# MTH01-SPI
Temperature & Humidity Sensor

- Relative humidity and temperature sensor
- 相对温度和湿度传感器
- Pre-calculated temperature and humidity read out, no extra calculation needed
- 预先计算的温度和湿度读出，无需额外计算
- <mark>Dew Point Calculation possible</mark>
- 可以计算露点
- Fully Calibrated, Digital Output
- 完全校准，数字输出
- Excellent Long Term Stability
- 良好的长期稳定性
- No Extra Component Required
- 无需额外的组件
- Ultra Low Power Consumption
- 超低功耗
- Fully Interchangeable

- 完全可互换
- Small Size
- 尺寸小
- Automatic Power Down
- 自动断电
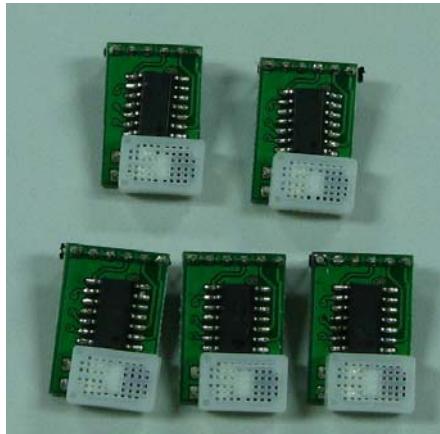
Product Summery
产品简介

The MTH01 is a MCU based temperature and relative humidity sensor module, comprising a SPI interface (master mode) for direct temperature and humidity read out. The digital output is pre-calculated and no extra calculation is required. The system applied two sensor elements: NTC type high precision temperature sensor and a resistor type relative humidity sensor from Japan. With a very unique and patented relative humidity calculation algorithm, the system can assure accurate relative humidity output through fine tuned temperature compensation mechanism. Thus very high accurate reading of humidity in the full temperature range (0-50C) can be assured.

该 MTH01 是一款基于温度和相对湿度的传感器模块，包括一个 SPI 接口（主模式）为直接的温度和湿度读出。数字输出是预先计算并无额外计算要求所必须的。改系统适用于两个传感器要素：NTC 型高精度温度传感器和电阻式相对湿度传感器来自日本。有一个非常独特的专利和相对湿度的计算算法，该系统通过良好的温度调节补偿机制可以保证准确相对湿度输出。如此高精度读湿度在整个温度范围（0－50С）都可以保证。

Applications
适用产品

- <mark>HVAC</mark>
- <mark>暖通控制产品</mark>
- <mark>Consumer Products</mark>
- 消费类产品
- Weather Stations
- 气象站
- Humidifiers

- 加湿机
- Dehumidifiers
- 抽湿机
- Test and Measurements
- 测试与测量
- Data Logging
- 数据记录
- <mark>White Goods</mark>
- 白色家电

MTH01-SPI
Temperature & Humidity Sensor

Block Diagram
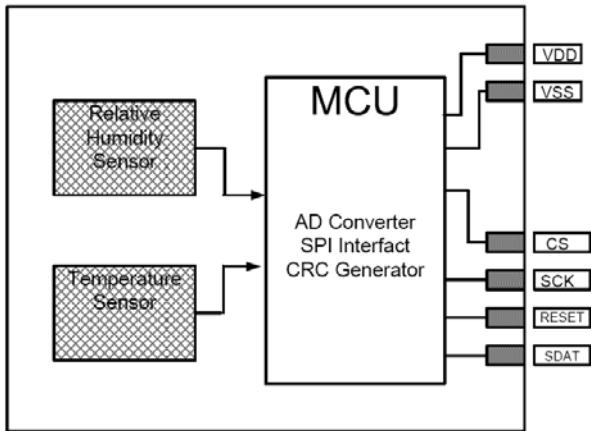
1. Module Performance Specification
模块性能规格

Parameter 参数	Conditions 条件	Min 最小值	Typ	Max 最大值	Unit 单位
Humidity 湿度					
Resolution 分辨率				1	%
Repeatability 重复性			1		%
Accuracy Uncertainty 精度不准确性	Temperature at 0C – 50C range	0	+,-3	+,-5	%
Interchangeability 互换性		Fully Inter Changeable			
Nonlinearity 非线性			1	2	%
Range 范围	Temperature at 0C – 50C range	18		98	%
Response Time 响应时间	63% slowly move air		60		Second
Hysterisis 重复性	Non-condensate 不结露		1	2	%
Long Term Stability 长期稳定性	Non-condensate		2		%/yr
Temperature 温度					
Resolution			0.1		°C
Repeatability			0.1	0.2	°C
Range		-40		70	°C
Accuracy	25		+/-0.5	+/-1.0	°C
Response Time	delta T=1.0		60		S

MTH01-SPI
Temperature & Humidity Sensor

## 2. Sensor Interface 传感器接口
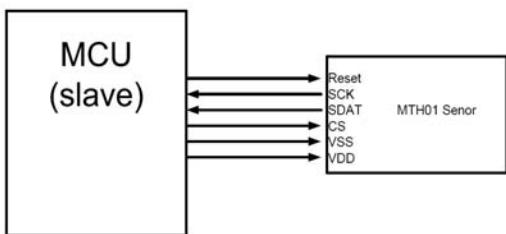
Figure 1: Typical Application
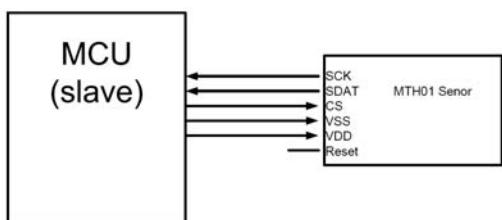
Figure 2: Reduce IO Usage Application

2.1 Power Pins

The MTH01 sensor module requires a voltage supply between 2.0 to 5.5. After power, the sensor needs 20ms to complete its internal reset process. After reset finished, the sensor will make a measurement automatically and if the CS pin is low, then the measured data will be output automatically.

MTH01 传感器模块需要 2.0 到 5.5 伏的电压供应。上电后，传感器需要 20 豪秒时间来完善其内部的重置过程。重置完成后，传感器会自动测量，若 CS 脚被拉低，测量数据会自动输出。

Power pins should be decoupled through a 10-100nF capacitor. Where in those applications with high power noise environment, it is strongly recommended to use a 10uF tantalum capacitor to protect the sensor from interferences.

需通过一个 10-100nF 的电容退耦。凡是在高分贝噪音环境下的，强烈建议使用一个 10uf 钽电容，以保护传感器免被干扰。

2.2 Serial Interface 串行接口

The serial interface is optimized for convenient reading and reducing IO usage. Application engineer should be kept in mind the characteristics of the IO pins for applications where current consumption is critical.

串型接口是为方便读取和减少 IO 口用量进行的优化设计。应用工程师应牢记 IO 口的特性，对电流消耗有要求的场合是至关重要的

Reset, INPUT pin, has 50k pull up resister connected internally, thus during normal application, the pin should not be tied to low unless reset is really needed.

CS, INPUT pin, has 100k pull up. Negative edge will wake up the module and send the previous measured data first, after that temperature and humidity measurement will take place. As long as CS is low, measure<->send data cycle is repeated.

首先发送先前测到的数据，然后再开始温度和湿度的测量。只要 CS 拉低，测量，发数据的过程就会一直持续下去。

SDAT and SCK, OUTPUT pin, is in CMOS output mode. Thus for external MCU connection, ports connecting to this two pins should be in input mode without pull high resister to avoid high current.

因此，对于外部的 MCU 连接，端口应该设定在输入模式，不带内部上拉电阻以避免额外电流产生

2.2.1 CS

CS is to activate the sensor and triggers to send out the previous measured temperature and humidity value through SCK and SDAT line. After data sent out, an internal AD convert cycle will start automatically. If CS line is kept low after AD convert, the new measured data will be output through the data lines until CS

line goes high, which is to terminate the data output and AD convert process. When CS is high, the system enters sleep mode to reduce power consumption.

通过拉低 CS 就会立刻将上次测量数据发送出来。数据发出后，一个将开始一个新温度湿度数据的测量。如果测量完成后 CS 还是低电平，新测量数据将会通过数据线输出直到 CS 线变高为止，即数据输出和 AD 转换过程终止。当 CS 高时，系统进入睡眠模式，以降低电力消耗。

2.2.2 SCK

SCK is to synchronize the communication between MCU and the sensor module. It runs at around 2～3 KHZ speed, which should be an ideal speed for both high and low speed MCU systems. Once all data are out, SCK line will remain low.

SCK 是使 MCU 和传感器模块间保持同步。它以 3-6 千赫的速度运行，这应是高速和低速单片机系统都理想的速度。一旦所有数据都出来了，SCK 线将保持为低。

2.2.3 SDAT

SDAT is to transfer data to MCU. The SDAT line is valid after SCK goes high.

Once data are all out, SDAT line will remain low.

SDAT 是将数据传输到微控制器。当 SCK 线变高时 SDAT 线才有效。当数据都出现后，SDAT 线将保持为低。

2.2.4 Reset

During normal time, Reset pin should be kept high. In cases when sensor module can't output data after CS line is set low for 50ms, then the Reset pin should be pulled low for 15ms to make a proper reset for the sensor module to release the system from unknown state.

在正常期间，复位脚应该要保持高。如果当 CS 线拉低 50ms 后仍然不能有数据输出的话，可以拉低复位脚来复位系统。对高可靠性要求环境中建议使用此复位功能。**3. Bus Timing**

To start a data reading, set the CS line low will wake up the sensor module and start to transfer data through the data line. The bus timing is specified in the following way:

开始数据读取时，将 CS 线调低能激活传感器模块并通过数据线开始转换数据。数据时序以如下方式被指定的：

Figure 3: Bus timing

T1: 2-5ms Nominal   T2:250uS   T3: 200-400mS

Once CS is set high, sensor module will terminate data transfer process right way, and one extra measurement will start automatically before entering power down mode. The purpose of doing so is that data will be always ready to be transferred once a read command is initiated by pulling low CS pin. This is helpful in reducing waiting time before getting any data back from the sensor module. The imperfect part of this practice is that

the data is for previous measured, not up to the current second.

一旦 CS 拉高，传感器模块将会立即终止数据传输进程，并且一个额外的测量将会在进入断电模式前自动开始。这样做的目的是因为一旦一个读取命令由于拉低 CS pin 就会被启动，数据随时准备好被转换。这有利于减少从传感器模块接收反馈的数据前所需等待的时间。不完善的一方面是接收的数据是先前测量的，并不是由当前这秒决定的。

Data output bit stream starts with MSB of temperature (2 byte) data, followed with one byte humidity and one byte CRC.

数据输出位数流量先从 MSB 的温度（2 字节）数据开始，随后是一字节的湿度和一字节 CRC。

| 0 | 0 | 0 | 0 | T_bit800 | ⬛⬛⬛⬛⬛⬛⬛⬛⬛⬛⬛⬛⬛ | T_bit1 | RH_bit80 | | | RH_bit1 | CRC_bit80 | | | CRC_bit1 |
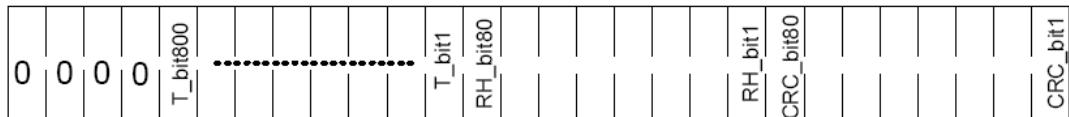
Figure 4: Data organization illustration

3.1 Converting output data to temperature and humidity 转换输出数据为温度和湿度

The temperature value has added an offset value of 40C to avoid negative temperature sign flag problem and multiplied by ten. Thus real temperature can be obtained by deducting 0190(Hex).

温度值增加了一个 40C 的抵消值以避免负温度符号，再乘以 10。因此，准确的温度可以通过减掉 0190H（400）得到。

For example:

Data stream: 0000001010110011010100010000000

$T\_offset=0x02B3=691$　　　　$T\_real=691-400=291=29.1C$

$RH=0x51=81\%$　　　　$CRC=0x7F$

4. DC Characteristics

Parameter 参数	Conditions 条件	Min	Typ	Max	Unit
Power Supply DC 直流电		2	3	5.5	V
Power Supply Current 当前电流供应	AD measuring AD 测量		100	200	uA
	data transferring 数据转换		0.2		mA
	Standby 待机		0.2		uA
Low Level Output Voltage 低水平电压输出		0		20%	Vdd
High Level Output Voltage 高水平电压输出		80%		100%	Vdd
Low Level Input Voltage 低水平输入电压	Negative Edge 负极	0%		30%	Vdd
High Level Input Voltage 高水平输入电压	Positive Edge 正极	70%		100%	Vdd
Pads Leakage Current			1		uA

Output High Current 输出高电流	80%VDD		10		mA	
Output Low Current 输出低电流	20%VDD		20		mA	

5. Package Information 包装信息

The module is in 12 x 20 mm size, with six pins:
模块尺寸为 12×20mm，6 个引脚：

VDD VSS Reset SDAT SCK CS.
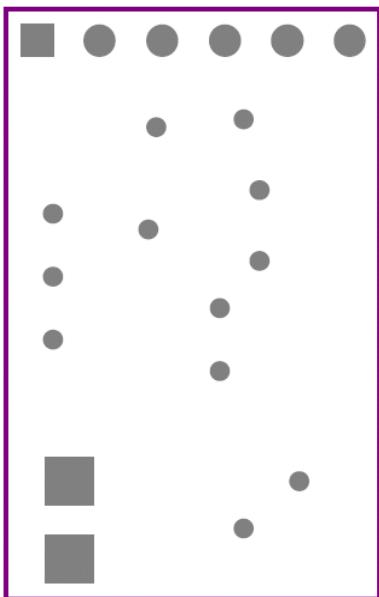
Figure5: outline of the sensor module

6. CRC

CRC stands for Cyclic Redundancy Check. It is one of the most effective error detection schemes and requires a minimal amount of hardware.
CRC表示循环冗余检查。这是其中最有效的一个错误检测方案，对硬件的要求很小。

The polynomial used in the MTH01 is: $x^8 + x^5 + x^4.$ The types of errors that are detectable with this polynomial are:

在MTH01中的多项式使用是：$x^8 + x^5 + x^4.$。这类型的错误用该多项式是可以检测出来：

1. Any odd number of errors anywhere within the transmission.
   在传输范围内任何地方任何单数的误差。
2. All double-bit errors anywhere within the transmission.
   在传输范围内任何地方所有双比特错误。
3. Any cluster of errors that can be contained within an 8-bit "window" (1-8 bits

incorrect).

任何群集的错误可以被一个8位的"窗口"包含（1-8位不正确）。

4. Most larger clusters of errors.

最大群集错误。

The receiver can perform the CRC calculation upon the first part of the original message and then compare the result with the received CRC- 8. If a CRC mismatch is detected, the MTH01 should be reset and a new measurement should be repeated.

接收机可以显示CRC通过源信息的第一部分计算结果，然后与接收到的CRC- 8的结果进行对比。如果一个CRC被检测到是不匹配的，则MTH01需重置，重复新的测量。

This application note will cover two methods for checking the CRC. The first "Bitwise" is more suited for hardware or low level implementation while the later "Bytewise" is the preferred method for more powerful microcontroller solutions.

该应用指南包括两种检测CRC的方法。第一种"位运算符"更适合于硬件或低一级的检测，而后面的"位运算符"则是更高级的微控制器解决方案的首选。

6.1 Bitwise位运算符

With the bitwise method, the receiver copies the structure of the CRC generator in hard- or software.

用为运算符方法，接收器复制CRC中的硬件或软件结构

An algorithm to calculate this could look like this:

一个运算来解决计算问题可以看成是这样的：

1) Initialize CRC Register to 0

使CRC寄存器初始化为0

2) Compare each (transmitted and received) bit with bit 7

用位7与每个（发送和接收）位比较

3) If the same: shift CRC register, bit0='0' else: shift CRC register and then ivert bit4 and bit5, bit0='1'   (see figure 1)

如果相同：bit0='0'，移位CRC寄存器；bit0='1'，移位CRC寄存器并对bit4和bit5取反，（见图1）

4) receive new bit and go to 2)

接收新的位，继续第二步

5) After the last byte is treated, the result in CRC register is the calculated.

将最后一个字节调好后，在CRC注册的结果就被计算出来了。
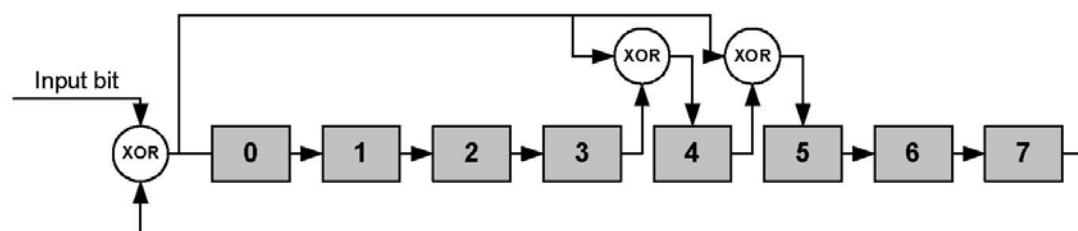
Figure 6 CRC generator state machine

6.1.1 Example for bitwise 位运算符举例

Example 1: 0x05-0x09-0x31 CRC calculate example

Input bit's	bit7 … bit0	0x	dec	Comment
	0000'0000			Start value
0	0000'0000	00	0	1st bit of command
0	0000'0000	00	0	2nd bit of command
0	0000'0000	00	0	…
0	0000'0000	00	0	
0	0000'0000	00	0	
1	0011'0001			CRC EXOR polynom
0	0110'0010			
1	**1111'0101**	**F5**	**245**	**CRC after command**
0	1101'1011			1st byte (MSB) of measurement
0	1000'0111			
0	0011'1111			
0	0111'1110			
1	1100'1101			
0	1010'1011			
0	0110'0111			
1	**1111'1111**	**FF**	**255**	**CRC value**
0	1100'1111			2nd byte (LSB) of measurement
0	1010'1111			
1	0101'1110			
1	1000'1101			
0	0010'1011			
0	0101'0110			
0	1010'1100			
1	**0101'1000**	**58**	**88**	**Final CRC value**

Example 1: 0x40 CRC calculate example

Input bit's	bit7 … bit0	0x	dec	Comment
	0000'0000			Start value see below [(1)]

0	0000'0000	00	0	1st bit of command
0	0000'0000	00	0	2nd bit of command
0	0000'0000	00	0	…
0	0000'0000	00	0	
0	0000'0000	00	0	
1	0011'0001			CRC EXOR polynom
1	0101'0011			
1	**1001'0111**	**97**	**151**	**CRC after command**
0	0001'1111			1st bit (MSB) of status register
1	0000'1111			
0	0001'1110			
0	0011'1100			
0	0111'1000			
0	1111'0000			
0	1101'0001			

6.2 Bytewise

With this implementation the CRC data is stored in a 256 byte lookup table.
CRC数据的执行罗列在以下256字节的查询表内
Perform the following operations:
执行下列步骤：

1. Initialize the CRC register with value 0
   是CRC注册值初始化为0
2. XOR each (transmitted and received) byte with the previous CRC value. The result is the new byte that you need to calculate the CRC value from.
   和上次得到的CRC值XOR(异或)处理，结果是用来查表值。
3. Use this value as the index to the table to obtain the new CRC value.
   使用该值作为表中的指针以获取新的CRC值。
4. Repeat from 2.) until you have passed all bytes through the process.
   从步骤2开始重复，直到所有字节都通过整个检测。
5. The last byte retrieved from the table is the final CRC value.
   最后一个字节来自该表的最后的CRC值。

6.2.1 256 byte CRC Lookup table 256字节CRC查询表

MTH01-SPI
Temperature & Humidity Sensor

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	49	98	83	196	245	166	151	185	136	219	234	125	76	31	46	67	114	33	16	135	182	229	212	250	203	152	169	62	15	92	109
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
134	183	228	213	66	115	32	17	63	14	93	108	251	202	153	168	197	244	167	150	1	48	99	82	124	77	30	47	184	137	218	235
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
61	12	95	110	249	200	155	170	132	181	230	215	64	113	34	19	126	79	28	45	186	139	216	233	199	246	165	148	3	50	97	80
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
187	138	217	232	127	78	29	44	2	51	96	81	198	247	164	149	248	201	154	171	60	13	94	111	65	112	35	18	133	180	231	214
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
122	75	24	41	190	143	220	237	195	242	161	144	7	54	101	84	57	8	91	106	253	204	159	174	128	177	226	211	68	117	38	23
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
252	205	158	175	56	9	90	107	69	116	39	22	129	176	227	210	191	142	221	236	123	74	25	40	6	55	100	85	194	243	160	145
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
71	118	37	20	131	178	225	208	254	207	156	173	58	11	88	105	4	53	102	87	192	241	162	147	189	140	223	238	121	72	27	42
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255
193	240	163	146	5	52	103	86	120	73	26	43	188	141	222	239	130	179	224	209	70	119	36	21	59	10	89	104	255	206	157	172

6.2.2 Code example for lookup table 查询表代码举例

The following procedure calculates the CRC-8. The result accumulates in the variable CRC.
Var
CRC : Byte;
Procedure calc_CRC(X: Byte);

Const
CRC_Table : Array[0..255] of Byte = (
0, 49, 98, 83, 196, 245, 166, 151, 185, 136, 219, 234, 125, 76, 31, 46, 67, 114, 33, 16, 135, 182, 229, 212, 250, 203, 152, 169, 62, 15, 92, 109, 134, 183, 228, 213, 66, 115, 32, 17, 63, 14, 93, 108, 251, 202, 153, 168, 197, 244, 167, 150, 1, 48, 99, 82, 124, 77, 30, 47, 184, 137, 218, 235, 61, 12, 95, 110, 249, 200, 155, 170, 132, 181, 230, 215, 64, 113, 34, 19, 126, 79, 28, 45, 186, 139, 216, 233, 199, 246, 165, 148, 3, 50, 97, 80, 187, 138, 217, 232, 127, 78, 29, 44, 2, 51, 96, 81, 198, 247, 164, 149, 248, 201, 154, 171, 60, 13, 94, 111, 65, 112, 35, 18, 133, 180, 231, 214, 122, 75, 24, 41, 190, 143, 220, 237, 195, 242, 161, 144, 7, 54, 101, 84, 57, 8, 91, 106, 253, 204, 159, 174, 128, 177, 226, 211, 68, 117, 38, 23, 252, 205, 158, 175, 56, 9, 90, 107, 69, 116, 39, 22, 129, 176, 227, 210, 191, 142, 221, 236, 123, 74, 25, 40, 6, 55, 100, 85, 194, 243, 160, 145, 71, 118, 37, 20, 131, 178, 225, 208, 254, 207, 156, 173, 58, 11, 88, 105, 4, 53, 102, 87, 192, 241, 162, 147, 189, 140, 223, 238, 121, 72, 27, 42, 193, 240, 163, 146, 5, 52, 103, 86, 120, 73, 26, 43, 188, 141, 222, 239, 130, 179, 224, 209, 70, 119, 36, 21, 59, 10, 89, 104, 255, 206, 157, 172);

Begin
CRC := CRC_Table[X xor CRC];
End;

7. Important Notice

Do not use this product as safety or emergency stop devices or in any other

applications where failure of the product could result in personal injury. Failure to comply with this instruction could result in death or fatal injury.
勿将该产品作为安全或紧急避险系统或其他由于损坏而会导致人身伤害的产品中。未能遵守这项指示，可能会导致死亡或致命的伤害。

7.1 ESD precautions <mark>ESD</mark> 预防措施

To prevent ESD related damage and/or degradation, take normal ESD precautions when handling the device.
为防止公共服务电子化有关的损害和/或退化，处理装置时采取一般 ESD 预防措施。

7.2 Warranty 保修
We make no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor do we assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation consequential or incidental damages.
至于是否适合其产品在任何特定场合的用途，我们不作任何保修，陈述或保证，我们也不承担任何由应用或使用任何产品或电路而产生的责任，特别声明任何和所有责任，包括但不限于相应而生或附带的损害赔偿。