

HN18B20 可编程单线式数字温度计

概述

HN18B20 数字温度传感器提供 9-12 位摄氏度温度测量数据, 可编程非易失存储器设置温度监测的上限和下限, 提供温度告警。HN18B20 通过单总线通信, 只需要一条数据线 (和地线) 即可与微处理器进行数据传输。它的测温范围为 $-55\sim+125^{\circ}\text{C}$, 在 -10°C 至 $+85^{\circ}\text{C}$ 范围内测量精度为 $\pm 0.5^{\circ}\text{C}$ 。此外, HN18B20 还可以直接利用数据线供电 (寄生电源), 无需外部电源。

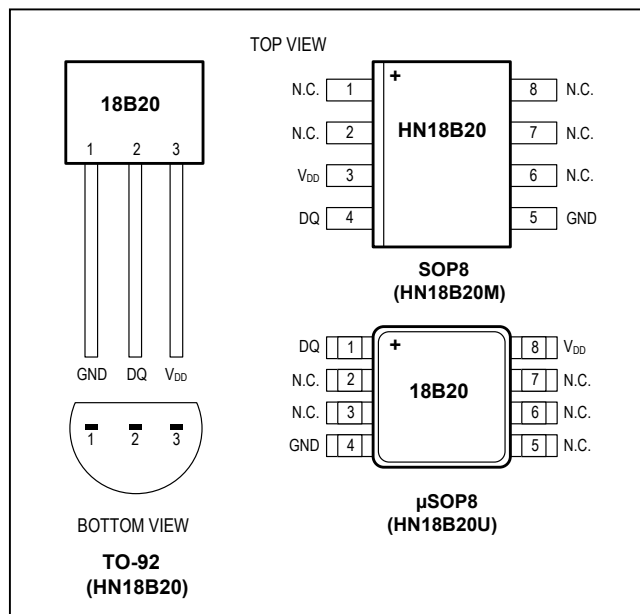
每个 HN18B20 都有一个独特的 64 位序列号, 从而允许多个 HN18B20 挂接在同一根单总线上。可以方便地采用一个微处理器控制多个分布在较大区域的 HN18B20。该功能非常适合 HVAC 环境控制、楼宇/大型设备/机器/过程监测与控制系统内部的温度测量等应用。

关键特性

- 独特的单总线接口仅需一个端口引脚进行通讯
- 每个器件具有唯一的 64 位序列号, 存储在器件 ROM 内
- 多点通信简化了分布式温度的测量
- 无需外部元件
- 可通过数据线供电, 供电电压范围 3.0-5.5V。
- 测量温度范围: -55°C 至 $+125^{\circ}\text{C}$ (-67°F 至 $+257^{\circ}\text{F}$)
- -10°C 至 $+85^{\circ}\text{C}$ 温度范围内测量精度为 $\pm 0.5^{\circ}\text{C}$
- 温度计分辨率可以设定为 9~12 位
- 最多在 750ms 内将温度转换为 12 位数字 (最大)
- 用户可设定不会丢失的温度告警值
- 告警搜索命令可识别、并标定超过用户限定温度 (温度告警条件) 的器件
- 寄生供电模式下只需对 2 个引脚(DQ 和 GND)进行控制操作
- ESD 静电保护可达 2000V
- 提供 8 引脚 SO(150 mils)、8 引脚 μSOP 、3 引脚 TO-92 封装

应用

- 温度调控的控制
- 工业系统
- 消费类产品
- 温度计
- 热敏系统



极限使用条件:

各引脚对地电压: -0.5 到+6.0V

工作温度: -55 到+125°C

储存温度: -55 到+125°C

直流电气特性 (-55°C to +125°C; $V_{DD}=3.0V$ to $5.5V$)

参数	符号	条件	最小	典型	最大	单位	备注
电源电压	V_{DD}	外部供电	3.0		5.5	V	
上拉式电源电压	V_{PU}	寄生电源	3.0		5.5	V	1, 2
	V_{PU}	外部供电	3.0		V_{DD}	V	1
温度误差	t_{ERR}	-10°C - 85°C ($V_{DD} = 4.5-5.0V$)		±0.5		°C	3
		-55°C - 125°C		±2.0		°C	
逻辑输入-低	V_{IL}	$V_{DD} = 5V$	-0.3		+0.8	V	1, 4
		$V_{DD} = 3V$	-0.3		+0.5	V	1, 2, 4
逻辑输入-高	V_{IH}	V_{DD} 脚供电	2.2		$V_{DD} + 0.3$	V	1, 5
		DQ 脚供电, ($V_{DD} = 0V$)	3.0		$V_{DD} + 0.3$	V	1, 2, 5
灌电流	I_L	$V_{IO} = 0.4V$	4.0			mA	
偏置电流	I_{DDS}	$0V \leq V_{DQ} \leq 0.4V$ 或 $V_{DD} - 0.3V \leq V_{DQ} \leq V_{DD}$		0.75	1.0	μA	
工作电流	I_{DD}	$V_{DD} = 5.0V$		0.75	1.5	mA	25°C
				0.75	1.55	mA	-55°C - 125°C
DQ 脚输入电流	I_{DQ}	$V_{DD} = 5.0V$, DQ 脚高阻态		5		μA	-55°C - 125°C
温度漂移	ΔT	$V_{DD} = 5.5V$, 125°C, 1000 小时老化测试		±0.2		°C	125°C

备注: 1. 所有电压参考点都是接地点。

2. 温度范围从-55 到+100°C

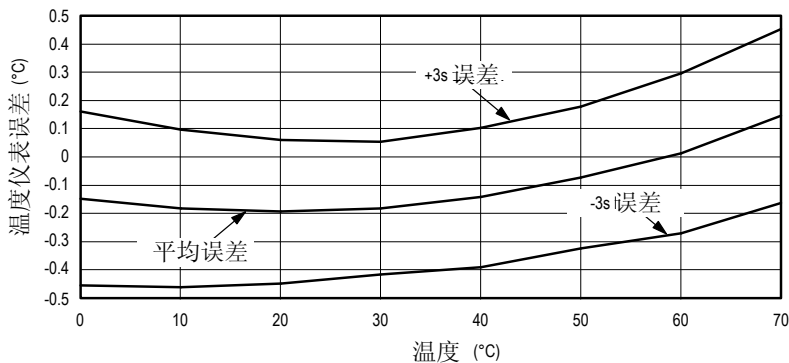
3. 见典型特性曲线图 1。

4. 逻辑-低 电压是在灌电流为 4mA 时得到。

5. 逻辑-高 电压是在源电流为 1mA 时得到。

特性曲线图 1

HN18B20A 典型误差曲线

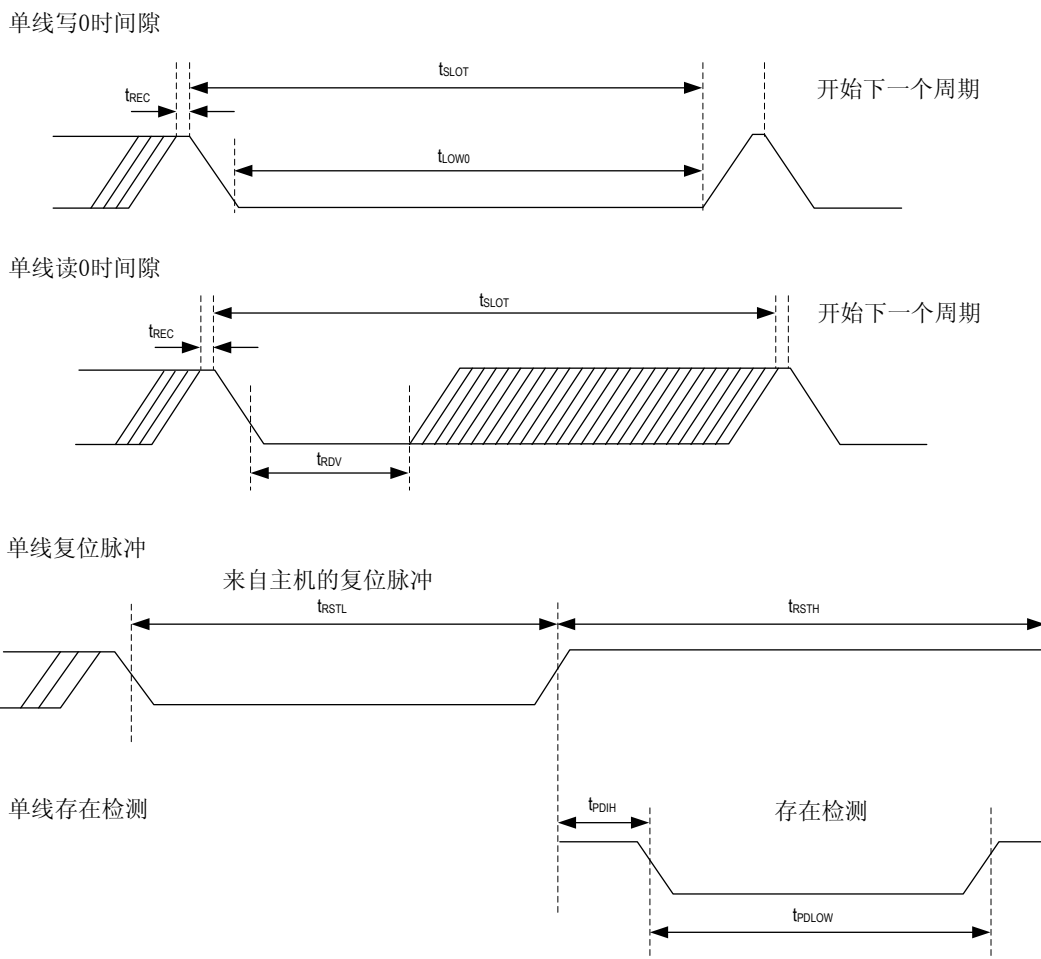


交流电气特性 (-55°C to +125°C; V_{DD}=3.0V to 5.5V)

参数	符号	条件	最小	典型	最大	单位	备注
EEPROM 写周期	t _{WR}	V _{DD} = 5.0V			10	ms	-55°C - 100°C
温度转换时间	t _{CONV}	9 位方案			93.75	ms	6
		10 位方案			187.5	ms	
		11 位方案			375	ms	
		12 位方案			750	ms	
时间隙	t _{SLOT}		60		120	μs	6
恢复时间	t _{REC}		1			μs	6
写 0 低电平时间	t _{LOW0}		60		120	μs	6
写 1 低电平时间	t _{LOW1}		1		15	μs	6
读数据有效时间	t _{RDV}				15	μs	6
复位高电平时间	t _{RSTH}		480			μs	6
复位低电平时间	t _{RSTL}		480			μs	6
存在检测高电平时间	t _{PDHIGH}		15		60	μs	6
存在检测低电平时间	t _{PDLLOW}		60		240	μs	6

备注：6. 见时序曲线图2

时序曲线图 2



订购信息

型号	工作温度范围	封装形式	标识
HN18B20	$-55^{\circ}\text{C} \leq T_A \leq 125^{\circ}\text{C}$	TO-92	18B20
HN18B20M	$-55^{\circ}\text{C} \leq T_A \leq 125^{\circ}\text{C}$	SOP8	HN18B20
HN18B20U	$-55^{\circ}\text{C} \leq T_A \leq 125^{\circ}\text{C}$	μSOP8	18B20

引脚说明

脚位			名称	功能
TO-92	SOP8	μSOP8		
—	1,2,6,7,8	2,3,5,6,7	N.C.	悬空
3	3	8	V_{DD}	可选用的 V_{DD} 引脚。具体接法见“寄生电源”节
2	4	1	DQ	数据输入/输出。单线连接时漏极开路接口引脚。当采用“寄生电源”供电方式时，还使用该脚向设备提供电源。(见“寄生电源”节)
1	5	4	GND	接地

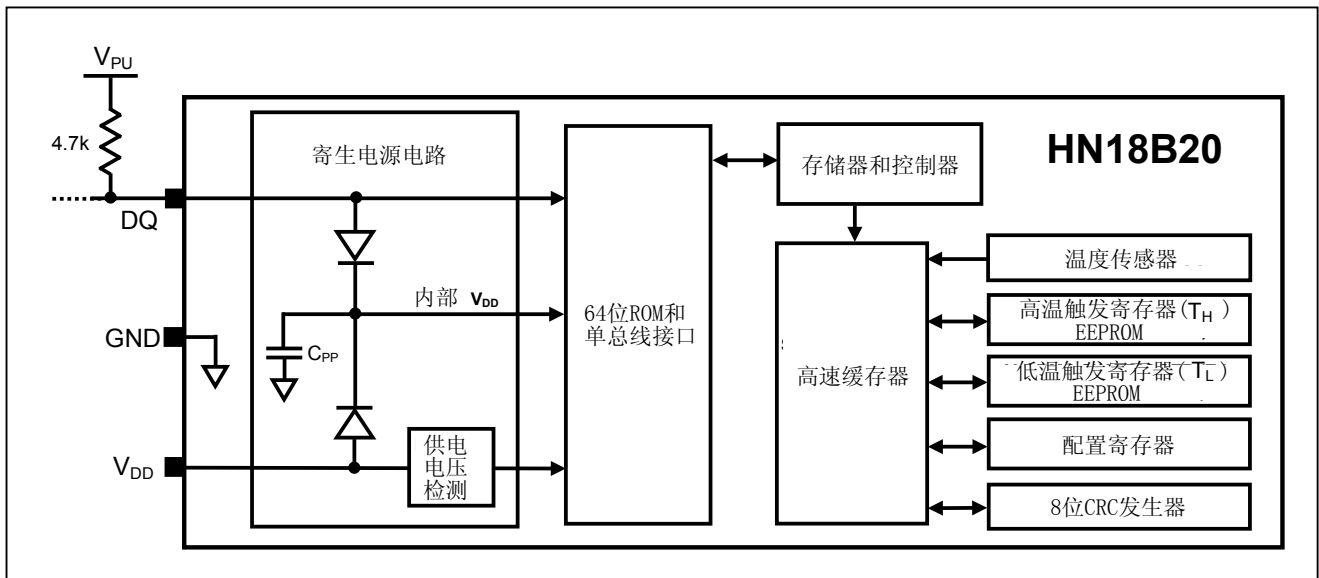
概述

图 3 是表示 HN18B20 的方框图，上述表格给出了引脚说明。64 位只读存储器(ROM)储存器件的唯一片序列号。高速缓存器含有两个字节的来自温度传感器输出的温度触发器值。除此之外，高速缓存器各提供一个字节的高温及低温告警触发器 (T_H 和 T_L)，和一个字节的配置寄存器。配置寄存器允许用户将温度的精度设定为 9, 10, 11 或 12 位。 T_H , T_L 和配置寄存器是非易失性的可擦除存储寄存器 (EEPROM)，所以其数据在器件掉电时不会丢失。

HN18B20 通过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 (HN18B20 的 DQ 引脚)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独特的片序列号，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。

HN18B20 的另一个功能是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，与上拉电阻连接的单总线通过 DQ 脚对器件供电。处于高电平状态的总线对内部电容 (C_{PP}) 充电；在总线处于低电平状态时，该电容给器件供电。这种通过总线为器件供电的方式称为“寄生电源”。HN18B20 也可以由外部电源通过 V_{DD} 引脚供电。

图 3 HN18B20 的方框图



说明-温度测量

HN18B20 的核心功能是进行温度-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°C、0.25°C、0.125°C、0.0625°C 分辨率。值得注意的是，上电默认为 12 位转换精度。HN18B20 上电后工作在低功耗闲置状态下。控制器必须向 HN18B20 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 2 字节的暂存存储器的温度寄存器中，并且 HN18B20 将会恢复到闲置状态。如果 HN18B20 是由外部供电，当发送完温度转换命令[44h]后，控制器可以执行“读数据时序”（请参阅“单总线系统”章节）。若此时温度转换正在进行，HN18B20 将会响应“0”，若温度转换完成则会响应“1”。如果 HN18B20 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“HN18B20 的供电”章节中详细讲解。

HN18B20 的温度输出数据是在摄氏度下校准的；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 4）。符号标志位（S）表示温度的正负极性：正数则 S=0，负数则 S=1。如果 HN18B20 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（位 0）为未定义的。若为 10 位转换精度，则 bit 1（位 1）和 bit 0（位 0）为未定义的。若为 9 位转换精度，则 bit 2（位 2）、bit 1（位 1）和 bit 0（位 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 4 温度寄存器格式

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
LS BYTE	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹	2 ⁻²	2 ⁻³	2 ⁻⁴
	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
MS BYTE	S	S	S	S	S	2 ⁶	2 ⁵	2 ⁴

S=测得温度正负值

表 1 温度/数据关系

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
+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

*芯片上电复位后温度寄存器默认值为 +85°C

说明-告警信号

当 HN18B20 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告警触发寄存器 T_H 和 T_L（详见图 5）中的一个字节值进行比较。符号标志位（S）表示温度的正负极性：正数则 S=0，负数则 S=1。高温和低温（T_H 和 T_L）触发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。如何读取暂存器中第 2、3 个字节中的高温和低温（T_H 和 T_L）数据在“存储器”章节中有说明。

图 5 高温和低温 (T_H 和 T_L) 触发器

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
S	2^6	2^5	2^4	2^3	2^2	2^1	2^0

因为高温和低温 (T_H 和 T_L) 触发器是一个 8 位的寄存器，所以在与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 T_L 及大于或等于 T_H ，HN18B20 内部就会设置一个告警标识。每进行一次测温就对这个标识进行一次更新；因此，如果告警条件不成立了，在下一次温度转换后告警标识将被移去。

控制器可以通过告警查询命令[ECh]查询该总线上的 HN18B20 的告警标识。任何出现告警标识的 HN18B20 都会响应该命令，所以总线控制器能精确定位每一个满足告警条件的 HN18B20。如果温度告警标识已存在，而高温和低温 (T_H 和 T_L) 的告警设置已经被改变，下一个温度转换将重新确认告警条件。

HN18B20 的供电

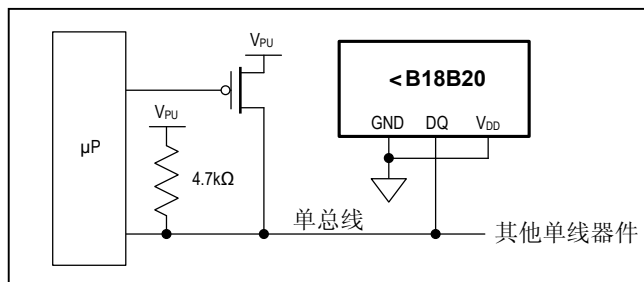
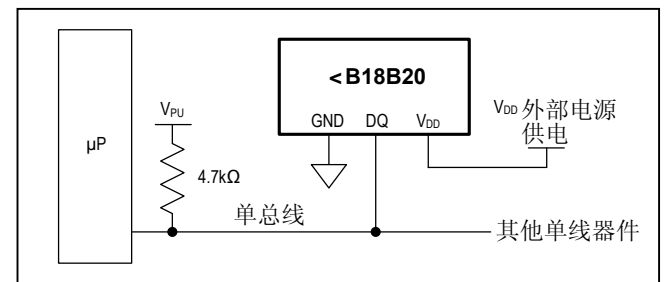
HN18B20 可以通过 V_{DD} 引脚由外部供电，或者可以由“寄生电源”供电，这使得 HN18B20 可以不采用当地的外部电源供电而实现其功能。“寄生电源”供电方式在远程温度检测或空间比较有限的地方有很大的应用。图 3 展示的就是 HN18B20 的“寄生电源”控制电路，当总线拉高电位时器件通过 DQ 脚“偷电”，并为内部“寄生电源”的电容 (C_{PP}) 充电，可用于总线低电位时为器件供电。当 HN18B20 为“寄生电源”供电模式时，该 V_{DD} 引脚必须连接到地。

在“寄生电源”供电模式下，只要工作在指定的时序下，则该单总线和 C_{PP} 可以提供给 HN18B20 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 HN18B20 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。通过单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 HN18B20 有足够的电流供应，有必要在单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 6 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。必须注意的是，单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 10 μ s（最长）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。当强制拉高时该单总线上不允许有任何其他动作。

当然，HN18B20 也可以采用常规的通过外部电源连接至 V_{DD} 引脚的供电方式，如图 7 所示。这种供电方式的优点是不需要上拉的 MOSFET，并且单总线在温度转换期间还可执行其他动作。

“寄生电源”供电方式在温度超过+100 $^{\circ}$ C 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 HN18B20 由外部电源供电。

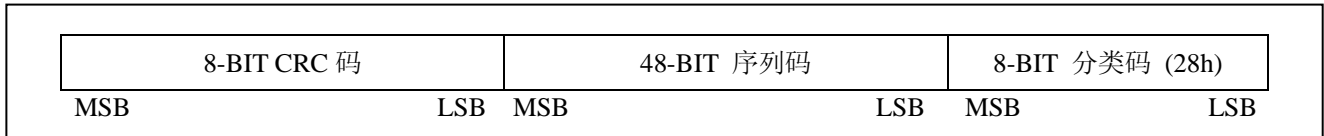
在某些情况下，总线上的控制器可能不知道连接到该总线上的 HN18B20 是由“寄生电源”供电还是由外部电源供电。此时该控制器就需要得到一些信息来决定在温度转换期间是否要强制拉高。为了得到这些信息，控制器可以在发送一个跳过 ROM 命令[CCh]之后，再发送一个读取供电模式命令[B4h]，再紧跟一个“读时段”。在读时段中，“寄生电源”供电方式的 HN18B20 会将总线拉低，但是，由外部供电方式的 HN18B20 将会让该总线继续保持高。所以，如果总线被拉低，控制器就必须要在温度转换期间将总线强制拉高。

图 6 温度转换期间的“寄生电源”供电方式

图 7 外部电源供电方式


64 位光刻 ROM 编码

每片 HN18B20 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 HN18B20 单总线的分类编码：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 位的循环冗余校验码 (CRC) 字节值，关于 CRC 的说明详见循环冗余校验码 (CRC) 发生器章节。该 64 位 ROM 编码及相关的 ROM 功能控制逻辑允许 HN18B20 作为单总线协议上的设备，更加详细的介绍见“单总线系统”章节。

图 8 64 位光刻 ROM 编码



存储器

HN18B20 的存储器构型结构如图 9 所示。该存储器包含了 SRAM 暂存寄存器和存储着高温和低温 (T_H 和 T_L) 触发器、及配置寄存器的非易失性 EEPROM。值得注意的是当 HN18B20 的温度告警功能没有用到的时候，高温和低温 (T_H 和 T_L) 触发器可以当做通用功能的存储单元。所有的存储命令在“HN18B20 功能命令”章节有详细描述。

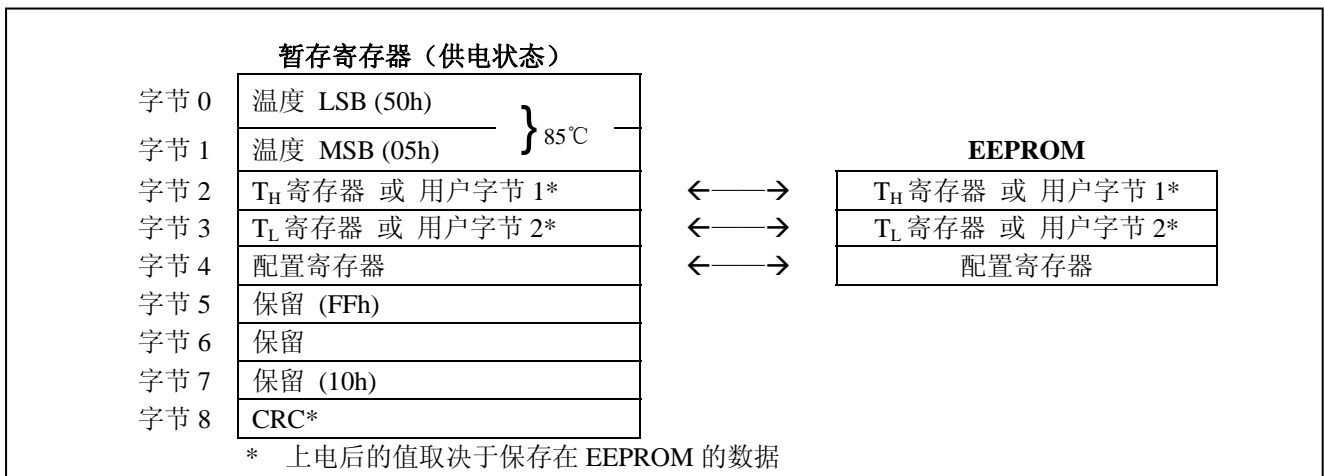
暂存寄存器中的字节 0 和字节 1 分别作为温度寄存器的低位字节 (LSB) 和高位字节 (MSB)。同时这两个字节是只读的。字节 2 和字节 3 作为高温和低温 (T_H 和 T_L) 触发器。字节 4 保存着配置寄存器的数据，详见“配置寄存器”章节。字节 5、6、7 作为内部使用的字节而保留使用，不可被写入。

暂存寄存器的字节 8 为只读字节，其中存储着该暂存寄存器中字节 0 至字节 7 的循环冗余校验码 (CRC)。HN18B20 计算该循环冗余校验码 (CRC) 的方法在“循环冗余校验码 (CRC) 发生器”章节中有详细描述。

使用写暂存寄存器命令[4Eh]才能将数据写入字节 2、3、4 中；这些写入 HN18B20 中的数据必须从字节 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，从单总线传送的数据是以字节 0 的最低位开始的。为了将暂存寄存器中的高温和低温 (T_H 和 T_L) 温度告警值及配置寄存器数据转移至 EEPROM 中，控制器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。控制器通过发送重新装载 EEPROM 命令后发出读时序，HN18B20 通过响应 0 表示重新装载正在进行、响应 1 表示重新装载已完成来反映自己的状态。

图 9 HN18B20 存储器构型图



配置寄存器

暂存寄存器中的字节 4 包含着配置寄存器；如图 10 所示。用户通过改变表 2 中 R0 和 R1 的值来配置 HN18B20 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（位 7）和 Bit 0（位 0）至 Bit 4（位 4）作为内部使用而保留使用，不可被写入。

图10 配置寄存器

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0	R1	R0	1	1	1	1	1

表 2 温度分辨率配置

R1	R0	分辨率(位数)	最大转换时间	
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})

循环冗余校验码（CRC）发生器

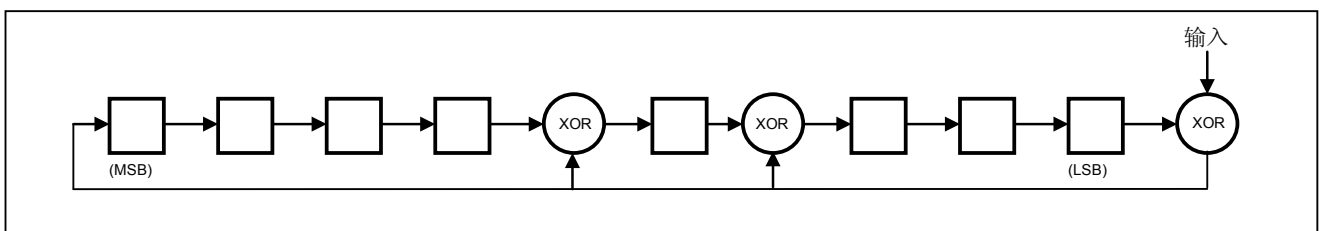
HN18B20 的 64 位 ROM 编码的一部分和暂存寄存器的第 9 字节都为循环冗余校验码(CRC)计算字节。ROM 编码的循环冗余校验码（CRC）是由 ROM 编码的低 56 位计算而来，并且该 CRC 计算值存储在 ROM 编码的最高字节。暂存寄存器的 CRC 值是由存储在暂存寄存器中的值计算而来，故当暂存寄存器中的值发生改变后，该 CRC 值也会随之发生改变。当总线上的控制器从 HN18B20 中读取数据时循环冗余校验码（CRC）给控制器提供一个数据验证码。为了验证读取到的数据是正确的，控制器必须根据读取到的数据重新进行 CRC 计算，计算得到的值再与 ROM 编码 CRC（从 ROM 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。如果控制器计算得到的 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控制器决定的。如果控制器计算的 CRC 值与从 HN18B20 中（ROM 或暂存寄存器）读取的 CRC 值不匹配，HN18B20 内部没有任何电路能够阻止从控制器发送过来的命令。

CRC 校验（ROM 或暂存寄存器）的多项式等效公式为：

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

控制器可以根据图 11 中的多项式发生器重新计算 CRC 值与从 HN18B20 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字节 0 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 位或暂存寄存器中字节 7 的最高位左移后，该多项式发生器将会保存 CRC 校验值。下一步，将从 HN18B20 中的 8 位 ROM 编码中或暂存寄存器中读取到的 CRC 校验值左移进入该发生器。这些都完成后，如果重新计算的 CRC 值为正确的，则该发生器中的所有左移寄存器都会是 0。

图 11 CRC 发生器



单总线系统

单总线系统即一个总线控制器控制一个或多个从器件。HN18B20 始终是一个从器件。当总线上只有一个从器件时，此系统被称为“单节点”系统；当总线上有多个从器件连接时，此系统被称之为“多节点”系统。

单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。

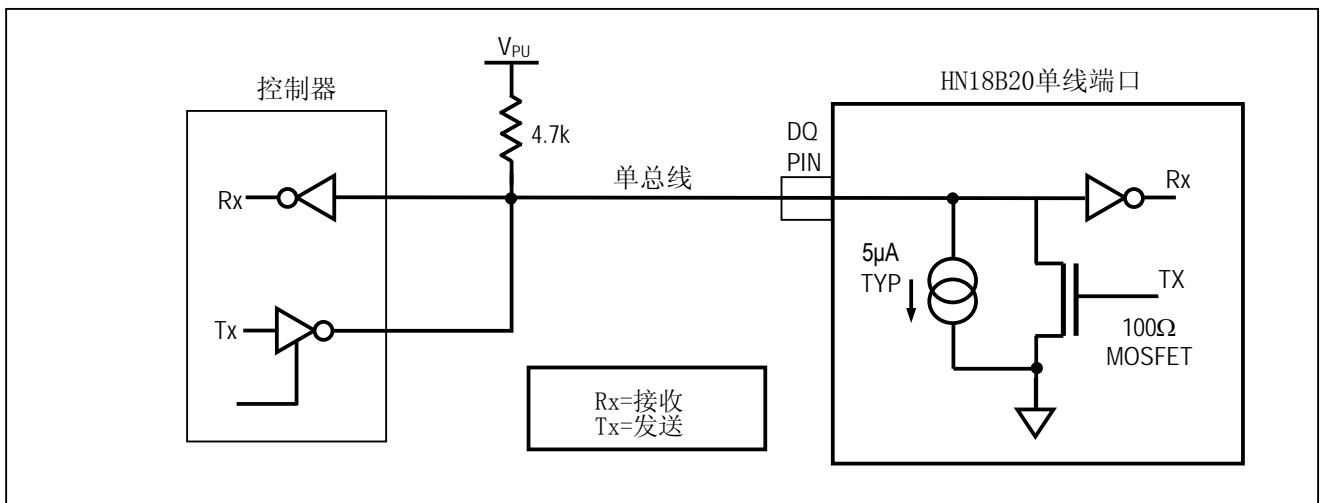
接下来关于单总线系统的描述将会分成三个部分：硬件配置，事件序列和单总线信号（信号定义和时序）。

硬件配置

单总线被定义为仅有一根数据线。每个设备（控制器或从器件）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时其他设备可以有效地使用数据线。HN18B20 的单总线接口（DQ 引脚）是其内部电路组成的漏极开路（如图 12 所示）。

单总线需要一个 $5k\Omega$ 左右的外部上拉电阻；因此，单总线在闲置情况下是高电平。无论因为任何原因一个事件需要被取消，如果该事件要重新开始则该总线必须先进入闲置状态。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。如果总线被拉低超过 $480\mu s$ ，则该总线上的所有设备都会复位。

图 12 硬件结构



处理顺序

访问 HN18B20 的处理顺序如下：

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：HN18B20 功能命令（紧跟任何数据交换请求）

每次对 HN18B20 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 HN18B20 将不会响应，除非是 ROM 搜索命令[F0h]和告警搜索命令[ECh]。当执行完这些 ROM 命令之后，控制器必须回到上述步骤中的第一步。

初始化

单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控制器发出的复位脉冲以及紧跟着从器件回应的存在脉冲构成。该回应脉冲让总线上的控制器知道在该总线上有从器件（例如 HN18B20），并且已经准备好进行操作。复位及存在脉冲时序详见“单线信号”章节。

ROM 命令

当总线上的控制器检测到了存在脉冲后，就可以执行 ROM 命令。这些命令是对每个从器件独一无二的 64 位 ROM 编码进行操作的，当总线上连接有多个从器件时，可以通过这些命令识别各个器件。这些命令同时也可以使

控制器确定该总线上有多少个什么类型的器件、或者哪一个器件有温度告警信号。总共包含有 5 种 ROM 命令，每个命令的长度都是 8 位。控制器在执行 HN18B20 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 13 所示。

➤ Search Rom 搜索 ROM [F0h]

当系统上电初始化后，控制器必须识别该总线上所有从器件的 ROM 编码，这样就可以使得控制器确定总线上从器件的类型及数量。控制器通过排除方式来了解 ROM 指令的结果，要根据需要循环地发送搜索 ROM[F0h]命令（即搜索 ROM 命令，后面紧跟着数据交换）来确定总线上所有的从器件。如果仅有一个从器件在该总线上，可用更简单的读取 ROM 命令（下一段落有详解）代替搜索 ROM 的过程。当每个搜索 ROM 周期完成后，总线控制器都必须回到处理顺序的第一步（初始化）。

➤ Read Rom 读取 ROM [33h]

该命令在总线上仅有一个从器件时才能使用。该命令使得总线上的控制器不需要搜索 ROM 命令过程就可以读取从器件的 64 位 ROM 编码。当总线上有超过一个从器件时，若再发送该命令，则当所有从器件都会回应时，将会引起数据冲突。

➤ Match Rom 匹配 ROM [55h]

该匹配 ROM 命令之后跟随发送 64 位的 ROM 编码使得总线上的控制器能够匹配特定的从器件。只有完全匹配该 64 位 ROM 编码的从器件才会响应总线上的控制器发出的功能命令；总线上的其他从器件将会等待下一个复位脉冲。

➤ Skip Rom 跳过 ROM [CCh]

控制器可以使用该命令来同时访问总线上的所有设备而不需要送出 ROM 序列码信息。例如，控制器通过发送跳过 ROM 命令后再发送温度转换[44h]命令，则总线上所有的 HN18B20 将会同时进行温度转换。

需要注意的是，当总线上仅有一个从器件时，跳过 ROM 命令后面可以紧跟读取暂存寄存器[BEh]命令。这样控制器可以读取从器件的数据而不发送 64 位 ROM 编码，以节省时间。但当总线上有多个从器件时，若在跳过 ROM 命令后再发送读取暂存寄存器命令，则所有的从器件将会同时开始传送数据而导致总线上的数据冲突。

➤ Alarm Search 告警搜索 [ECh]

该命令的操作与跳过 ROM 命令基本相同，但不同的是有告警标识出现的从器件才会响应。该命令使得控制器确定在最近一次温度转换期间是否有 HN18B20 温度告警。当每个告警搜索命令周期（即告警搜索命令再接数据交换）执行后，总线控制器都必须回到处理顺序的第一步（初始化）。详见“说明—告警信号”章节。

HN18B20 功能命令

当总线上的控制器通过 ROM 命令确定了哪个 HN18B20 能够进行通信时，控制器可以向它发送一个 HN18B20 的功能命令。这些命令使得控制器可以向 HN18B20 的暂存寄存器写入或者读出数据，开始温度转换及确定供电模式。HN18B20 的功能命令见下面的描述，汇总见表 3、图 14 是流程图。

➤ Convert T 温度转换 [44h]

该命令为启动温度转换。温度转换完后，温度转换的数据存储在暂存寄存器的 2 个字节大小的温度寄存器中，之后 HN18B20 恢复到低功耗的闲置状态。如果该器件是采用的“寄生电源”供电模式，控制器在该命令发出 10 μ s（最大）后，于温度转换期间(t_{CONV})必须强制拉高单总线（见“HN18B20 的供电”章节）。如果该设备是采用的外部供电模式，控制器在温度转换命令之后可以执行读数据时序，若 HN18B20 正在进行温度转换则会响应 0，温度转换完成则响应 1。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。

➤ Write Scratchpad 写入暂存寄存器 [4Eh]

该命令使得控制器向 HN18B20 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 T_H 寄存器（暂存寄存器的字节 2），第二个字节的数据写入 T_L 寄存器（字节 3），第三个字节的数据写入配置寄存器（字节 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控制器必须先对从器件复位，否则数据将会损坏。

➤ **Read Scratchpad 读暂存寄存器 [BEh]**

该命令使得控制器可以读取暂存寄存器中存储的值。数据从字节 0 的低位开始传送直到第 9 个字节（字节 8 - CRC）读取完毕。控制器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。

➤ **Copy Scratchpad 拷贝暂存寄存器 [48h]**

该命令为将暂存寄存器中的 T_H 、 T_L 及配置寄存器的值（字节 2，字节 3 和字节 4）拷贝至 EEPROM 中。如果该器件采用“寄生电源”供电模式，在该命令发送后的 10 μ s（最大）内控制器必须强制拉高单总线 10ms 以上，见“HN18B20 的供电”章节描述。

➤ **Recall E² 召回 E² [B8h]**

该命令将温度告警触发值（ T_H 和 T_L ）及配置寄存器的数据从 EEPROM 中召回至暂存寄存器中的字节 2、字节 3 和字节 4 中。控制器可以在召回 EEPROM 命令之后发出读数据时序，HN18B20 通过响应 0 表示召回正在进行、响应 1 表示召回已完成来反映自己的状态。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。

➤ **Read Power Supply 读供电模式[B4h]**

控制器通过发出该命令之后再执行读取数据时序来确定总线上的 HN18B20 是否存在由“寄生电源”供电的器件。在读取数据时序中，“寄生电源”供电的 HN18B20 将会拉低总线，外部电源独立供电模式的 HN18B20 则会释放总线让其保持在高电平。更多详细请参阅“HN18B20 的供电”章节

表 3 HN18B20 功能命令

命令	说明	代码	命令发出后单总线的动作	备注
温度转换命令				
Convert T 温度转换	启动温度转换	44h	HN18B20 通知控制器其温度转换状态（在“寄生电源”供电模式下，不适用）	1
存储器命令				
Read Scratchpad 读暂存寄存器	读取包括 CRC 字节的全部暂存寄存器中的数据	BEh	HN18B20 传送 9 个字节的数据到控制器	2
Write Scratchpad 写入暂存寄存器	向暂存寄存器写入字节 2, 3 和 4 的数据（ T_H 和 T_L 和配置寄存器）	4Eh	控制器向 HN18B20 传输 3 个字节的数据	3
Copy Scratchpad 拷贝暂存寄存器	将暂存寄存器中的 T_H 、 T_L 及配置寄存器的数据拷贝至 EEPROM 中	48h	无	1
Recall E ² 召回 EEPROM	将 T_H 、 T_L 及配置寄存器的数据从 EEPROM 中召回至暂存寄存器中	B8h	HN18B20 通知控制器其召回状态	
Read Power Supply 读供电模式	向控制器标定 HN18B20 的供电模式	B4h	HN18B20 通知控制器其供电模式	

备注：1：如果是采用“寄生电源”供电模式，控制器在温度转换期间必须强制拉高单数据总线，将暂存寄存器中的数据拷贝至 EEPROM 中。这时不允许总线上的其他操作。

2：控制器可以在任何时候通过发出复位命令来中断数据的传输。

3：所有的 3 个字节在复位命令发出前都必须写入数据。

图 13 ROM 命令流程图

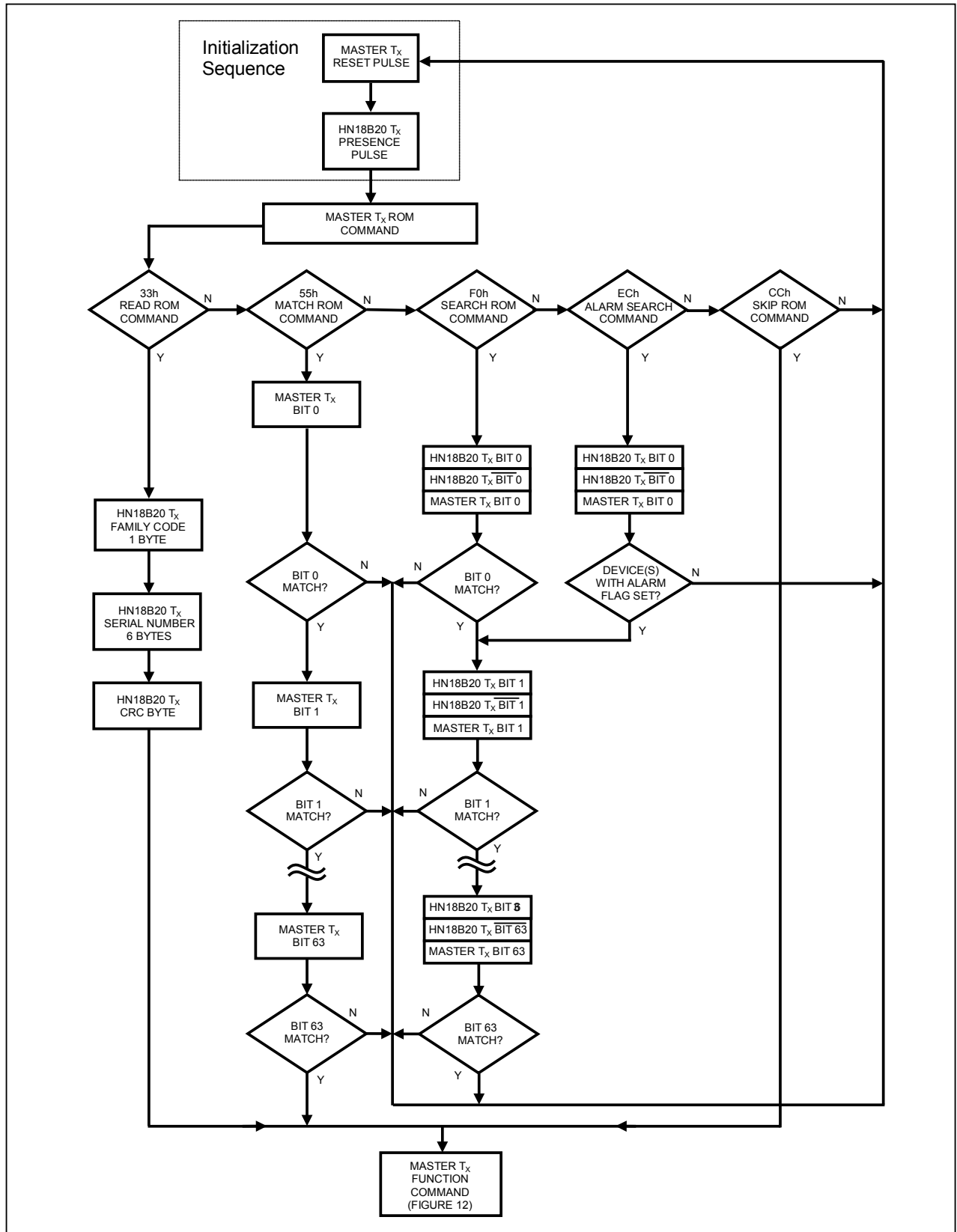
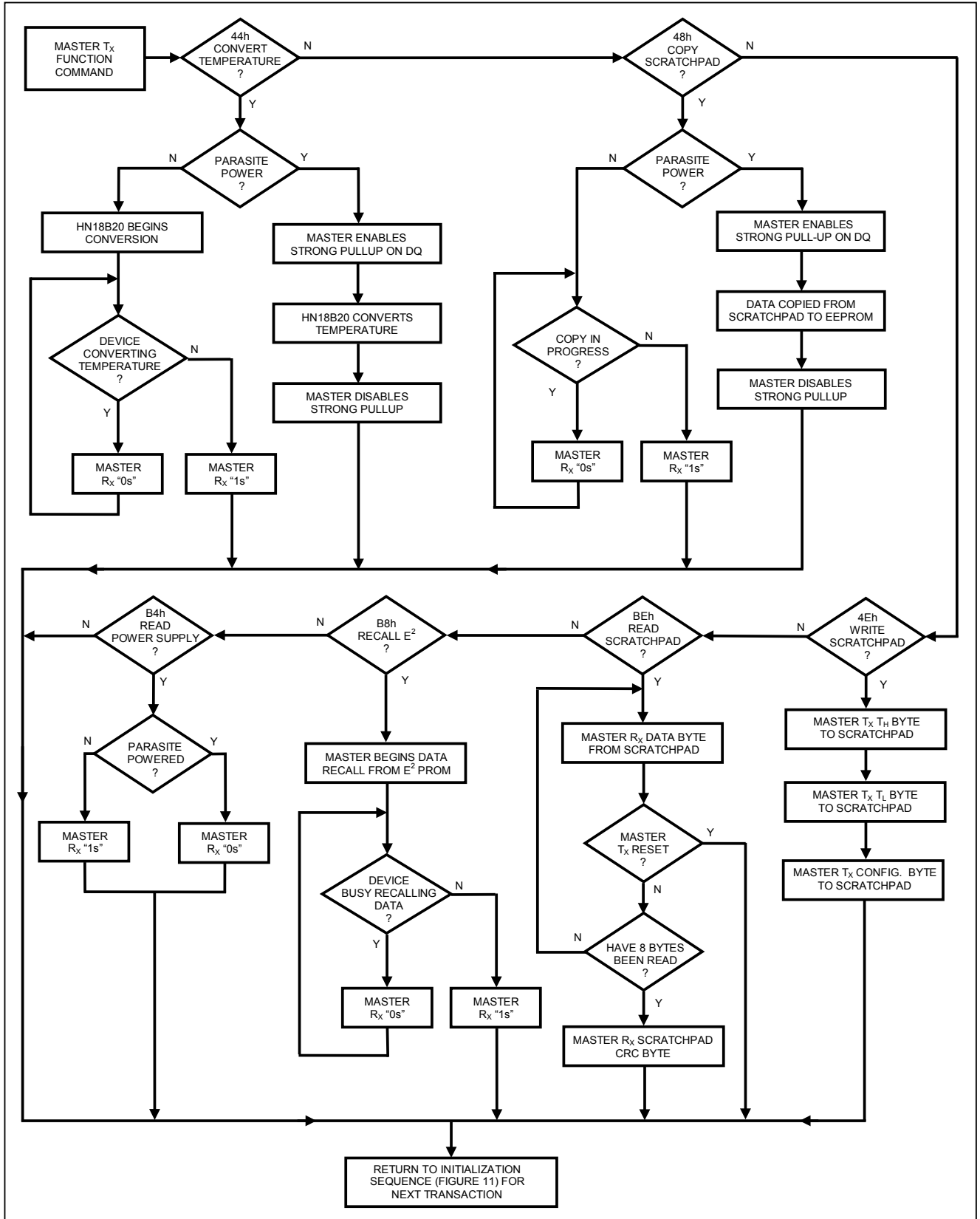


图 14 HN18B20 功能命令流程图



单线信号

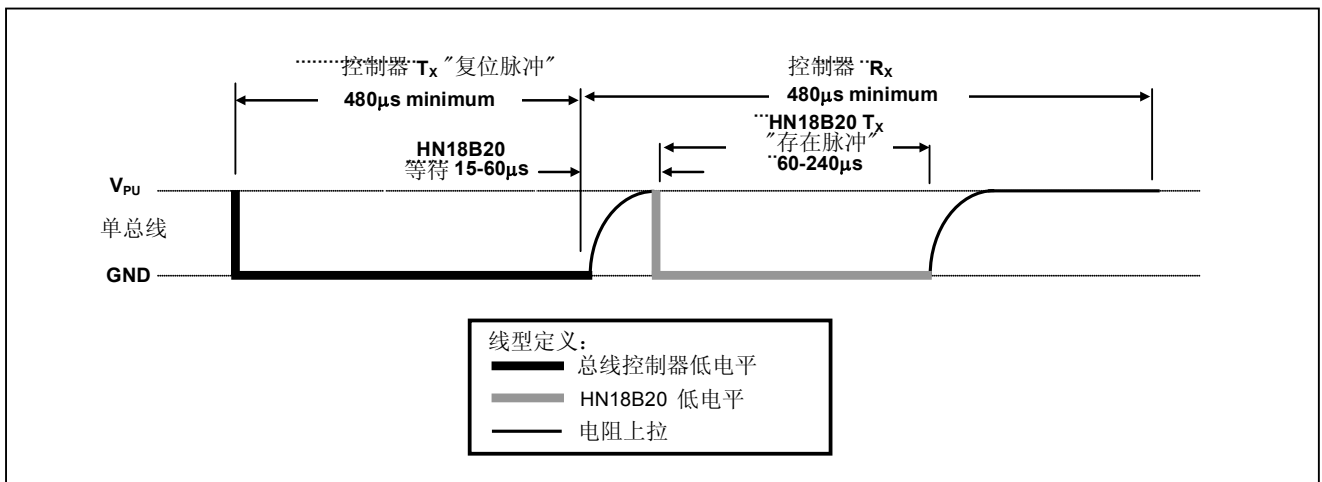
HN18B20 采用严谨的单线通信协议来保证数据的完整性。该协议定义多个信号形式：复位脉冲，存在脉冲，写 0，写 1，读 0，读 1。控制器可以发出除了存在脉冲外的所有其他信号。

初始化程序—复位和存在脉冲

与 HN18B20 所有的通信都是由初始化序列开始的，该序列包括从控制器发出的复位脉冲及从 HN18B20 响应的存在脉冲组成。如图 15 所示。当 HN18B20 响应复位信号的存在脉冲后，则其向控制器表明其在该总线上，并且已经做好操作命令。

在初始化序列期间，总线上的控制器通过拉低单总线超过 $480\mu\text{s}$ 来发送 (T_x) 复位脉冲。之后控制器释放总线而进入接收模式 (R_x)。当总线释放后， $5\text{k}\Omega$ 左右的上拉电阻将单总线拉至高电平。当 HN18B20 检测到该上升沿信号后，其等待 $15\mu\text{s}$ 至 $60\mu\text{s}$ 后通过将单总线拉低 $60\mu\text{s}$ 至 $240\mu\text{s}$ 来实现发送一个存在脉冲。

图 15 初始化时序



读/写时段

控制器通过写时段向 HN18B20 中写入数据，通过读时段从 HN18B20 中读取数据。单总线上每一个读写时段只能传送一个位的数据。

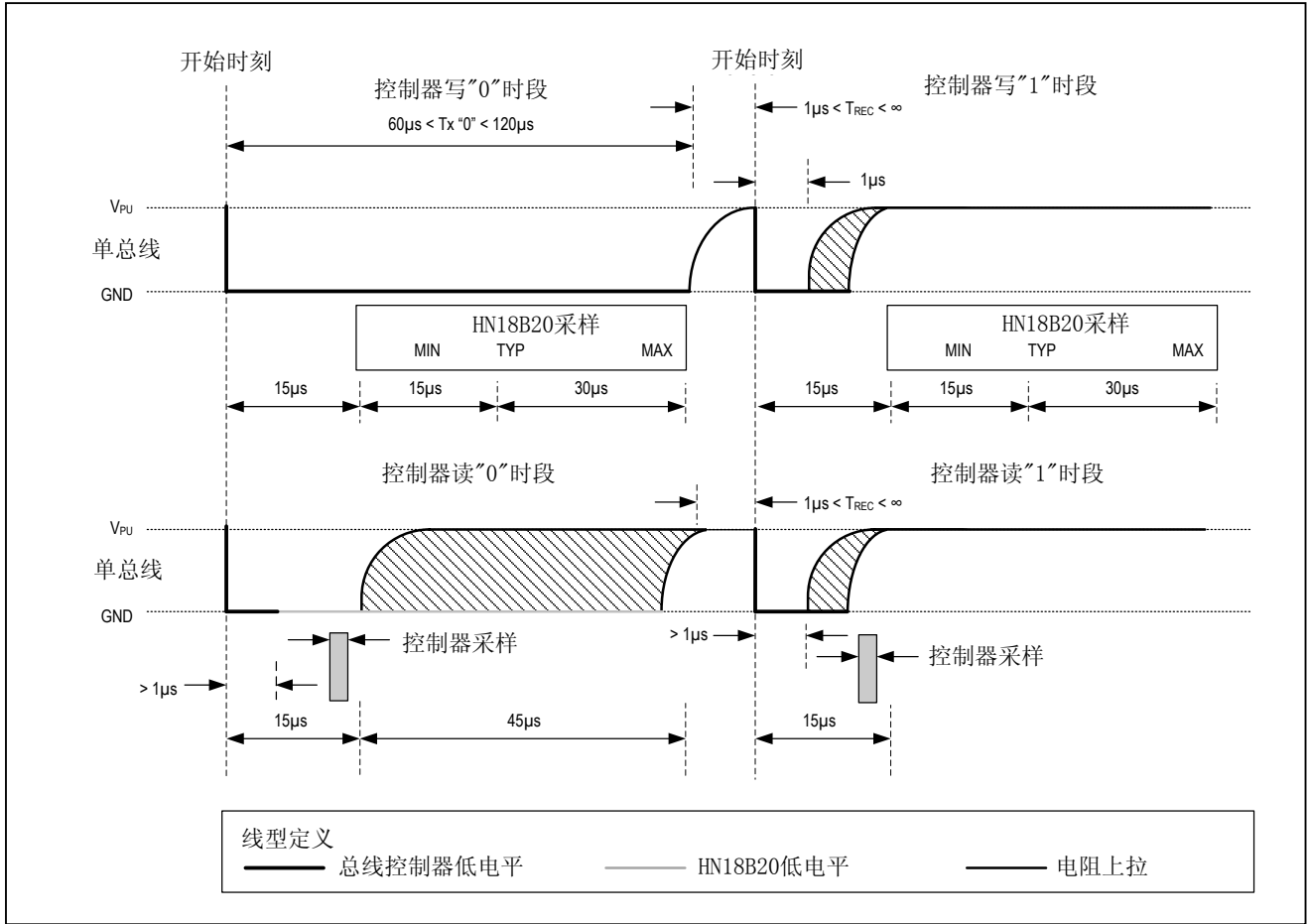
写时段

写时段有两种情况：“写 1”时段和“写 0”时段。控制器通过写 1 时段来向 HN18B20 中写入逻辑 1 以及通过写 0 时段来向 HN18B20 中写入逻辑 0。每个写时段最小必须有 $60\mu\text{s}$ 的持续时间且独立的写时段间至少有 $1\mu\text{s}$ 的恢复时间。两个写时段都是由控制器通过将单总线拉低来进行初始化（详见图 16）。

为了形成写 1 时段，在将单总线拉低后，控制器必须在 $15\mu\text{s}$ 之内释放总线。当总线释放后， $5\text{k}\Omega$ 的上拉电阻将单总线拉至高电位。为了形成写 0 时段，控制器将单总线拉低后，在整个时段必须一直拉低总线（至少 $60\mu\text{s}$ ）。

在控制器初始化写时段后，HN18B20 将会在 $15\mu\text{s}$ 至 $60\mu\text{s}$ 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 HN18B20；若总线是低电平，则逻辑 0 被写入 HN18B20。

图 16 读/写时段时序图



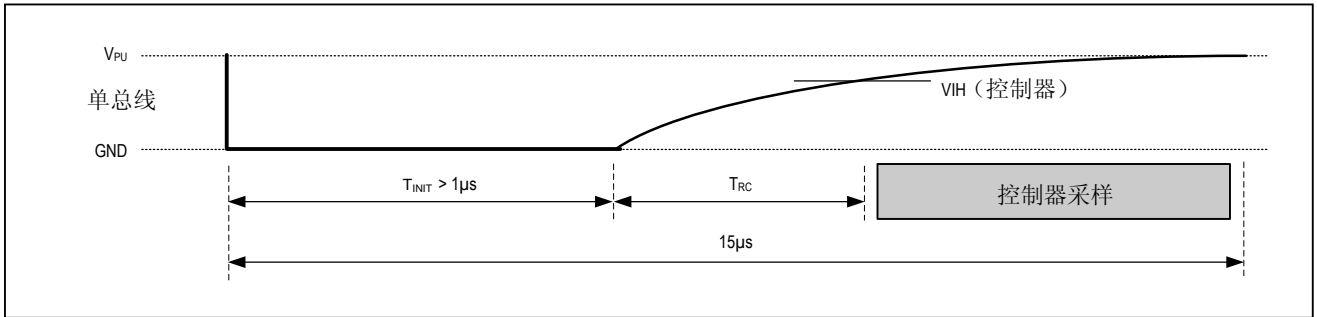
读时段

仅在读时段 HN18B20 才能向控制器传送数据。因此，控制器在发出读暂存寄存器[B4h]或读取供电模式[B4h]命令后，必须及时地生成读时段，这样 HN18B20 才能提供所需的数据。此外，控制器可以在发出温度转换 [44h]或召回 E² [B8h]命令后生成读时段，以便获得“HN18B20 功能命令”章节中提到的操作信息。

每个读时段最小必须有 60μs 的持续时间，各时段之间至少有 1μs 的恢复时间。读时段通过控制器将总线拉低超过 1μs 再释放总线来实现初始化（见图 16）。当控制器开始读时段后，HN18B20 将会向总线发送 0 或者 1。HN18B20 通过释放总线至高电位来发送逻辑 1，将总线拉至低电位来发送逻辑 0。当发送完 0，该时段结束后，HN18B20 将会释放总线，通过上拉电阻将总线恢复到高电平的闲置状态。在下降沿启动读时段后从 HN18B20 输出的数据仅有 15μs 的有效时间。因此，控制器必须在读时段后的 15μs 之内释放总线，并且完成对总线采样。

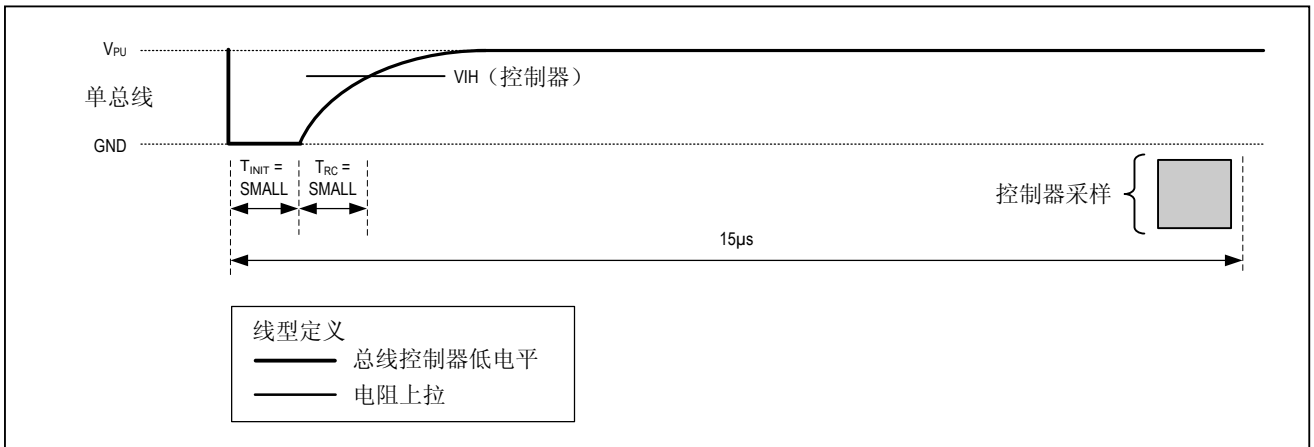
读时段中， T_{INIT} 、 T_{RC} 及 T_{SAMPLE} 之和必须小于 $15\mu s$ ，见图 17。

图 17 控制器读 1 的时序过程



如图 18 所示，系统时序的最大极限是，让 T_{INIT} 和 T_{RC} 的时长尽可能短，控制器在读时段 $15\mu s$ 结束前完成采样。

图 18 建议控制器读 1 的时序



HN18B20 操作实例 1

在该例中，有多个 HN18B20 挂在总线上，同时它们都采用的“寄生电源”供电。控制器启动其中的一个 HN18B20 的温度转换，再读取它的暂存器中的数据，进行 CRC 校验来验证数据。

控制器状态	数据(LSB 在前)	内容
Tx	复位	控制器发出复位脉冲
Rx	存在	HN18B20 响应存在脉冲
Tx	55h	控制器发出匹配 ROM 指令
Tx	64 位 ROM 编码	控制器发出 HN18B20 ROM 编码
Tx	44h	控制器发温度转换指令
Tx	DQ 数据线强上拉	控制器对 DQ 引脚强上拉高电平，以保证温度转换的完成(t_{CONV})
Tx	复位	控制器发复位脉冲
Rx	存在	HN18B20 响应存在脉冲
Tx	55h	控制器发出匹配 ROM 指令
Tx	64 位 ROM 编码	控制器发出 HN18B20 ROM 编码
Tx	BEh	控制器发读暂存器指令
Rx	9 个数据字节	控制器读含有 CRC 的整个暂存器，并重新计算从暂存器读到的前 8 个数据字节的 CRC 值，把计算的 CRC 和读取的 CRC（字节 9）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。

HN18B20 操作实例 2

在该实例中，仅有一个 HN18B20 连接到总线上，并且其采用的“寄生电源”供电。控制器向 HN18B20 的暂存寄存器 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。最后控制器将暂存寄存器中的值拷贝至 EEPROM 中。

控制器状态	数据(LSB 在前)	内容
Tx	复位	控制器发出复位脉冲
Rx	存在	HN18B20 响应存在脉冲
Tx	CCh	控制器发跳过 ROM 指令
Tx	4Eh	控制器发出写入暂存寄存器指令
Tx	3 个数据字节	控制器发出 3 个数据字节到暂存寄存器 (TH, TL, 和配置寄存器)
Tx	复位	控制器发出复位脉冲
Rx	存在	HN18B20 响应存在脉冲
Tx	CCh	控制器发跳过 ROM 指令
Tx	BEh	控制器发读暂存器指令
Rx	9 个数据字节	控制器读含有 CRC 的整个暂存器，并重新计算从暂存器读到的前 8 个数据字节的 CRC 值，把计算的 CRC 和读取的 CRC（字节 9）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	复位	控制器发出复位脉冲
Rx	存在	HN18B20 响应存在脉冲
Tx	CCh	控制器发跳过 ROM 指令
Tx	48h	控制器发拷贝暂存寄存器指令
Tx	DQ 数据线强上拉	控制器在执行拷贝操作时给 DQ 引脚一个强上拉电平，并至少保持 10ms