

3.1 LTM8000 系列模块及仪表与上位机通讯技术说明

3.1.1 概述

LTM-8000 系列模块及仪表与上位机通讯采用 RS485 通讯方式。为了避免多个设备在同一时间发送数据而导致竞争，所有的 RS-485 网的通讯由上位机控制。采用上位机主导的命令/响应方式。模块自身不断的巡检温度。

模块不发送数据时处于监听模式。主机对一个模块发出一条带地址的指令，然后等待一段时间等模块响应，如果模块在一定时间内不响应，主机将判定为“超时”。所以，在 LTM-8000 系列模块应用软件的终端仿真模式下如果发出了非法指令，要等一段时间后返回错误提示。

3.1.2 通讯特性

通讯距离： 1200 m

最大网络节点容量： 128

通讯格式： 10 位（1-起始位 8-数据位 1-停止位），无校验
波特率：

波特率代码	波特率 (bps)
06h	9600
07h	19.2k
08h	38.4k

3.1.3 语法

[命令符][地址][命令][数据][回车]。

所有的命令以命令符开始。共有七个合法命令符：\$、#、%、@、&、/、*。

命令符后面跟着两个字符地址用于区别目标模块。两个字符的命令紧接在地址后面。不同的命令后面会有不同的数据段。所有命令以回车（CR）结尾。

注：所有命令均为大写字符！

3.1.4 模块命令速查表

命令语法	命令描述	页码
%AANNTCCFF	设定配置参数	19-20
\$AA2	返回指定地址模块的配置参数	21-22
\$AAF	返回指定地址模块的软件版本号	22-23
\$AAM	返回指定地址模块的名称	23-24
\$AA6	返回指定地址模块所有通道的状态	24-25
#AA8	返回所有通道传感器及前端测控单元的数据	25-27
#AAN	返回指定模块指定通道传感器及前端测控单元的数据	27-29
&AA8	返回指定地址模块的所有 ID 号码	29-31
&AAN	返回指定地址模块指定通道的所有 ID 号码	31-32
&AA9	重新刷新指定地址模块中传感器的 ID 号码	32-33

*AAN	返回指定地址模块指定通道的所有通道传感器及前端测控单元的编号	33-35
\$AAE	读回地址为 AA 的的错误代码	35-37
/AAN DH DL OH OL (cr)	向指定通道、指定地址的 ITU 输出数据	37-38
\$AAT	读取各通道传感器类型命令	38-39
@ AA CH ID1 ID2 FUNC DN DL OD	ITU 单元设定命令	39-41

注：

1. 以上命令均为 ASCII 字符。
2. 每条命令均以回车符(cr,ODh)为结束符，有些指令在结束符后面有校验和。。
3. 所有 LTM8000 模块均支持以上命令(特殊说明除外)。
4. LTM8600 指令单独说明

3.1.5 模块命令解释

%AANNTCCFF

名称	参数配置
描述	设定模块的地址、波特率等信息 建议：此项命令用于设定模块的一些重要信息，一般不用使用，如需使用应慎重。
语法	%AANNTCCFF (cr)
	% 是命令符
	AA (范围 00h~FFh) 代表被配置模块的两字符 ASCII 码表示的 16 进制地址
	NN 代表模块将被配置的新地址，范围 00h~FFh
	TT 代表模块类型（现保留，值为 80h）
	CC 代表波特率代码，参见下表。
	FF 现在值固定为 02h，代表数据传送格式为 16 进制数、无校验。
(cr) 为结束符 (0Dh)	
响应	! AA (cr) 模块收到命令有效
	? AA (cr) 命令中有无效参数或命令格式有误
	遇到不存在的地址，将没有响应。
	! 命令符表示模块收到有效命令
	? 命令符表示模块收到无效命令
	AA 为要访问的模块的两字符 ASCII 码表示的 16 进制地址（范围 00h~FFh）
	(cr) 为结束符 (0Dh)
示例	命令 %0109800602(cr)
	响应 ! 09 (cr)

	模块地址为 01h，波特率为 9600 的 LTM-8002 模块的地址变为 09h，! 09(0) 响应指示命令收到，且已执行完毕。
--	---

波特率代码表

波特率代码	波特率 (BPS)
06h	9600
07h	19.2k
08h	38.4k

\$AA2

名称	读配置状态
描述	要求地址为 AA 的模块返回其配置数据
语法	\$AA2 (cr)
	\$ 是命令符
	AA (范围 00h~FFh) 代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址
	2 是读取配置状态命令
	(cr) 为结束符 (0Dh)
响应	! AATCCFF (cr) 模块收到命令有效
	? AA (cr) 命令中有无效参数
	遇到不存在的地址，将没有响应。
	! 命令符表示收到有效命令
	? 命令符表示收到无效命令
	AA: 要访问的模块两字符 ASCII 码表示的 16 进制地址 (范围 00h~FFh)
	TT 代表模块类型 (现保留，值为 80h)

	CC 代表波特率代码
	FF 现在值固定为 02h, 代表数据传送格式为 16 进制数、无校验。
	(cr) 为结束符 (0Dh)
示例	命令 \$012(cr)
	响应 ! 01800602 (cr)
	! -收到有效命令; 01-模块地址 01 (H); 80-模块类型 80h (保留, 不可更改); 06-波特率代码 06 (9600); 02-数据传送格式为 16 进制、无校验。

\$AAF

名称	读模块版本号
描述	要求地址为 AA 的模块返回其软件版本号
语法	\$AAF (cr)
	\$ 是命令符
	AA (范围 00h~FFh) 代表被呼叫模块的两字符 ASC II 码表示的 16 进制地址
	F 是返回软件版本号命令
	(cr) 为结束符 (0Dh)
响应	! AA (版本号) (cr) 命令有效
	? AA (cr) 命令无效
	遇到不存在的地址, 将没有响应。
	! 命令符表示收到有效命令
	? 命令符表示收到无效命令
	AA 为模块两字符 ASC II 码表示的 16 进制地址 (范围 00h~FFh)

	(版本号) 是地址为 AA 的模块的软件版本号
	(cr) 为结束符 (0Dh)
示例	命令 \$02F(cr)
	响应 !02V1.60(cr)
	02—地址为 02h
	V1.60—版本为 1.60 版

\$AAM

名称	读模块名称
描述	要求地址为 AA 的模块返回其模块名称
语法	\$AAM (cr)
	\$ 是命令符
	AA (范围 00h~FFh) 代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址
	M 是返回模块名称命令
	(cr) 为结束符 (0Dh)
响应	! AA (模块名称) (cr) 命令有效
	? AA (cr) 命令无效
	遇到不存在的地址, 将没有响应。
	! 命令符表示收到有效命令
	? 命令符表示收到无效命令
	AA 响应模块的两字符 ASCII 码表示的 16 进制地址(范围 00h~FFh)
	(模块名称) 是地址为 AA 的模块的名称, 比如: LTM8002
	(cr) 为结束符 (0Dh)

示例	命令 \$11M(cr)
	响应 !11LTM8002(cr)
	11—地址为 11h
	LTM8002—模块名称

\$AA6

名称	读模块通道状态
描述	要求地址为 AA 的模块返回其通道状态
语法	\$AA6 (cr)
	\$ 是命令符
	AA (范围 00h~FFh) 代表被呼叫模块的两字符 ASC II 码表示的 16 进制地址
	6 读通道状态命令
	(cr) 为结束符 (0Dh)
响应	! AAVV (0011223344556677) (cr) 命令有效
	? AA (cr) 命令无效
	遇到不存在的地址，将没有响应。
	! 命令符表示收到有效命令
	? 命令符表示收到无效命令
	AA 响应模块的两字符 ASC II 码表示的 16 进制地址(范围 00h~FFh)
	VV 是地址为 AA 的模块的通道状态，两字符组成一个字节(格式参见下表)
	0011223344556677 共 16 个字符，两个一组，以十六进制数形式表示每个通道传感器数量

	(cr) 为结束符 (0Dh)
示例	命令 \$026(cr)
	响应 ! 02 A0 00 00 00 00 40 00 01(cr)
	! -收到有效命令; 02-模块地址 02 (H); A0-通道状态 A0h(10100000BIN) (CH5、7: 有传感器); CH5:40h (64个传感器); CH7: 01h (1个) 传感器。

VV 模块的通道状态

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

CH7 CH6 CH5 CH4 CH3 CH2 CH1 CH0

V (高 4 位)

V (低 4 位)

1: 有传感器

0: 无传感器

例如: VV : 80h 即 10000000b CH7 : 有传感器,其余没有

#AA8

名称	读模块所有通道传感器及前端测控单元数据
描述	读回地址为 AA 的模块的所有通道,所有传感器及前端测控单元数据
语法	#AA8 (cr)
	# 是命令符
	AA (范围 00h~FFh) 代表被呼叫模块的两字符 ASC II 码表示的 16 进制地址
	(cr) 为结束符 (0Dh)
响应	> AA (数据数量)(数据) (cr) (校验和) 命令有效

	? AA (cr) 命令无效。 遇到不存在的地址, 将没有响应。
	> 是命令符
	AA 响应模块的两字符 ASCII 码表示的 16 进制地址 (范围 00h~FFh)
	(数据数量) 两个字节 16 进制 (HEX) 数 (注:不是 ASCII 码!), 读回数据的个数 (0000h~0200h), 即所接传感器及前端测控单元数据的个数, 高位在前, 低位在后。
	(数据) 是地址为 AA 的模块, 所连接的所有传感器及前端测控单元数据, 每个传感器及前端测控单元数据 4 个字节 16 进制 (HEX) 数 (注:不是 ASCII 码!) (格式参见各模块说明) 数据格式、数据的组数随所接 ITU 的不同而不同。 (具体格式参见附录 4)
	(cr) 为结束符 (0Dh) (校验和) 数据累加和 读入的 ITU 数据按 ITU 地址码由小到大的顺序传送
示例	使用 8662, 模块地址为 00. 3 个 8901 发送: #008 使用串口程序接受到的十六进制数据: <u>3E 30 30 00 03 01 18 54 21 01 19 51 21 01 19 4F 21 0D</u> <u>52</u> 按照顺序: 3E 是 > 的 ASCII 码 30 30 是 00 的 ASCII 码 00 03 是 数据数量 代表 3 个 8901 01 18 54 21 01 19 51 21 01 19

	<p>4F 21 每四个字节代表一个 8901 的数据 0D 回车符的 ASCII 码</p> <p>52 校验和</p> <p>3E+ 30+ 30+ 00+ 03+ 01+ 18+ 54+ 21+ 01+ 19+ 51+ 21+ 01+ 19+ 4F+ 21+ 0D =02 52</p> <p>取后两位 52</p>
	<p>注意：为了提高处理大量测温点的效率,LTM-8000 系列模块及仪表传回的是 16 进制数据,所以使用命令行方式显示时无法直接看到数据,[]中为读到的数据,而非直接看到的数据。</p> <p>因为响应命令符为“>”的数据使用的是 16 进制,所以不能使用 0Dh 作结束符 (cr) 来判断一帧数据是否结束,而须使用(数据数量)来判定一帧数据是否收完。读入的大量温度值按使用&AA8 读入的传感器 ID 的顺序传送 (参见&AA8)</p>

#AAN

名称	读指定通道所有传感器及前端测控单元数据
描述	读回地址为 AA 的模块通道 N 的所有传感器及前端测控单元数据
语法	#AAN (cr)
	# 是命令符
	AA (范围 00h~FFh)代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址
	N 是指定通道号 (0~7)

	(cr) 为结束符 (0Dh)
响应	>AA(数据数量)(数据) (cr) (校验和) 命令有效
	? AA (cr) 命令无效。 遇到不存在的地址, 将没有响应。
	> 是命令符
	AA 模块两字符 ASCII 码表示的 16 进制地址 (范围 00h~FFh)
	(数据数量) 两个字节 16 进制 (HEX) 格式数 (注: 不是 ASCII 码!) 读回数据的个数 (0000h~0040h), 即通道 N 所接传感器及前端测控单元的个数, 高位在前 (总为 00H), 低位在后。
	(数据) 是地址为 AA 的模块的通道 N, 连接的所有传感器及前端测控单元的数据, 每个数据 4 个字节 16 进制 (HEX) 格式数 (注: 不是 ASCII 码!) (格式参见各模块说明)
	(cr) 为结束符 (0Dh) (校验和) 数据累加和
	读入的大量数据按使用&AAN 读入的传感器 ID 号码顺序传送及使用*AAN 读入的传感器编号由小到大的顺序传送
	注意: 为了提高处理大量数据的效率,LTM-8000 系列模块及仪表传回的是 16 进制数据,所以使用命令行方式显示时无法直接看到数据,[]中为读到的数据,而非直接看到的数据。 因为响应命令符为“>”的数据使用的是 16 进制,所以不能使用 0Dh 作结束符 (cr) 来判断一帧数据

	<p>是否结束，而须使用(数据数量)来判定一帧数据是否收完。</p> <p>读入的大量数据按使用*AA8 读入的传感器编号由小到大的顺序传送（参见*AA8）</p>
--	--

&AA8

名称	读所有通道的传感器 ID 号码
描述	读回地址为 AA 的模块的所有通道的所有传感器 ID 号码，此命令主要针对 LTM88XX(1-wire Buse) 传感器及前端测控单元。
语法	&AA8 (cr)
	& 是命令符
	AA (范围 00h~FFh) 代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址
	(cr) 为结束符 (0Dh)
响应	> AA (ID 数量)(ID) (cr) (校验和) 命令有效
	? AA (cr) 命令无效。
	遇到不存在的地址，将没有响应。
	> 是命令符
	AA 模块两字符 ASCII 码表示的 16 进制地址 (范围 00h~FFh)
(ID 数量) 两个字节 16 进制 (HEX) 数 (注：不是 ASCII 码!)，是读回 ID 号码的个数(0000h~0200h)，即所接传感器的个数，高位在前，低位在后。	

	<p>(ID) 是地址为 AA 的模块连接的所有传感器的 ID, 每个 ID 占 8 个字节 16 进制 (HEX) 数 (注: 不是 ASCII 码!), 共 64 位, 这便是每个传感器出厂自带的全球唯一的 64 位识别号码。(有关 ID 号详细内容参见附录 1)</p>
	<p>(cr) 为结束符 (0Dh) (校验和) 数据累加和</p>
<p>示例</p>	<p>使用 8662, 模块地址为 00。 2 个 18b20 发送: &008</p> <p>使用串口程序接受到的十六进制数据: <u>3E 30 30 00 02 28 C1 37 66 00 00 00 FA 28 87 46</u> <u>66 00 00 00 9D 0D 25</u></p> <p>按照顺序: 3E 是 > 的 ASCII 码 30 30 是 00 的 ASCII 码 00 02 是 数据数量 代表 2 个 18b20 <u>28 C1 37 66 00 00 00 FA 28 87 46 66 00 00 00</u> <u>9D</u> 表示两个 18b20 的 id 号 0D 回车符的 ASCII 码</p> <p>25 校验和</p> <p>3E + 30 + 30 + 00 + 02 + 28 + C1 + 37 + 66 + 00 + 00 + 00 + FA + 28 + 87 + 46 + 66 + 00 + 00 + 00 + 9D + 0D = 05 25 取后两位 25</p>
	<p>注意: 为了提高处理大量数据的效率, LTM-8000 系</p>

	<p>列模块及仪表传回的是 16 进制数据,所以使用命令行方式显示时无法直接看到数据。</p> <p>因为响应命令符为“>”的数据使用的是 16 进制,所以不能使用 0Dh 作结束符 (cr) 来判断一帧数据是否结束,而须使用(数据数量)来判定一帧数据是否收完。</p>
--	--

&AAN

名称	读指定通道的所有的传感器 ID
描述	读回地址为 AA 的模块指定通道的所有传感器及前端测控单元 ID,此命令主要针对 LTM88XX(1-wire Buse) 传感器及前端测控单元
语法	&AAN (cr)
	& 是命令符
	AA (范围 00h~FFh)代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址
	N 是指定通道数 (0~7)
	(cr) 为结束符 (0Dh)
响应	> AA (ID 数量)(ID) (cr) 命令有效
	? AA (cr) 命令无效。 遇到不存在的地址,将没有响应。
	> 是命令符
	AA (范围 00h~FFh)代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址

	(ID 数量) 两个字节 16 进制 (HEX) 格式数 (注: 不是 ASCII 码!), 读回数据的个数 (0000h~0040h), 即通道 N 所接传感器的个数, 高位在前 (总为 00H), 低位在后。
	(ID) 是地址为 AA 的模块通道 N 连接的所有传感器的 ID, 每个 ID 占 8 个字节 16 进制 (HEX) (注: 不是 ASCII 码!), 共 64 位, 这便是每个传感器出厂自带的全球唯一的 64 位识别号码. (ID 号详细内容, 参见附录 1)
	(cr) 为结束符 (0Dh)
	(校验和) 数据累加和命令 &017(cr)
	注意: 为了提高处理大量数据的效率, LTM-8000 系列模块及仪表传回的是 16 进制数据, 所以使用命令行方式显示时无法直接看到数据。 因为响应命令符为 “>” 的数据使用的是 16 进制, 所以不能使用 0Dh 作结束符 (cr) 来判断一帧数据是否结束, 而须使用 (数据数量) 来判定一帧数据是否收完。

&AA9

名称	模块复位命令
描述	命令模块地址为 AA 的模块复位, 重新开始执行模块的内部程序
语法	&AA9 (cr)
	& 是命令符

	AA (范围 00h~FFh)代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址
	9 模块复位命令
	(cr) 为结束符 (0Dh)
响应	> AA (cr) 命令有效
	? AA (cr) 命令无效。 遇到不存在的地址, 将没有响应。
	> 是命令符
	AA (范围 00h~FFh)代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址
	(cr) 为结束符 (0Dh)
示例	命令 &019(cr)
	响应 >01 (cr)
注意	此命令相当于软复位命令, 发送后需等待最多 30 秒钟 (如果满 512 点) 后, 该模块才会对新的命令响应, 模块所接传感器数目越少, 须等待的时间越短. 最少须等待 5 秒钟 (只接 1 个传感器).

*AAN

名称	读指定通道所有传感器及前端测控单元的编号
描述	读回地址为 AA 的模块通道 N 的所有传感器及前端测控单元的编号或各通道 ITU 的地址码。
语法	* AAN (cr)
	* 是命令符
	AA (范围 00h~FFh)代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址

	N 是指定通道号 (0~7)
	(cr) 为结束符 (0Dh)
响应	>AA(数据数量)(数据) (cr) (校验和) 命令有效
	? AA (cr) 命令无效。 遇到不存在的地址, 将没有响应。
	> 是命令符
	AA 模块两字符 ASCII 码表示的 16 进制地址 (范围 00h~FFh)
	(数据数量) 两个字节 16 进制 (HEX) 格式数 (注: 不是 ASCII 码!) 读回数据的个数 (0000h~0040h), 即通道 N 所接传感器及前端测控单元的个数, 高位在前 (总为 00H), 低位在后。
	(数据) 是地址为 AA 的模块的通道 N, 连接的所有传感器及前端测控单元的编号, 每个编号是 1 个字节 16 进制 (HEX) 格式数 (注: 不是 ASCII 码!), 最大为 03FH, 自动升序排列。
	(cr) 为结束符 (0Dh) (校验和) 数据累加和 使用 #AAN 读入的大量数据, 按使用 * AAN 读入的传感器及前端测控单元编号 (地址码) 由小到大的顺序传送
示例	使用 8662, 模块地址为 00、3 个 8901, 8901 在 0 通道上 发送: *000 使用串口程序接受到的十六进制数据:

	<p><u>3E 30 30 00 03 00 01 02 0D B1</u></p> <p>按照顺序：3E 是 > 的 ASCII 码</p> <p>30 30 是 00 的 ASCII 码</p> <p>00 03 是 数据数量 代表 3 个 8901</p> <p>00 01 02 代表三个传感器的编号</p> <p>0D 回车符的 ASCII 码</p> <p>B1 校验和</p> <p>$3E+ 30+ 30+ 00+ 03+ 00+ 01+ 02+ 0D = B1$</p>
	<p>注意：为了提高处理大量数据的效率,LTM-8000 系列模块及仪表传回的是 16 进制数据,所以使用命令行方式显示时无法直接看到数据。</p> <p>因为响应命令符为“>”的数据使用的是 16 进制，所以不能使用 0Dh 作结束符（cr）来判断一帧数据是否结束，而须使用(数据数量)来判定一帧数据是否收完。</p>

\$AAE

名称	读模块的错误代码
描述	读回地址为 AA 的错误代码
语法	\$AAE (cr)
	<p>\$ 是命令符</p> <p>AA （范围 00h~FFh）代表被呼叫模块的两字符 ASCII 码表示的 16 进制地址</p> <p>E 是返回模块的错误代码</p> <p>(cr) 为结束符 (0Dh)</p>
响应	!AA EE CC WW (cr) 命令有效

	<p>? AA (cr) 命令无效。 遇到不存在的地址，将没有响应。</p>																								
	<p>! 是命令符</p>																								
	<p>AA 模块两字符 ASC II 码表示的 16 进制地址(范围 00h~FFh)</p>																								
	<p>EE: 用 ASC II 码表示的 16 进制数, 用于表示错误表示: 定义如下:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">MSB</td> <td colspan="6"></td> <td style="text-align: center;">LSB</td> </tr> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">×</td> <td style="text-align: center;">×</td> <td style="text-align: center;">×</td> <td style="text-align: center;">×</td> </tr> </table> <p style="text-align: center;">保留</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>1: 有相同排序编号警告, (D) 0: 无错误。</p> <p>1: 有相同 ID 号错误; (C) 0: 无错误。</p> <p>1: 此通道传感器温度值经 CRC 校验有错误; (B) 0: 无错误。</p> </div> <div style="width: 45%; border: 1px solid black; padding: 5px;"> <p>1: 此通道传感器 ID 经 CRC 校验有错误; (A) 0: 无错误。</p> </div> </div> <p>注: 其中 A、B、C 三类错误产生的主要原因如下:</p> <ol style="list-style-type: none"> 1. ---测温线路过长, 超过其负载能力; 2. ---测温连线存在不合理分支现象; 3. ---测温线缆所处现场存在强干扰源。 	MSB							LSB	7	6	5	4	3	2	1	0	0	0	0	0	×	×	×	×
MSB							LSB																		
7	6	5	4	3	2	1	0																		
0	0	0	0	×	×	×	×																		
	<p>CC: 用 ASC II 码表示的一个 16 进制数, 用于表示上述错误的通道号 定义如下:</p>																								

	<table border="1" data-bbox="269 182 787 225"> <tr> <td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td> </tr> </table> <p>分别代表 CH0、CH1、……、CH7 通道。 所读数据 = 1：表示此通道有错误； 0：表示此通道无错误。 注：对 LTM8201/3 只有 CH0 通道有效。</p>	7	6	5	4	3	2	1	0
7	6	5	4	3	2	1	0		
	<p>WW：用 ASC II 码表示的一个 16 进制数，用于表示发生警告的通道号 定义如下：</p> <table border="1" data-bbox="269 640 787 684"> <tr> <td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td> </tr> </table> <p>分别代表 CH0、CH1、……、CH7 通道。 所读数据 = 1：表示此通道有重复编号传感器警告； 0：表示此通道无警告。 注：对 LTM8201/3 只有 CH0 通道有效。</p>	7	6	5	4	3	2	1	0
7	6	5	4	3	2	1	0		
	<p>(cr) 为结束符 (0Dh)</p>								

/AAN DH DL OH OL (仅对 8905/8906 有效)

名称	通道数据输出指令
描述	向指定通道、指定地址的 ITU 设定输出数据
语法	/AAN DH DL OH OL (cr)
	用 ASC II 字符表示的 16 进制数
	/ 是命令符 AA (范围 00h~FFh) 代表被呼叫模块的两字符 ASC II 码表示的 16 进制地址

	N	是指定通道号 (0~7)
	DH DL	ITU 为地址号的十位数与个位数。
	OH、OL	为 ITU DO 输出值的高 4 位与低 4 位值。
	(cr)	为结束符 (0Dh)
响应	! AA (cr)	命令有效
	? AA (cr)	命令无效。 遇到不存在的地址，将没有响应。
	!	是命令符
	AA	模块两字符 ASCII 码表示的 16 进制地址 (范围 00h~FFh)
	(cr)	为结束符 (0Dh)

例：向地址为 02 号的 LTM8662 的通道 3 的第 5 号地址 ITU LTM8905 输出 DO 值为：76H (0111 0110 BIN)

即：LTM8905 的 0, 3, 7 继电器不动作

1, 2, 4, 5, 6 继电器动作。命令为：(ASCII 字符)

$\frac{\text{/}}{\uparrow}$ $\frac{0}{\text{---}} \frac{2}{\text{---}}$ $\frac{3}{\uparrow}$ $\frac{0}{\text{---}} \frac{5}{\text{---}}$ $\frac{7}{\text{---}} \frac{6}{\text{---}}$ (0X 0D)
 命令符 8662 地址 8662 通道号 ITU 地址 DO 输出值

\$AAT 读取各通道传感器类型命令

系统中有 3 个 8901, 2 个 18b20

发送:\$00T

使用串口程序接受到的 ASCII 数据：!00 01 40

! 表示命令有效

00,表示模块地址为 00

第三个字节 01 表示有 ITU 产品的通道,0 为高位,1 为低位
把 0,1 (10 进制) 分别换算成 4 位的 2 进制数

0	0	0	0	0	0	0	1
0							1

分别代表从 CH7 到 CH0, 即在 0 通道上有 ITU 传感器

第四个字节 40 表示有 1 wire 产品的通道,4 为高位,0 为低位
把 4,0 (10 进制) 分别换算成 4 位的 2 进制数

0	1	0	0	0	0	0	0
4							0

分别代表从 CH7 到 CH0, 即在 0 通道上有 1 wire 传感器

@ AA CH ID1 ID2 FUNC DN DL 0D ITU 单元

设定命令

其中: 命令字符均为 ASCII 字符

CH——通道号, 取值范围: “0” ~ “7”;

应答: ! AA 0D

ID1, ID2 —— 某通道的 ITU 地址高位及低位

ID1 —— 取值范围: “0” ~ “3”;

ID2 —— 取值范围: “0” ~ “9”

FUNC —— 功能命令符 取值范围: “1”, “2”

其中: “1” —— 为修改 ITU 地址命令

“2” —— 修改 ITU 偏移量命令

DH, DL —— 数据字节高位与低位, 其内容与 FUNC 命令相

关。

“1” ——修改 ITU 地址命令

只有部分 ITU 支持修改地址命令

如：LTM8901V52 版本，LTM8902V20 版本等，最新信息请与长英公司联系或访问公司网站：www.lance-cn.com

此时 DH —— 取值范围为：“0” ~ “3”；

DL —— 取值范围为：“0” ~ “9”；

某一通道 ITU 地址范围为 0~31，所选 DH 与 DL 不可超过此范围

DH — DL 为此地 ITU 的新地址

“2” —— 修改 ITU 偏移量命令

目前仅有 LTM8901V52 支持对其湿度偏移量进行设定。

其中 DH —— 取值范围：“A”，“F”

DL —— 取值范围：“0” —— “15”

DH = “A” 时，表示偏移量 DL 为正值；

DH = “F” 时，表示偏移量 DL 为负值；

示例： 在系统的 ch6 上有一个新版 8901，编号为 00，我们给改为 05

发送：@00600105

接收到：! 00 表示成功

命令解释 @ 00 6 00 1 05

00 代表模块地址

6 代表通道号

00 代表需要修改编号的 8901

1 代表修改编号（2 代表修改偏移量）

附录 1. &AA8 和 &AAN 读到温度/湿度等传感器及智能现场采集控制单元 ID

“1-wire Bus”温度/湿度等传感器及智能现场采集控制单元有一个厂家设定的 64 位全球唯一码，其格式如下：

8 位 CRC 校验码		48 位 序列码	8 位产品序列码	
D63	MSB LSB	MSB LSB	MSB	LSB D0
DS18B20 温度传感器			28H	
LTM8872 壁挂式封装温度探头			28H	
LTM8873 防水铜管封装温度探头			28H	
LTM8875 管道式不锈钢防水温度探头			28H	
LTM8876 超小型防水封装温度探头			28H	
DS18S20 温度传感器			10H	
LTM8802 温湿度一体化探头			26H	
LTM8803-CO2 温度/CO2 探头			26H	
LTM8805 模拟量探头			26H	

附录 2. “1-wire Bus”温度/湿度等传感器及智能现场采集控制单元数据格式

产品 ID 为 28H 的的温度值数据格式：

DS18B20、LTM8872、LTM8873、LTM8875、LTM8876 等

BYTE1 (A)	BYTE2(B)	BYTE3	BYTE4
温度低位字节	温度高位字节	保留	保留
TEMPERATURE LSB	TEMPERATURE MSB	RESERVED	RESERVED

单位(°C)															
2 ³	2 ²	2 ¹	2 ⁰	2 ⁻	2 ⁻	2 ⁻	2 ⁻	S	S	S	S	S	2 ⁶	2 ⁵	2 ⁴
温度算法:							SSSSS = 11111b D=-1, SSSSS = 00000b D=1								
(最小分辨率为							当 D=1 时, 温度值 $T = [(B \text{ and } 7) * 256 + A] * 0.0625$								
0.0625°C)							当 D=-1 时, 温度值 $T = D * [(256 - B) * 256 - A] * 0.0625$								
例如 数据为															
91h							01h			保留		保留			
温度值 Temp = [(1 and 7) * 256 + 145] * 0.0625 = 25.0625°C															

温度值对应举例:

温度(°C)	对应数据(HEX)		温度(°C)	对应数据
+125	07D0		-55	FC90
+85	0550		-25.0625	FF6F
+25.0625	0191		-10.125	FF5E
+10.125	00A2		-0.5	FFF8
+0.5	0008		0	0000

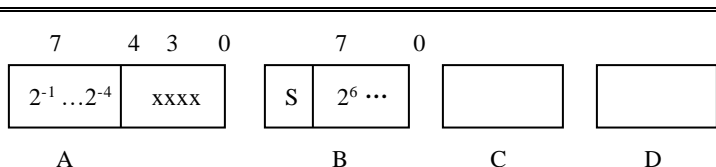
产品 ID 为 10H 的温度值数据格式: DS1820、DS18S20

BYTE1	BYTE2	BYTE3	BYTE4
温度低位字节 TEMPERATURE LSB	温度高位字节 TEMPERATURE MSB	计数常数 B COUNT REMAIN	计数常数 A COUNT per °C

当 BYTE2=00h 时 $D=(+1) * \text{BYTE1}$ 当 BYTE2=FFh 时 $D=(256-\text{BYTE1})$		B	A
标准 0.5℃精度温度算法		正温度值= $0.5 * D$, 负温度值= $-0.5 * D$	
高精度 0.1℃温度算法		1) $D1=D$ and FEh	
高精度正温算法		2) 温度值 $T=0.5 * D1 - 0.25 + (A - B) / A$	
高精度负温算法		3) 温度值 $T= -0.5 * D1 - 0.25 - (A - B) / A$	
例如 数据为			
30h	00h	32h	4bh
标准 0.5℃精度温度算法		温 度 值 = $030h * 0.5=18h(24.0^{\circ}\text{C})$	
高精度 0.1℃温度算法		温 度 值 = $0.5 * 30h-0.25+(4B-32)/4B=18h-0.25+0.333=24.08^{\circ}\text{C}$	

产品 ID 为 26H 的数据格式:

LTM8802 温湿度探头数据格式



其中：A 与 B 两字节为温度数据，C—D 两字节为湿度数据

B 最高位为：符号位。S=0,温度为正值，S=1，温度为负值

A 中低 4 位保留。其余与 B 中低 7 位，为温度数据

温度计算方法：当 S=0 时，为正温： $T=(B*16+(A \& 0XF0)/16)*0.0625$

当 S=1 时，为负温： $A1=255-B$

$$B1=16-(A\&0XF0)/16$$

$$T=(A1*16+B1)*(-0.0625)$$

湿度计算： $h1=0X70+C$

$$h2=0X150+D$$

$$hum=(h1/h2-0.2354)/0.00474 \quad (\%)$$

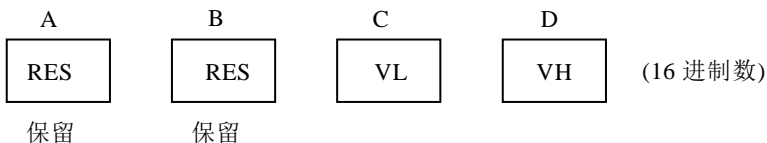
注：C=0XFE 时，表示温度为非法数据。

 { D=0XFE 时，表示温湿度探头供电电压不正常。

上述两种状况均表示 LTM8802 探头有故障，请检查接线及测量输入工作电压。（同 LTM8002）

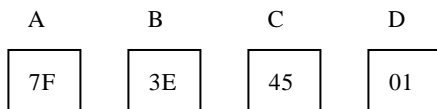
LTM8805 模拟量输入值数据格式

对接入通用型模拟量输入单元，返回上位机仍为 4 个字节。



电压值： $VAD=0.01 \times (VH \times 256 + VL)$ (V)

例：



则： $VAD=0.01 \times (1 \times 256 + (4 \times 16 + 15))$

$=0.01 \times (256 + 64 + 15) = 0.01 \times 325 = 3.25V$

附录 3. “ITU Bus”温度/湿度等传感器及智能现场采集控制单元类型码

每种“ITU Bus 温度/湿度等传感器及智能现场采集控制单元”有一个厂家设定的类型码(可使用&AAN、&AA8 读回)

名称	类型码
LTM8901 智能温湿度探头	01H
LTM8902 K 型热电偶测量单元	02H
LTM8904 8 路光电隔离型开关量输入	04H
LTM8905 8 路继电器（常开型）输出	05H
LTM8906 4 路光隔离输入/4 路继电器输出	06H
LTM8911 4 路标准模拟量隔离型输入单元	0BH

“ITU Bus”温度/湿度等传感器及智能现场采集控制单元 ID 也为 8 个字节，其含义为：

D0—D1—D2—D3……D7

其中：D0 为：ITU 的类型码（一般为产品型号后两位如：8901 为 01）

D1：为 ITU 的版本号：H L

例：52H ， 版本为：V5.2

D2 为：ITU 的偏移量，不支持的 ITU 此值为：FFH

现在仅 LTM8901 支持此功能，为湿度偏移量

D2 取值范围：00—0FH

其中：00—07H 表示：+0~+15%RH

08H 表示：+0%RH

09—0FH 表示：-1~ -15%RH

D3~D7 均为 00H

实例：发送读全部通道 id 指令 &AA8

现在系统中，在 0 通道有 3 个老版 8901，在 6 通道有 1 个新版 8901

发送：&008

使用串口程序接受到的十六进制数据：

3E 30 30 00 04 01 41 FF 00 00 00 00 00 01 41 FF 00 00 00 00
00 01 41 FF 00 00 00 00 00 01 63 00 00 00 00 00 00 0D D6

数据中有下划线的部分是 4 个 8901 的 id,取第一个,和最后一个数据作对比,这两个数据分别是旧版,和新版的 8901

01 41 FF 00 00 00 00 00 (4.1 版 LTM8901)

01 63 00 00 00 00 00 00 (6.3 版 LTM8901)

第一个字节是 01 是 ITU 的类型码

第二个字节分别是 41, 63, 代表版本号, 41 就是 v4.1, 63 就是 v6.3

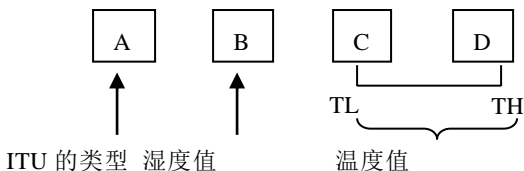
第三个字节分别是 FF, 00, 代表湿度漂移量, FF 代表不支持, 00 代表漂移量为 0%RH

附录 4. “ITU Bus”温度/湿度等传感器及智能现场采集控制单元数据格式

【LTM8901 温湿度探头】

类型编码 01H、返回 1 组数据、4 个字节

[1] 数据格式:



其中: B 为湿度值, 其范围为: 0~200 (为 8901 数据格式)

C 为温度低 8 位, D 为温度高 8 位值

[2] 温度应答数据格式:

D --- TH

7	6	5	4	3	2	1	0
001—温度数据			0	0	X	X	X
000—湿度数据							
数据处理类型			恒为零	FLG	TMP—H		
×	×	×			2^6	2^5	2^4

C --- TL

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	X
TMP—L							
2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}

[3] 温度数据：

TH— BIT3 为符号位 ， BIT3=1,为负温， BIT3=0 为正温。

TH 中， BIT7~5 为数据类型定义， BIT4 恒为 0

TH 中 BIT2~0 及 DATAL 为温度数据

温度值分辨率为: 0.0625℃ (2-4 位)

温度计算公式为：

正温： $TMP=(TH\&07H*256+TL)*0.0625$

负温： $T1=TH\&07H$

$TMP=-(T1*256+DATAL)*0.0625$

[4] 湿度数据：

数据符号节： B — 00~C8H (0~200 DEC)

湿度计算公式： $HUM=0.5\times DATA(\%RH)$

8901 的湿度分辨率为：0.5%RH

8901 的湿度量程为：1%~99%RH

注：当 LTM8901ITU 有故障时，B、C、D 均为 0FFH。

【LTM8902 ITU Bus K 型热偶探头】

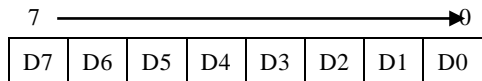
类型编码 02H、返回 1 组数据、4 个字节

数据格式： TYPE—DATAL—DATAH—CHECKSUM

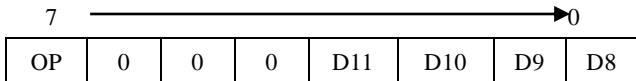
其中： TYPE—02H

CHECKSUM= TYPE+DATAL+DATAH(单字节累加和)

[1] DATAL 格式：



[2] DATAH 格式：



OP——热电偶开路指示； D11——数据最高位

OP=1，热电偶开路（断线），=0，正常； D0——数据最低位

[3] 温度计算公式：

$T = ((DATAH \& 0FH) \times 256 + DATAL) \times 0.25$

示例：DATAH=09H, DATAL=85H(133 DEC)

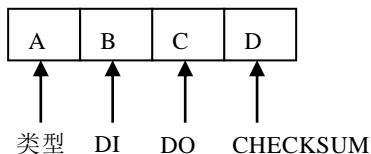
$$\begin{aligned} \text{则 } T &= ((09 \& 0FH) \times 256 + 133) \times 0.25 \\ &= (9 \times 256 + 133) \times 0.25 \end{aligned}$$

$$=2437 \times 0.25 = 609.25^{\circ}\text{C}$$

【LTM8904/8905/8906 ITU Bus 多路光电隔离输入/输出单元】

返回 1 组数据，4 个字节

数据格式：



类型定义： LTM8904 - 04 H

LTM8905 - 05 H

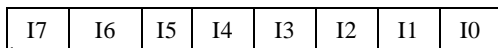
LTM8906 - 06 H

CHECKSUM= TYPE+DATA_L+DATA_H(单字节累加和)

DI : 开入数据； DO : 输出数据，

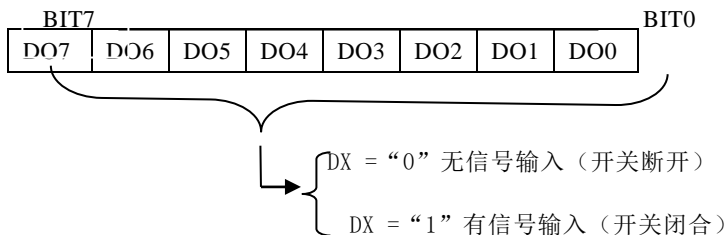
[1] LTM8904： DI 数据格式如下， DO 为 00H

BIT7 ————— BIT0



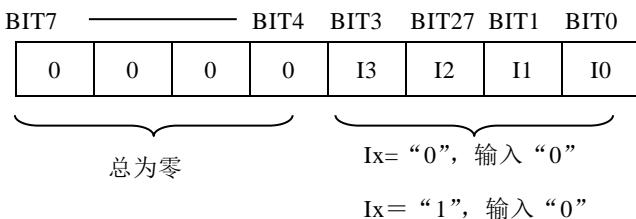
IX = “0” 无信号输入（开关断开）
IX = “1” 有信号输入（开关闭合）

[2] LTM8905： DI 为 00H， DO 数据格式如下

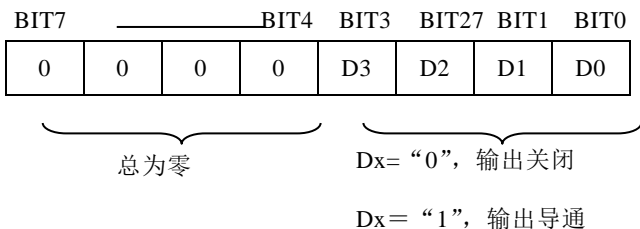


[3] LTM8906: DI/DO 数据格式如下

DI 部分: (输入状态)



DO 部分: 为当前输出状态值



【LTM8911 ITU Bus 4 通道光电隔离型标准模拟量输入单

元】

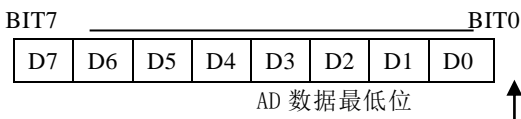
类型编码 - 0BH 返回 4 组数据, 16 个字节

数据格式: TYPE—DATAL—DATAH—CHECKSUM

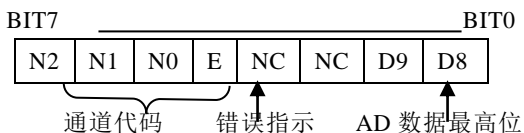
其中: TYPE—0BH

CHECKSUM= TYPE+DATAL+DATAH(单字节累加和)

[1] DATAL 格式:



[2] DATAH 格式:



其中:

N2	N1	N0	
0	0	0	—— CH0
0	0	1	—— CH1
0	1	0	—— CH2
0	1	1	—— CH3

当 E = 0 时, 数据正确。

E = 1, (同时, DATAL=FFH) 表示此通道有故障

AD 值计算公式： $V = ((DATAH \& 03H) \times 256 + DATAL) \times 5.0/1023$

示例： DATAL=36H， DATAH=42H

表示为 CH2 的 AD 数据，无错误。

42 & 03H=02H

$02 \times 256 + 36H = 512 + 54 = 566$

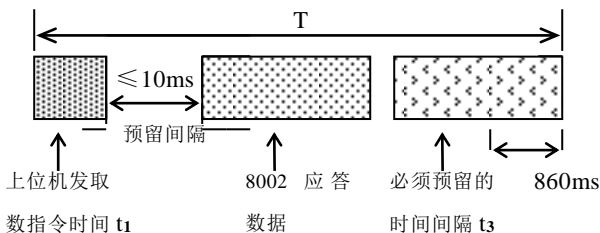
$V = 566 \times 5.0/1023 = 2.766V$

换算成 4~20mA 信号,则为： $I = 2.766 \times 4 = 11.064mA$.

附录 5. 上位机访问 LTM8000 系列模块的最小间隔时间

对于 LTM8002/8300/8301 等单 CPU 模块，因为要给 CPU 留出处理现场信息的时间，所以在访问同一需要有一定时间间隔。

上位机最小访问周期计算方法：



其中： $t_1 = 5 \times 9600 / B$ 为波特率（可为：9600、19200、38400）

$t_2 = (6 + N \times 4) \times 9600 / B$ N 为传感器个数

$t_3 = 860 + N \times 15$

即 $T_{\min}=t_1+t_2+t_3+10\text{ms}$

$$=10+9600/B \times (5+6+N \times 4)+860+N \times 15$$

$$=(11+4 \times N) \times 9600/B+870+15 \times N$$

例：波特率 $B=9600\text{BPS}$ ，传感器个数 $N=10$ 个，则上位机访问 8002 的最小周期为：

$$T_{\min}=(11+4 \times 10) \times 9600/9600+870+15 \times 10$$

$$=51+870+150$$

$$=1071(\text{ms})$$

读回的温度值为上位机上一次访问 LTM8002 采样时刻+ T_{\min} 时刻的值（第一次采样时为上电时刻+ T_{\min} 时刻的值）。

例如：第一次采样时刻为 T_0 ，第二次为 T_1 ...依此类推。

那么， T_0 时刻读到的为上电+ T_{\min} 时刻的值 T_1 时刻读到的为 T_1+T_{\min} 时刻的值...依此类推。所以，当采样周期 $> 2 * T_{\min}$ 时，应在采样时刻访问 LTM8002 两次，两次间隔 $\geq T_{\min}$ ，以便获得较准确时刻的温度值。

注：访问模块频率过快不会影响模块的运行，但是在 LTM8000 模块没有处理完毕时，对上位机不做任何应答。

LTM8303 等双 CPU 模块无需此时间间隔。