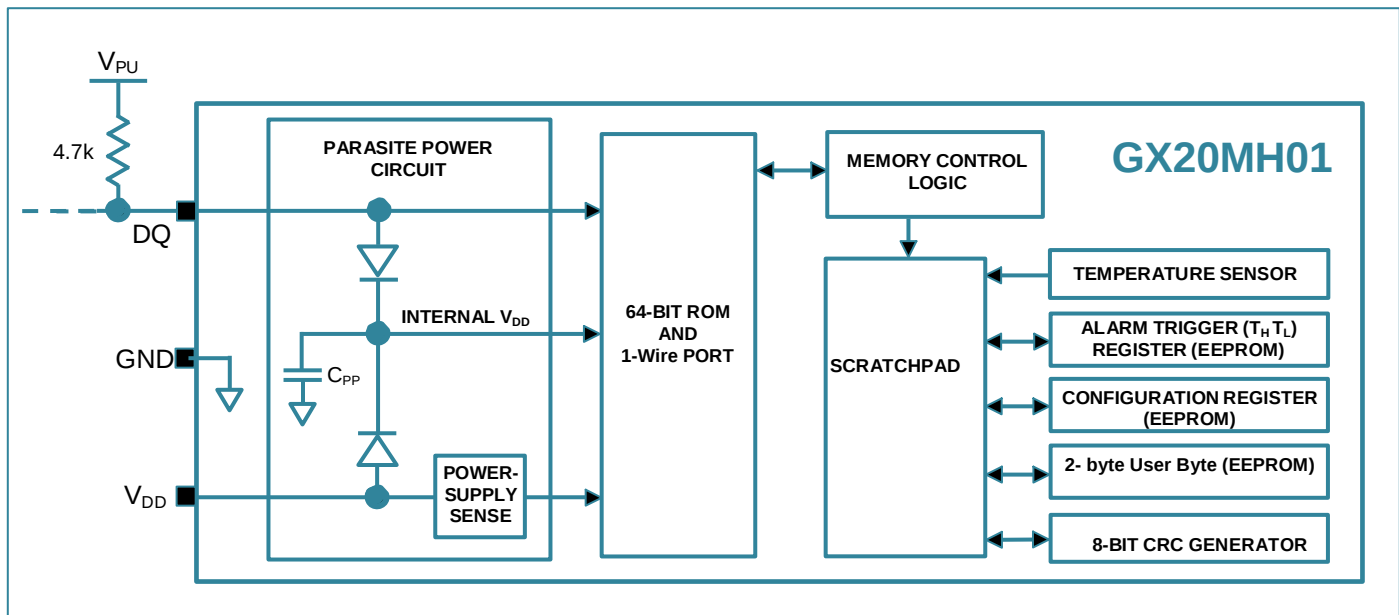
概况

如图 1 所示为 GX20MH01 的结构框图。芯片采用 64 位的只读存储器存储器件的唯一片序列号。芯片内部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，芯片还提供一个字节的温度报警阈值寄存器（TH 和 TL）和一个字节的配置寄存器。配置寄存器允许用户将温度测量分辨率设定为 9，10，11，12，13 或者 14 位，但为了保持精度建议用户设置为 12 位。TH，TL 和配置寄存器均为非易失性的可擦除寄存器（EEPROM），该存储的数据在器件掉电后不会消失。

GX20MH01 采用单总线协议，通过一个单线端口通信。当全部器件经由一个三态端口或者开漏端口与总线连接的时候，控制线需要连接一个弱上拉电阻。在这个总线系统中，微处理器（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唯一的片序列号，总线可以连接的器件数量实际上是无限的。单总线协议，包括指令的详细解释和“时序”见**单总线系统节**。

GX20MH01 的另一个功能是可以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（C_{PP}）充电，在总线处于低电平状态时，该电容提供能量给器件，该提供能量的方式成为“寄生电源”。当然 GX20MH01 可以通过 VDD 管脚连接外部电源供电。

图 1. GX20MH01 的结构框图



测温操作

GX20MH01 的核心功能是它的直接数字温度传感器。温度传感器的精度为用户可编程的 9，10，11，12，13 和 14 位。温度分辨率分别为 0.5℃、0.25℃、0.125℃、0.0625℃、0.03125℃ 和 0.015625℃。芯片在上电状态下默认的精度为 14 位。GX20MH01 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格式被存储到温度寄存器中，GX20MH01 继续保持等待状态。GX20MH01 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），GX20MH01 正在温度转换中返回 0，转换结束返回 1。如果 GX20MH01 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，否则将不会有返回值。寄生电源的总线要求在 GX20MH01 供电节详细解释。

图 2. 温度寄存器格式

CF BYTE	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
	R2	R1	R0	1	1	1	2 ⁻⁵	2 ⁻⁶
LS BYTE	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹	2 ⁻²	2 ⁻³	2 ⁻⁴
MS BYTE	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
	S	S	S	S	S	2 ⁶	2 ⁵	2 ⁴

表 1. 温度/数据关系

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

*上电复位时温度寄存器默认值为+85℃

报警操作

GX20MH01 完成一次温度转换后，就用该温度值与存储在 **T_H** 和 **T_L** 寄存器（如图 3 所示）中的一个字节的用户自定义的报警阈值进行比较。标志位（**S**）指示温度值的正负：正数 **S**=0，负数 **S**=1。**T_H** 和 **T_L** 寄存器是非易失性的（EEPROM），所以在掉电后数据仍然保持。在存储器节将解释 **T_H** 和 **T_L** 是如何存入寄存器的第 2 和第 3 个字节的。

图 3. **T_H** and **T_L** 寄存器格式

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

当 **T_H** 和 **T_L** 为 8 位寄存器时，在与 **T_H** 和 **T_L** 的比较中仅适用温度寄存器的 4 到 11 位。如果测得的温度高于 **T_H** 或低于 **T_L**，报警条件成立，GX20MH01 内部就会置位一个报警标识。没进行一次测温就对这个标识进行一个更新。因此，如果报警状态消失，则在下一个温度转换后，该标识将被关闭。

总线控制器通过发出报警搜索命令[ECh]检测总线上所有的 GX20MH01 报警标识。任何置位报警标识的 GX20MH01 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 GX20MH01。如果报警条件成立，而 **T_H** 或 **T_L** 的设置已经改变，另一个温度转换将重新确认报警条件。

GX20MH01 的供电

GX20MH01 可以通过管脚 VDD 外部电源供电，也可以工作于寄生电源模式。寄生电源模式下允许 GX20MH01 工作于外部电源需求状态。寄生电源模式在远距离测试或空间受限的应用场合是非常有用的。寄生电源的控制回路如图 1 所示，当总线为高电平时，该控制回路从总线上“窃取”能量。部分“窃取”的能量存储在寄生电源储能电容（C_{pp}）内，在总线处于低电平时释放该

能量提供给器件使用。当 GX20MH01 处于寄生电源模式时，VDD 管脚必须接地。

寄生电源模式下，单总线和 C_{PP} 在大部分操作中能够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 GX20MH01。然而，当 GX20MH01 正在执行温度转换或从寄存器向 EEPROM 传送数据时，工作电流可高达 1.5mA。这个电流可能会引起连接在单总线上的弱上拉电阻不可接受的压降，这需要更大的电流，而此时 C_{PP} 无法提供。为了保证 GX20MH01 有充足的供电，当进行温度转换或复制数据到 EEPROM 操作时，必须给单总线提供一个强上拉，采用 MOSFET 直接把总线上拉到电源上的方式实现，如图 4 所示。在发出温度转换指令[44h]或拷贝寄存器指令[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(t_{VONV})或拷贝数据时序($t_{ER}=10ms$)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 GX20MH01 供电的另外一种方法是传统的从 VDD 管脚接入一个外部电源，如图 5 所示。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。

对于温度高于 100°C 时，不推荐使用寄生电源，因为 GX20MH01 在这种温度下表现出的漏电流比较大，通信可能无法进行。在类似这种温度的情况下，强烈推荐使用 GX20MH01 的 VDD 管脚供电。

对于总线控制器不知道总线上的 GX20MH01 是使用寄生电源还是使用外部电源的情况，GX20MH01 预备了一种信号指示电源的使用示意图。总线控制器发出一个 Skip ROM 指令[CCh]，然后发出度电源指令[B4h]，这条指令发出后，控制器发出读时序命令，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要在温度转换期间对单总线提供强上拉。

Figure 4. Supplying the Parasite-Powered GX20MH01 During Temperature Conversions

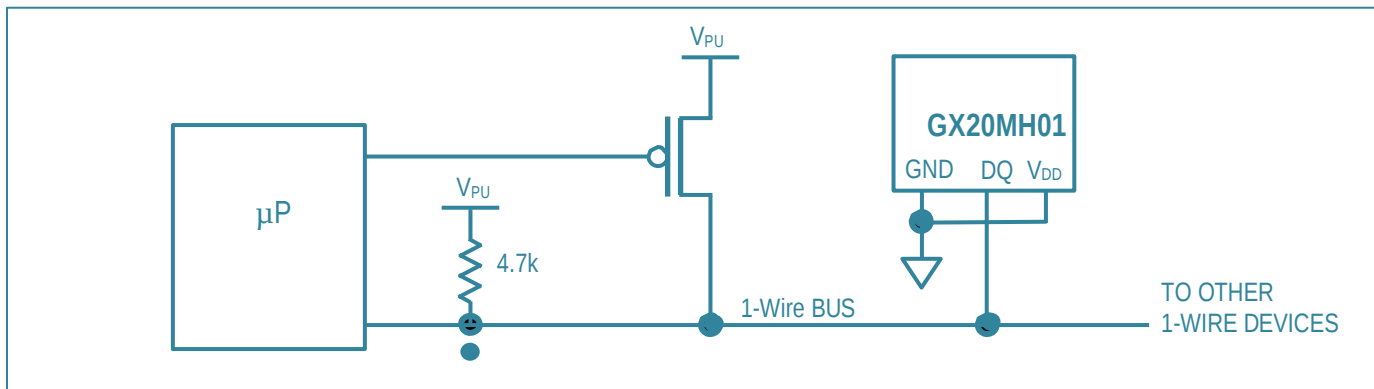
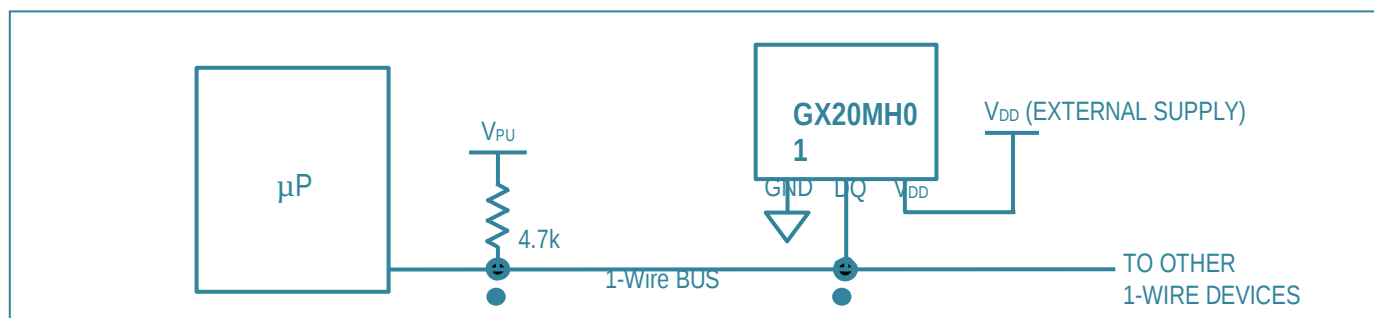


Figure 5. Powering the GX20MH01 with an External Supply



64 位只读存储器

每颗 GX20MH01 都有一个全球唯一的存储在 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 GX20MH01 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。

图 6. 64 位 ROM 码

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

存储器

GX20MH01 的存储器结构如图 7 所示。存储器有一个暂存器 SRAM 和一个存储底稿报警阈值 T_H 和 T_L 的非易失性电可擦除 EEOROM 组成。注意当报警功能不能使用时， T_H 和 T_L 寄存器可以被当做普通寄存器使用。所有的存储器指令被详述于 GX20MH01 功能指令节。

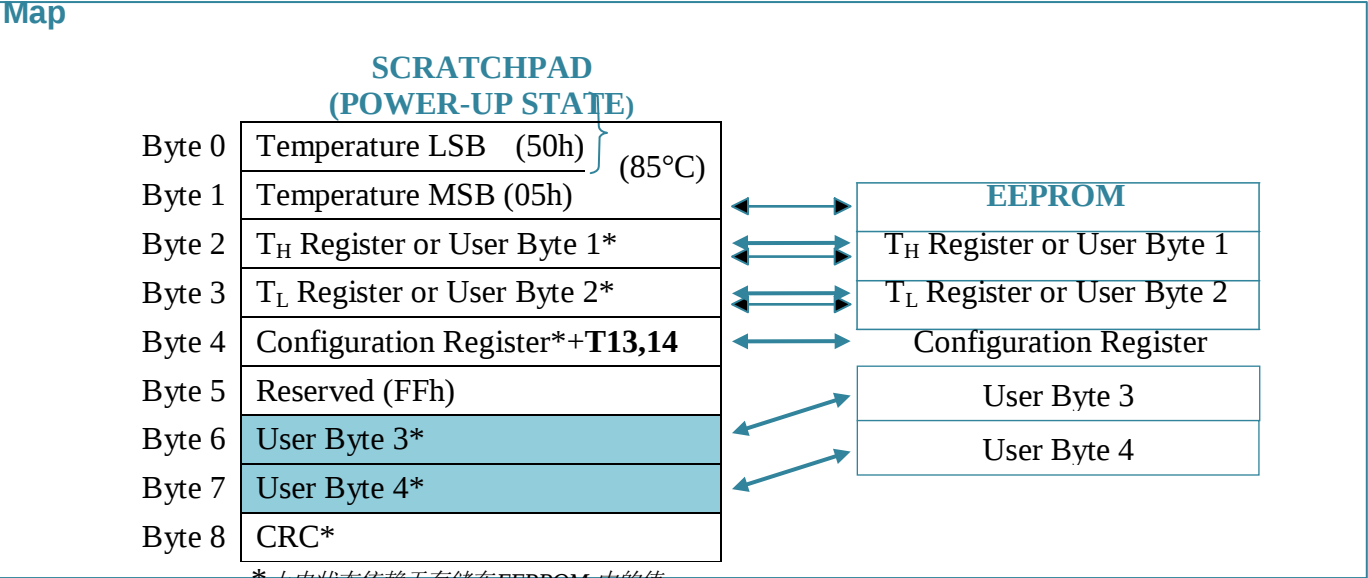
存储器的 byte 0 和 byte 1 字节分别为温度寄存器的 LSB 和 MSB，这两个字节的存储器为只读存储器。第 2 和第 3 字节是 T_H 和 T_L 。第 4 字节主要是配置寄存器数据，但该寄存器的最低的 2bits 被用作温度寄存器，详述于配置寄存器节。第 5 字节被器件保留，禁止写入；第 6 和第 7 字节用户可以使用。

存储器的第 8 字节是只读的，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。

数据通过写寄存器指令[4Eh]写入存储器的 2,3,4,6 和 7 为；数据必须以第 2 个字节为最低有效位开始传送。为了完整的验证数据，存储器能够在数据写入后被读取（使用读寄存器指令[BEh]）。在读寄存器时，数据以字节 0 为最低有效位从单总线移出。总线控制器从寄存器到 EEPROM 传递 T_H 、 T_L 和配置数据时必须发出拷贝寄存器指令[48h]。

EEPROM 存储器中的数据在器件掉电后仍然保持；上电时，数据被载入寄存器。数据也可以通过召回 EEPROM 命令从寄存器载入到 EEPROM 中。总线控制器在发出这条命令后发出读时序，GX20MH01 返回 0 表示正在召回中，返回 1 表示操作结束。

Figure 7. GX20MH01 Memory Map



配置寄存器

存储器的第 4 个字节是配置寄存器，其结构见图 8。用户可以通过表 2 所示设置 R0、R1 和 R2 位来设定 GX20MH01 的精度。上电默认设置：R0=1，R1=1，R2=1（14 位精度）。注意：精度和转换时间之间有直接的关系。配置寄存器的位 7 和位 0 到 4 被器件保留，禁止写入；在读数据时，它们全部表现为逻辑 1。

Figure 8. 配置寄存器

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
R2	R1	R0	1	1	1	2 ⁻⁵	2 ⁻⁶

Table 2. 传感器精度配置表

R2	R1	R0	RESOLUTION (BITS)	MAX CONVERSION TIME	
0	0	0	9	30ms	
0	0	1	10	60ms	
0	1	0	11	120ms	
0	1	1	12	250ms	
1	0	0	13	500ms	
1	0	1	14	1000ms	(t _{CONV})
1	1	*	14	1000ms	(t _{CONV})

CRC 发生器

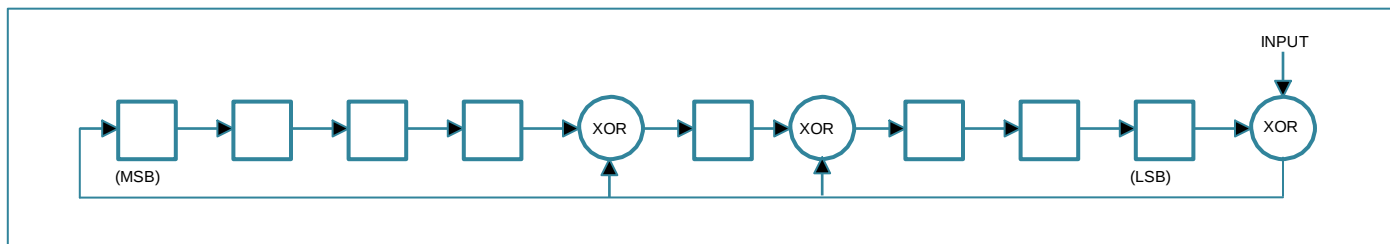
CRC 作为 GX20MH01 64 位 ROM 的一部分存储在存储器中。CRC 码由 ROM 的前 56 位计算得到，被包含在 ROM 的重要字节当中。CRC 由存储在存储器中的数据计算得到，因此当存储器中的数据发生改变时，CRC 的值也随之改变。

CRC 能够在总线控制器读取 GX20MH01 时进行数据校验。为了校验数据是否被正确读取，总线控制器必须用接收到的数据计算出一个 CRC 值和存储在 GX20MH01 的 64 位 ROM 中的值（读 ROM 时）或 GX20MH01 内部计算出的 8 位 CRC 值（读寄存器时）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 GX20MH01 中存储的或者由其计算得到的 CRC 值和总线控制器计算的值不相符时，GX20MH01 内部并没有一个能阻止命令序列进行的电路。CRC 的计算公式如下：

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

单总线 CRC 可以由一个移位寄存器和 XOR 门构成的多项式发生器来产生，如图 9 所示。这个回路包括一个移位寄存器和几个 XOR 门，移位寄存器的各个位都被初始化为 0。从 ROM 中的最低有效位或寄存器的 0 字节开始，一次一位移入寄存器。在传输了 56 位 ROM 中的数据或移入了寄存器的第 7 字节的最高位后，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 是争取的，移位寄存器将复位为 0。

图 9. CRC 产生器



单总线系统

单总线系统采用一个单总线控制器来控制一个或多个从器件。GX20MH01 总是充当从机。当只有一个从机挂在总线上时，系统被称为“单点”系统；如果由多个从机挂在总线上，系统被称为“多点”系统。

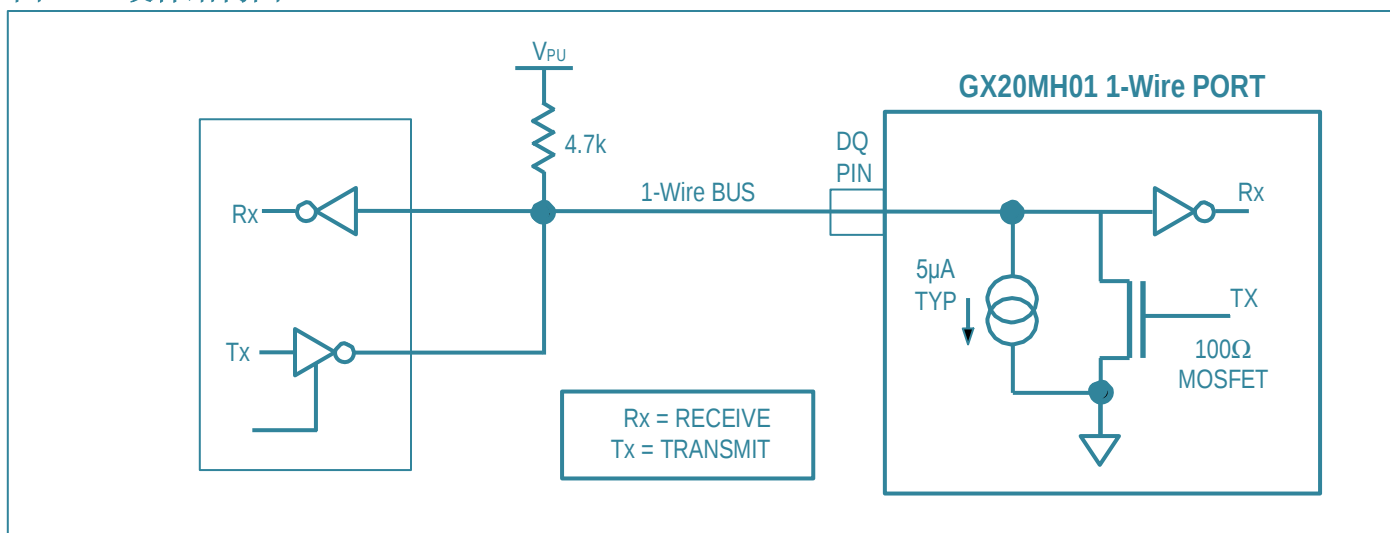
所有的数据和指令的传递都是从最低有效位开始通过单总线的。关于单总线系统分三个方面讨论：硬件结构、执行序列和单总线信号（信号类型和时序）。

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。每个总线上的器件（主或从）必须是漏极开路或三态输出。这样的机制就会使总线上的每一个不传输数据的器件释放总线来让其它器件使用。GX20MH01 的单总线端口（DQ 管脚）是漏极开路式的，内部等效电路如图 10 所示。

单总线需要一个约 $5K\Omega$ 的外部上拉电阻；单总线空闲状态是高电平。如有由于某种需要暂停某一传输，如果还想恢复该传输的话，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 $480\mu s$ ，总线上的所有器件都将被复位。

图 10. 硬件结构图



执行序列

通过单总线访问 GX20MH01 的执行序列如下：

步骤 1：初始化

步骤 2：ROM 操作指令

步骤 3：GX20MH01 功能指令

每一次 GX20MH01 的操作都必须满足以上步骤，若是缺少步骤或是顺序混乱，器件将不会有返回值。搜索 ROM 命令和报警搜索命令除外。当这两个命令执行时，主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都从一个初始化程序序列开始。初始化序列包含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 GX20MH01 在总线上且已经准备好操作，详见**总线信号**节。

ROM 指令

一旦总线控制器检测到一个存在脉冲，它就发出一条 ROM 指令。如果总线上挂有多颗 GX20MH01，这些指令将给予器件独有的 64 位 ROM 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颗，什么型号的器件挂在总线上，同样，他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 GX20MH01 功能指令之前发出一条 ROM 指令。ROM 指令操作图见图 11。

SEARCH ROM [F0h]

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 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。如果总线上只有一个从机，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。在每次搜索 ROM 指令之后，总线控制器必须返回步骤 1（初始化）。

READ ROM [33h]

只有在总线上存在单颗 GX20MH01 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 Search ROM 指令的情况下读取从机的 64 位序列码。如果总线上有不止一个从机而使用该命令时，所有从机试图同时传送信号时就会发生数据冲突。

MATCH ROM [55h]

MATCH ROM 指令后跟着 64 位 ROM 序列号，总线控制器在多点总线上定位一颗特定的从器件。只有和 64 位 ROM 序列号完全匹配的 GX20MH01 才能响应随后的存储器操作指令；所有和 64 位 ROM 序列号不匹配的从机都将等待复位脉冲。

SKIP ROM [CCh]

这条命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：当只有一个从机在总线上时，无论如何，忽略 ROM 指令之后只能跟着发出一条读取寄存器指令[BEh]。在单点总线情况下使用该命令，器件无需发回 64 为 ROM 编码，从而节省了时间。如果总线上不

止一个从机，若发出忽略 ROM 指令，由于多只从机同时传输数据，总线上就会发生数据冲突。

ALARM SEARCH [ECh]

这条指令的操作流程和搜索 ROM 指令相同，只有满足报警条件的从机才会对该命令作出响应。该命令允许主设备确定在最近一次的温度转换期间是否有任何 GX20MH01 经历了报警状态。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。

GX20MH01 功能指令

在总线控制器使用 ROM 命令来确定与其希望通信的 GX20MH01 之后，主机可以发出一个 GX20MH01 的功能指令。这些指令允许总线控制器读写 GX20MH01 的寄存器，发起温度转换和识别电源模式。GX20MH01 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]

这条命令时用于启动一次温度转换。温度转换指令被执行后，产生的温度转换结果数据以 2 个字节的形式被存储在温度寄存器中，而后 GX20MH01 保持低功耗的等待状态。如果在寄生供电模式下发出该指令，在温度转换期间（ t_{CONV} ），必须在 10 μs （最多）内给单总线一个强上拉，见 **GX20MH01 供电**节。如果 GX20MH01 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，GX20MH01 如处于转换中，则总线返回 0，若温度转换完成，则返回 1。在寄生供电模式下，总线被强上拉拉高前这样的通信方式不会被使用。

WRITE SCRATCHPAD [4Eh]

这条命令向 GX20MH01 的寄存器写入数据，开始位置在 TH 寄存器（寄存器的第 2 个字节），接下来写入 TL 寄存器（寄存器的第 3 个字节），最后写入配置寄存器（寄存器的第 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，否则会发生数据冲突。

READ SCRATCHPAD [BEh]

这条命令时主机读取寄存器命令。读取将从字节 0 的最低有效位开始，一直进行下去，直到第 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候发出复位命令来中止读取。

COPY SCRATCHPAD 48h]

这条命令把 T_H 、 T_L 和配置寄存器（第 2,3,4 字节）的内容复制到 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 10 μs 内启动强上拉并最少保持 10ms，见 **GX20MH01 供电**节所述。

RECALL E² B8h]

这条命令把 T_H 、 T_L 以及配置的数据从 EEPROM 拷回寄存器。总线控制器在发出该命令后发读时序，GX20MH01 会输出拷回标识：0 标识正在拷回，1 标识拷回结束。该操作在 GX20MH01 上电时自动执行，这样器件一上电寄存器里马上就存在有效的数据了。

READ POWER SUPPLY B4h]

总线控制器在这条命令发给 GX20MH01 后发出读时序，若是寄生电源模式，GX20MH01 将

拉低总线；若是外部电源模式，GX20MH01 将会把总线拉高。关于这条指令的用法信息详述于 GX20MH01 供电节。

Table 3. GX20MH01 功能指令表

COMMAND	DESCRIPTION	PROTOCOL	1-Wire BUS ACTIVITY AFTER COMMAND IS ISSUED	NOTES
TEMPERATURE CONVERSION COMMANDS				
Convert T	Initiates temperature conversion.	44h	GX20MH01 transmits conversion status to master (not applicable for parasite-	1
MEMORY COMMANDS				
Read Scratchpad	Reads the entire scratchpad including the CRC byte.	BEh	GX20MH01 transmits up to 9 data bytes to master.	2
Write Scratchpad	Writes data into scratchpad bytes 2, 3, 4, and 6, 7(T_H , T_L , configuration registers and User Bytes).	4Eh	Master transmits 3 or 4 or 5 data bytes to GX20MH01.	3
Copy Scratchpad	Copies T_H , T_L , config register and User Bytes data from the scratchpad to EEPROM.	48h	None	1
Recall E ²	Recalls T_H , T_L , config register and User Bytes data from EEPROM to the scratchpad.	B8h	GX20MH01 transmits recall status to master.	
Read Power Supply	Signals GX20MH01 power supply mode to the master.	B4h	GX20MH01 transmits supply status to master.	

Note 1: 对于寄生电源模式下 GX20MH01 在温度转换和拷贝数据到 EEPROM 期间爱你，必须给单总线一个强上拉，总线在这段时间内不能有其他活动。

Note 2: 总线控制器在任何时候都可以通过发出复位信号中止数据传输。

Note 3: T_H , T_L 和配置寄存器这三个字节的写入必须在复位信号发起之前。

Figure 11. ROM 指令流程图

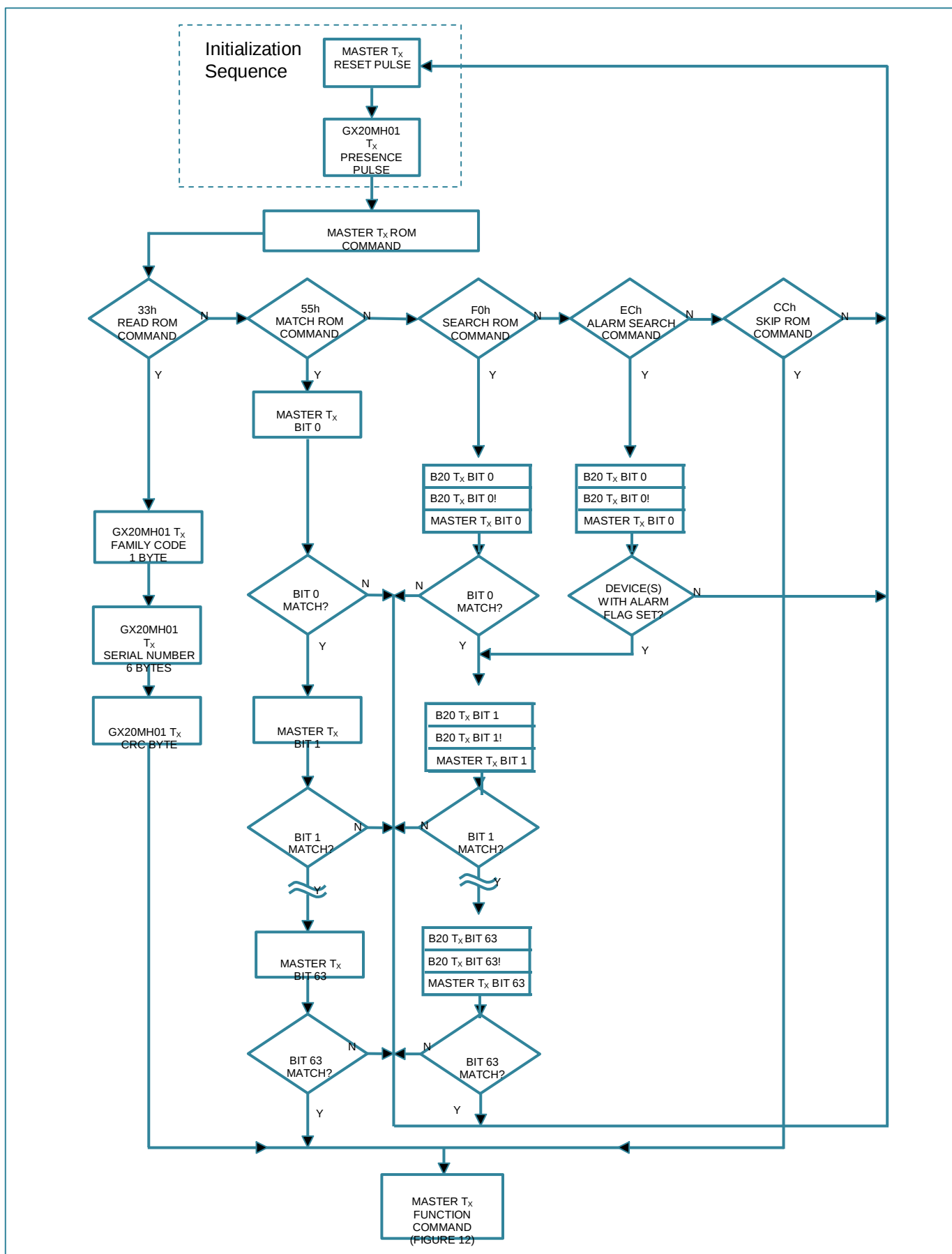
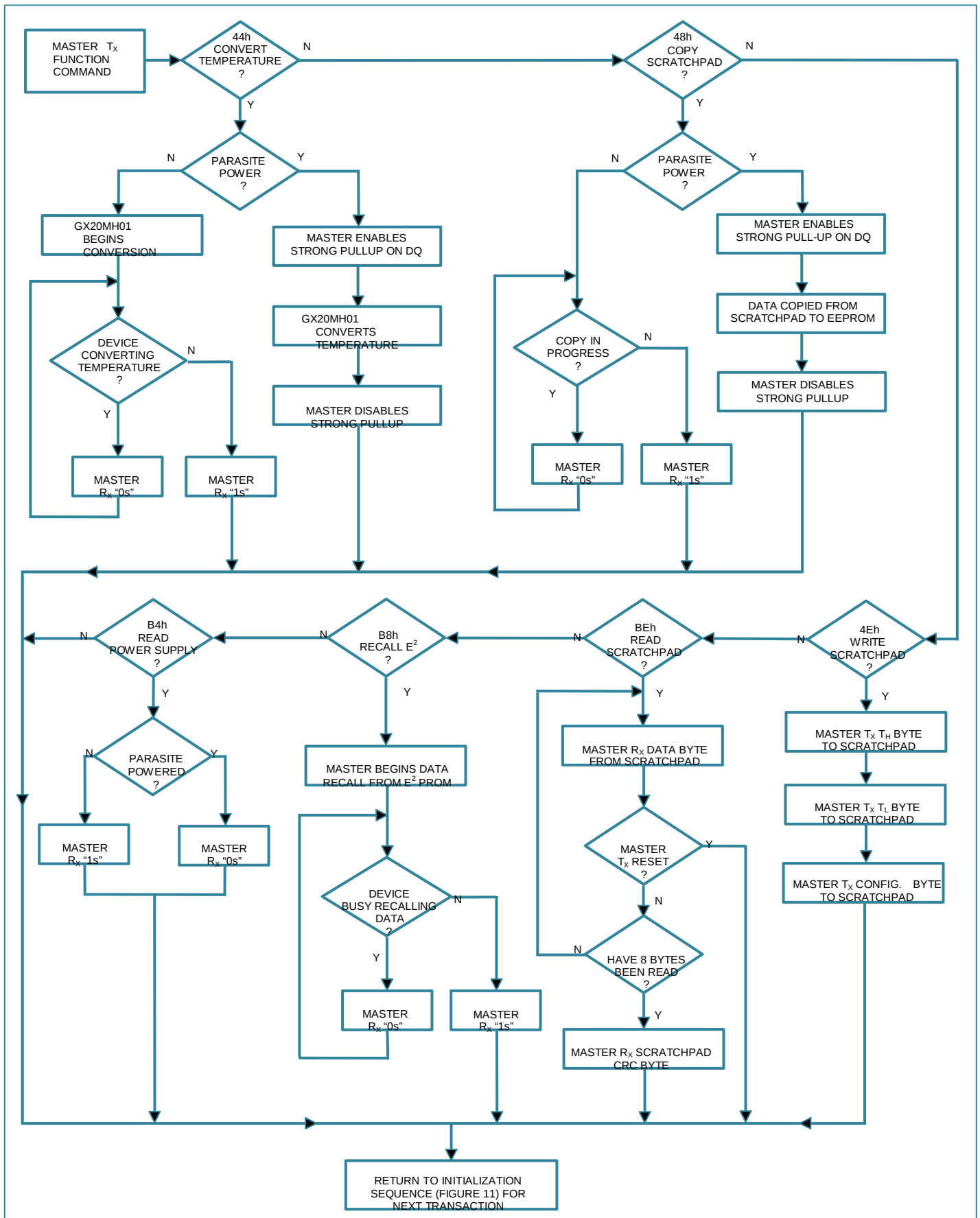


Figure 12. GX20MH01 功能指令流程图



单总线信号

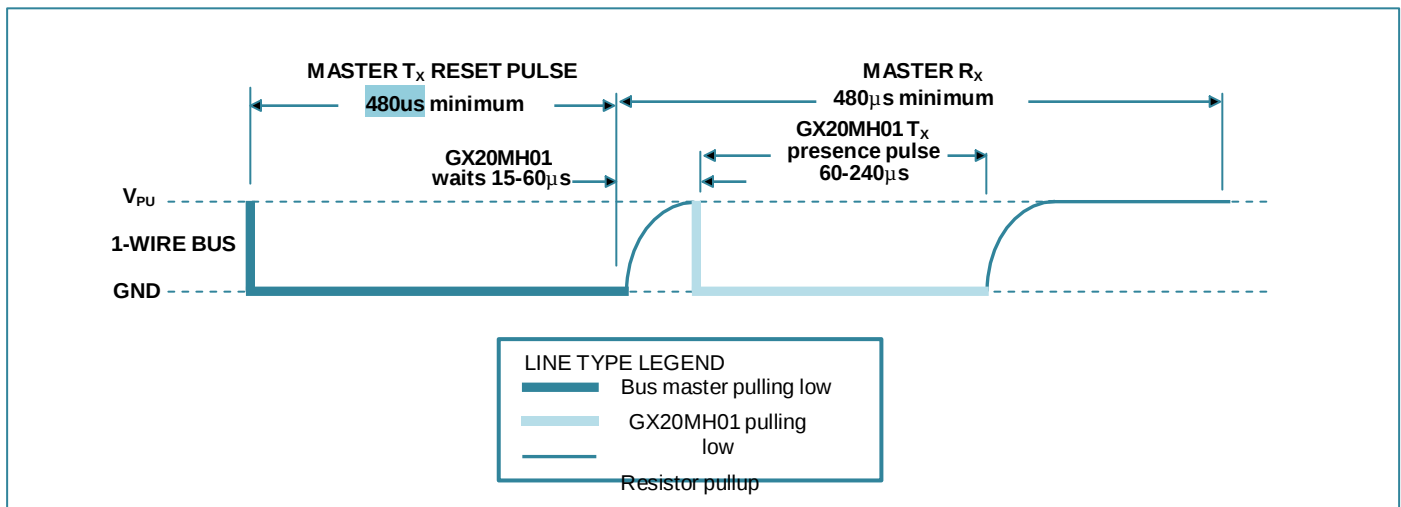
GX20MH01 需要严格的单总线协议以确保数据的完整性。协议定义了几种单总线信号的类型：复位脉冲、存在脉冲、写 0、写 1、读 0 和读 1。所有这些信号，除存在脉冲外，都是由总线控制器发出的。

复位序列：复位和存在脉冲

所有和 GX20MH01 间的通信都以初始化序列开始，初始化序列如图 13 所示。一个复位脉冲跟着一个存在脉冲表明 GX20MH01 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 480 μ s 以发出 (TX) 一个复位脉冲信号，然后释放总线，进入接收状态 (RX)。当总线被释放后，5k Ω 的上拉电阻将总线拉到高电平。当 GX20MH01 检测到 IO 引脚上的上升沿后，等待 15-60 μ s，然后发出一个由 60-240 μ s 低电平信号构成的存在脉冲。

Figure 13. 初始化时序



读/写时序

GX20MH01 的数据读写是通过时序处理来进行信息交换的，每个时序传输 1 位数据。

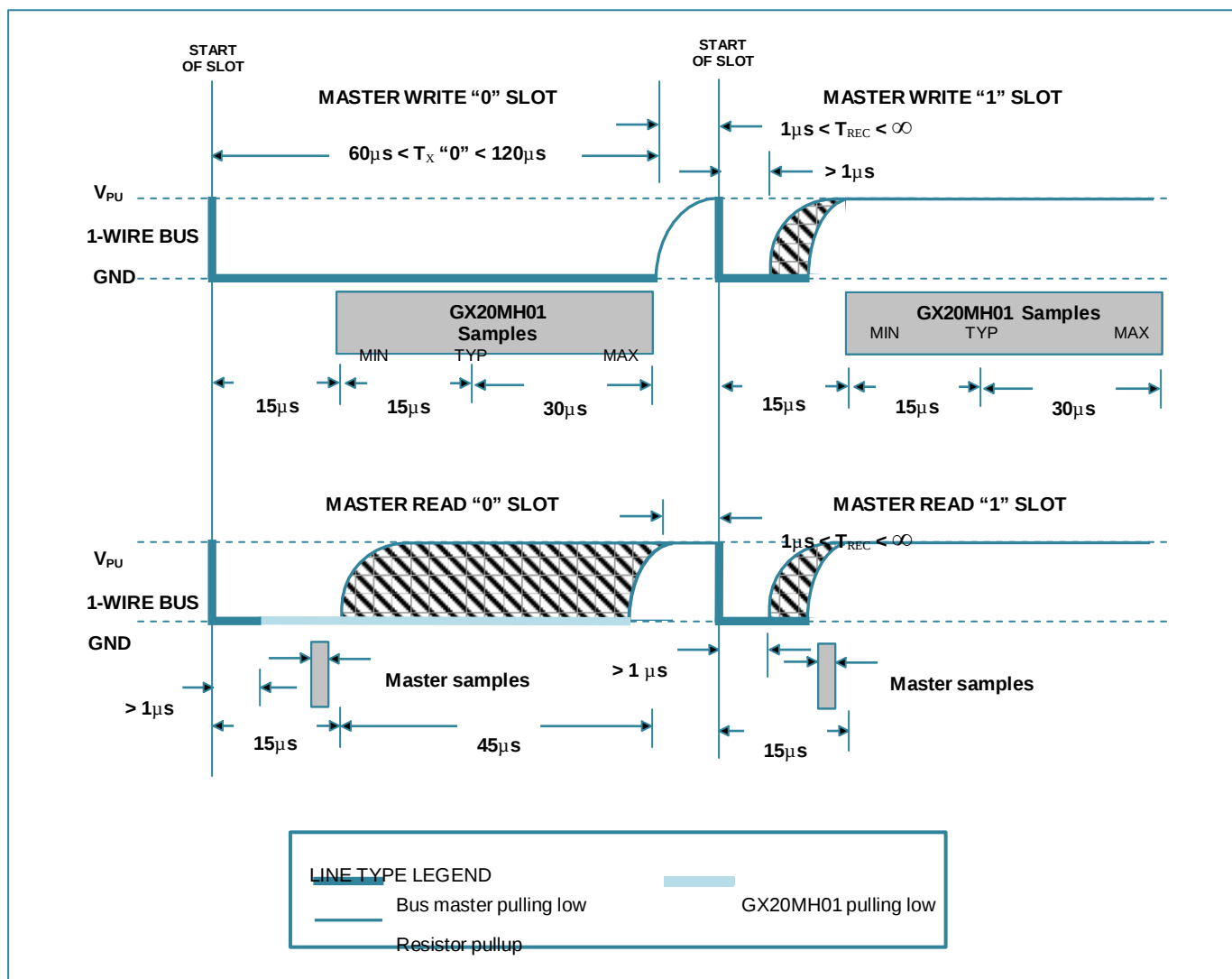
写时序

GX20MH01 有两种写时序：写 1 时序和写 0 时序。总线控制器通过写 1 时序来写逻辑 1；通过写 0 时序来写逻辑 0。写时序必须最少持续 60 μ s，包括两个写周期之间至少 1 μ s 的恢复时间。当总线控制器把数据线从逻辑高电平拉低到低电平的时候，写时序开始（见图 14）。

总线控制器要写产生一个写时序，必须把数据线拉到低电平然后释放，且需在 15 μ s 内释放总线。当总线被释放后，上拉电阻将总线拉高。总线控制器要生成写 0 时序，必须把数据线拉到低电平且继续保持至少 60 μ s。

总线控制器初始化写时序后，GX20MH01 在一个 15 μ s 到 60 μ s 的窗口内对信号线进行采样。如果线上是高电平，就是写 1。反之，如果线上是低电平，就是写 0。

Figure 14. Read/Write Time Slot Timing Diagram



读时序

总线控制器发起读时序时，GX20MH01 仅被用来传输数据给控制器。因此，总线控制器在发出读寄存器指令[BEh]或读电源模式指令[B4h]后必须立刻开始读时序，以便 GX20MH01 提供请求的数据。除此之外，总线控制器在发出发送温度转换指令平[44h]或召回 EEPROM 指令[B8h]之后读时序，详见 **GX20MH01 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。当总线控制把数据线从高电平拉低到低电平时，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，GX20MH01 通过拉高或拉低总线上来传输 1 或 0。当传输 0 结束后，总线将被释放，通过上拉电阻回到高电平空闲状态。从 GX20MH01 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 15us 内释放总线然后采样总线状态，以读取数据线的状态。

图 15 标识 T_{INIT} , T_{RC} , 和 T_{SAMPLE} 之和必须小于 $15\mu\text{s}$ 。图 16 指出, 系统时间可以通过以下方法达到最大: T_{INIT} 和 T_{RC} 保持时间尽可能短, 并且把控制器采样时间放到 $15\mu\text{s}$ 周期的最后。

图 15. 控制器读 1 的详细时序

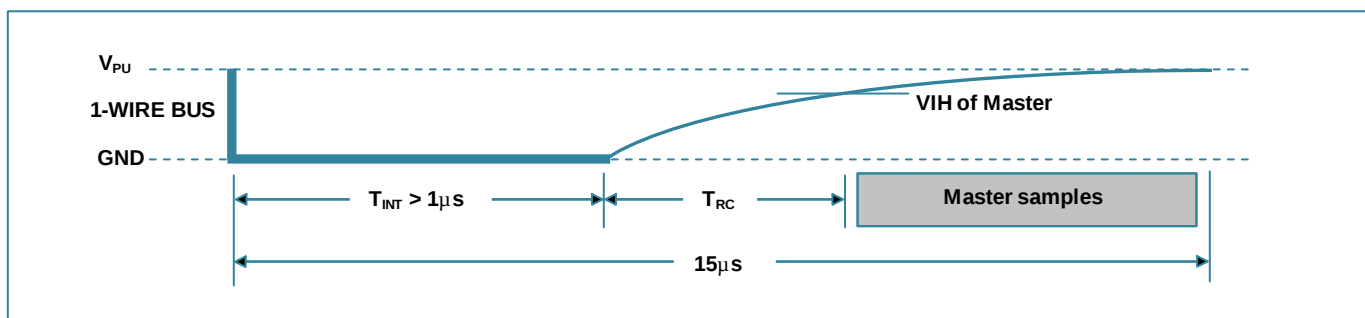
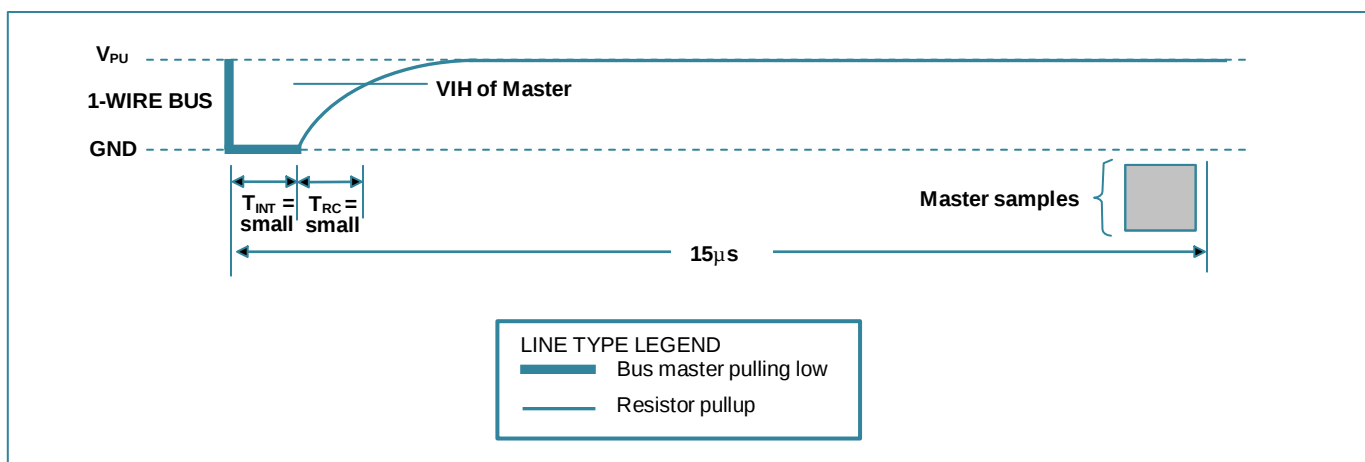


图 16. 推荐的控制度 1 时序



GX20MH01 操作举例 1

在这个例子里，总线上挂有多颗采用寄生电源供电的 GX20MH01。总线控制器启动对某个具体的 GX20MH01 进行温度转换，然后读取它的寄存器并重新计算 CRC 以确认数据。

MASTER MODE	DATA (LSB FIRST)	COMMENTS
Tx	Reset	控制器发出复位脉冲
Rx	Presence	GX20MH01s 返回存在脉冲
Tx	55h	主控制器发匹配 ROM 指令
Tx	64-bit ROM code	主控制器发 GX20MH01 地址
Tx	44h	主控制器发温度转换指令
Tx	DQ line held high by strong pullup	DQ 信号至少保持 500ms 高电平，已完成温度转换
Tx	Reset	复位脉冲
Rx	Presence	GX20MH01s 返回存在脉冲
Tx	55h	主控制器发匹配 ROM 指令
Tx	64-bit ROM code	主控制器发 GX20MH01 地址
Tx	BEh	主控制发读寄存器指令
Rx	9 data bytes	读整个寄存器加上 CRC：控制器重新计算从寄存器读到的 8 个数据字节的 CRC，把计算的 CRC 和读取的 CRC 进行比较，如果相同，控制器向下进行；如果不同，就重新操作

GX20MH01 操作举例 2

在这个例子中总线上仅有一个寄生电源供电的 GX20MH01。控制器写 T_H 、 T_L 和配置寄存器，然后读寄存器再计算 CRC 来验证数据。主控制器然后把寄存器中的数据拷贝到 EEPROM 中。

MASTER MODE	DATA (LSB FIRST)	COMMENTS
Tx	Reset	复位脉冲
Rx	Presence	GX20MH01 返回存在脉冲
Tx	CCh	跳过 ROM 指令
Tx	4Eh	写寄存器指令
Tx	3 data bytes	写 3 个数据到 T_H 、 T_L 和 配置寄存器。
Tx	Reset	复位脉冲。
Rx	Presence	GX20MH01 返回存在脉冲
Tx	CCh	跳过 ROM 指令
Tx	BEh	读寄存器指令。
Rx	9 data bytes	主控制器读包括 CRC 在内的所有寄存器：控制器重新计算从寄存器读到的 8 个字节的 CRC，把计算的 CRC 和读取的 CRC 进行比较，如果相同，控制器向下进行；如果不同，就重复读操作。
Tx	Reset	复位脉冲。
Rx	Presence	GX20MH01 返回存在脉冲
Tx	CCh	跳过 ROM 指令。
Tx	48h	拷贝寄存器指令
Tx	DQ line held high by strong pullup	控制器在执行拷贝操作时给 DQ 一个强上拉并至少保持 10ms

极限使用条件

各引脚对地的电压范围		-0.5V to +6.0V
工作温度范围		-55°C to +125°C
储存范围		-55°C to +125°C
焊接温度范围		参见 J-STD-020A 规则

以上指出器件在进行正常操作时所需要的环境条件，长期工作于极限条件下可能会影响器件的可靠性。

直流特性

(-55°C to +125°C; $V_{DD}=2.5V$ to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{DD}	Local Power	+2.5		+5.5	V	1
Pullup Supply Voltage	V_{PU}	Parasite Power	+2.5		+5.5	V	1,2
		Local Power	+2.5		V_{DD}		
Thermometer Error	t_{ERR}	-10°C to +70°C			± 0.1	°C	3
		-55°C to +125°C			± 1.0		
Input Logic-Low	V_{IL}		-0.3		+0.8	V	1,4,5
Input Logic-High	V_{IH}	Local Power	+2.2		The lower of 5.5 or $V_{DD} + 0.3$	V	1, 6
		Parasite Power	+2.5				
Sink Current	I_L	$V_{I/O} = 0.4V$	4.0			mA	1
Standby Current	I_{DDs}			750	1000	nA	7,8
Active Current	I_{DD}	$V_{DD} = 5V$		1	1.5	mA	9
DQ Input Current	I_{DQ}			5		μA	10
Drift				± 0.2		°C	11

备注:

- 1) 所有电压都是以地电位为参考电位
- 2) 上拉电压是这样得到的：假设上拉器件是完美的，因此上拉的高顶棚应该与 V_{PU} 相等。为了达到 GX20MH01 的 V_{IH} 规格，实际晶体管上拉供电必须包括电压降的极限；因此 $V_{PU_ACTUAL} = V_{PU_IDEAL} + V_{TRANSISTOR}$ 。
- 3) 典型曲线见图 17。
- 4) 逻辑 0 电平在吸收电流为 4mA 时得到。
- 5) 在寄生电源模式下低电压状态，为了保证存在脉冲 V_{ILMAX} 可能必须降低到 0.5V
- 6) 逻辑 1 电压在源电流为 1mA 时得到。
- 7) 待机电流在 70°C 时定义；125°C 时典型待机电流值为 3μA。
- 8) 为了减少 I_{DDs} , DQ 的范围如下： $GND \leq DQ \leq GND + 0.3V$ or $V_{DD} - 0.3V \leq DQ \leq V_{DD}$ 。
- 9) 动态电流涉及温度转换和写 EEPROM 存储器。
- 10) DQ 数据线为高 (“高阻”态)。
- 11) 数据的漂移是在 +125°C 电源电压 $V_{DD} = 5.5V$ 测试 1000 小时得到。

交流特性—非易失性存储器

(-55°C to +100°C; $V_{DD} = 2.5V$ to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
NV Write Cycle Time	t_{WR}			8	12	ms
EEPROM Writes	N_{EEWR}	-55°C to +55°C	1000			writes
EEPROM Data Retention	t_{EEDR}	-55°C to +55°C	10			years

交流特性

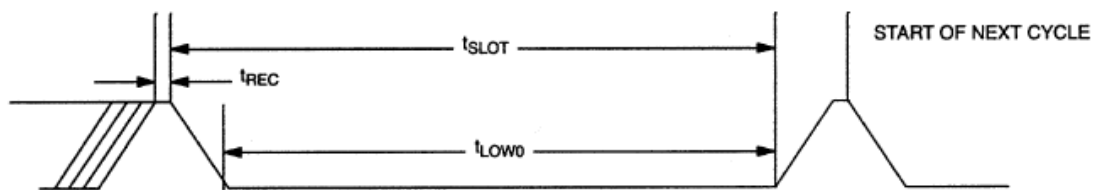
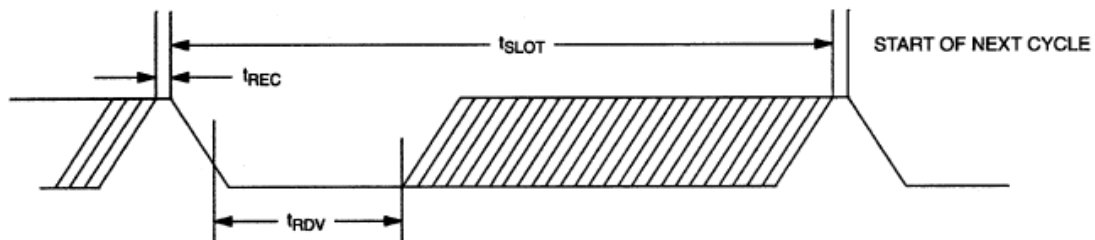
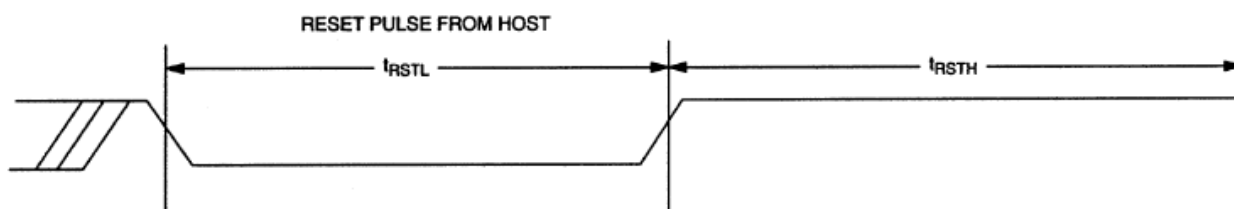
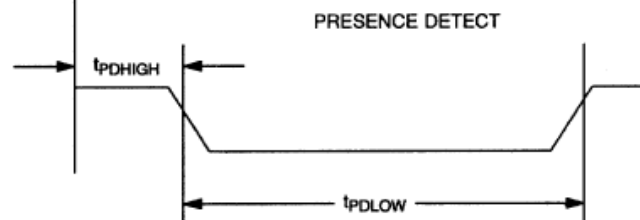
(-55°C to +125°C; $V_{DD} = 2.5V$ to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
Temperature Conversion Time	t_{CONV}	11-bit resolution		120		ms	1
		12-bit resolution		250			
		13-bit resolution		500			
		14-bit resolution		1000			
Time to Strong Pullup On	t_{SPON}	Start Convert T Command Issued		10		μs	
Time Slot	t_{SLOT}		60	120		μs	1
Recovery Time	t_{REC}		1			μs	1
Write 0 Low Time	t_{LOW0}		60	120		μs	1
Write 1 Low Time	t_{LOW1}		1	15		μs	1
Read Data Valid	t_{RDV}			15		μs	1
Reset Time High	t_{RSTH}		480			μs	1
Reset Time Low	t_{RSTL}		1			ms	1
Presence-Detect High	t_{PDHIGH}		15	60		μs	1
Presence-Detect Low	t_{PDLOW}		60	240		μs	1
Capacitance	$C_{IN/OUT}$			25		pF	

NOTES:

- 1) 关于时序见图 17.

Figure 17. Timing Diagrams

1-WIRE WRITE ZERO TIME SLOT**1-WIRE READ ZERO TIME SLOT****1-WIRE RESET PULSE****1-WIRE PRESENCE DETECT**

修改历史

REVISION DATE	DESCRIPTION
2019/4/27	First Version