18195-02
1.6.2018

METER

INTEGRATOR GUIDE
ATMOS 22 ULTRASONIC ANEMOMETER

SENSOR DESCRIPTION

The ATMOS 22 Ultrasonic Anemometer is designed for continuous monitoring of wind speed and direction (see Measurement Specifications). Ultra-low power consumption and a robust, no moving parts design that prevents errors because of wear or fouling, make the ATMOS 22 ideal for long-term, remote installations.

NOTE: The ATMOS 22 replaces the DS-2 (discontinued) and the outputs and order are not the same as the DS-2. Any DS-2 replaced by the ATMOS 22 will require data acquisition system reprogramming based on information located in the tech note Integrators replacing DS-2 with ATMOS 22.

APPLICATIONS

- Weather monitoring
- Microenvironment monitoring
- In-canopy wind measurement
- Spatially distributed environmental monitoring
- Wind profiling
- Crop weather monitoring
- Fire danger monitoring/mapping
- Weather networks

ADVANTAGES

- Robust, no moving parts design
- Small form factor
- Low-input voltage requirements
- Low-power design supports battery-operated data loggers
- Supports the SDI-12 three-wire interface
- Tilt sensor informs user of out-of-level conditions
- No configuration necessary

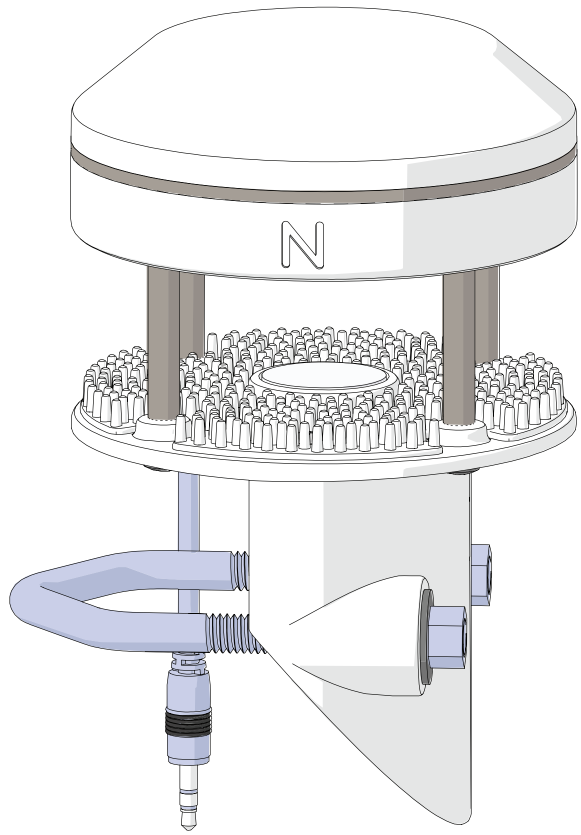

Figure 1 ATMOS 22 Ultrasonic Anemometer

PURPOSE OF THIS GUIDE

METER provides the information in this integrator guide to help ATMOS 22 Ultrasonic Anemometer customers establish communication between these sensors and their data acquisition equipment or field data loggers. Customers using data loggers that support SDI-12 sensor communications should consult the data logger user manual. METER sensors are fully integrated into the METER system of plug-and-play sensors, cellular-enabled data loggers, and data analysis software.

COMPATIBLE FIRMWARE VERSIONS

This guide is compatible with firmware versions 1.03 or newer.

METER Group, Inc. USA
2365 NE Hopkins Court, Pullman, WA 99163
T +1.509.332.2756 F +1.509.332.5158
E info@metergroup.com W metergroup.com

SPECIFICATIONS

MEASUREMENT SPECIFICATIONS

Horizontal Wind Speed	
Range:	0–30 m/s
Resolution:	0.01 m/s
Accuracy:	The greater of 0.3 m/s or 3% of measurement

Wind Gust	
Range:	0–30 m/s
Resolution:	0.01 m/s
Accuracy:	The greater of 0.3 m/s or 3% of measurement

Wind Direction	
Range:	0°–359°
Resolution:	1°
Accuracy:	±5°

Tilt	
Range:	0°–180°
Resolution:	0.1°
Accuracy:	±1°

Dimensions
10 cm diameter x 16 cm height

Cable Length
5 m (custom cable lengths are available for an additional cost)

ELECTRICAL AND TIMING CHARACTERISTICS

Supply Voltage (VCC) to GND	
Minimum	3.6 VDC continuous
Typical	
Maximum	15.0 VDC continuous

Digital Input Voltage (logic high)	
Minimum	2.8 V
Typical	3.0 V
Maximum	15.0 V

Digital Input Voltage (logic low)	
Minimum	−0.3 V
Typical	0.0 V
Maximum	0.8 V

Power Line Slew Rate	
Minimum	1.0 V/ms
Typical	
Maximum	

Current Drain (during measurement)	
Minimum	0.05 mA
Typical	0.125 mA
Maximum	0.5 mA

Current Drain (while asleep)	
Minimum	0.05 mA
Typical	0.125 mA
Maximum	0.15 mA

Operating Temperature Range	
Minimum	−40 °C
Typical	
Maximum	50 °C

Power Up Time (SDI ready)—aRx! Commands	
Minimum	
Typical	10 s
Maximum	

Power Up Time (SDI ready)—Other Commands	
Minimum	
Typical	800 ms
Maximum	

Measurement Duration	
Minimum	
Typical	110 ms
Maximum	3,000 ms

COMPLIANCE

Manufactured under ISO 9001:2015

EM ISO/IEC 17050:2010 (CE Mark)

EQUIVALENT CIRCUIT AND CONNECTION TYPES

Refer to Figure 2 and Figure 3 to connect the ATMOS 22 to a logger. Figure 2 provides a low-impedance variant of the recommended SDI-12 Specification v1.3.

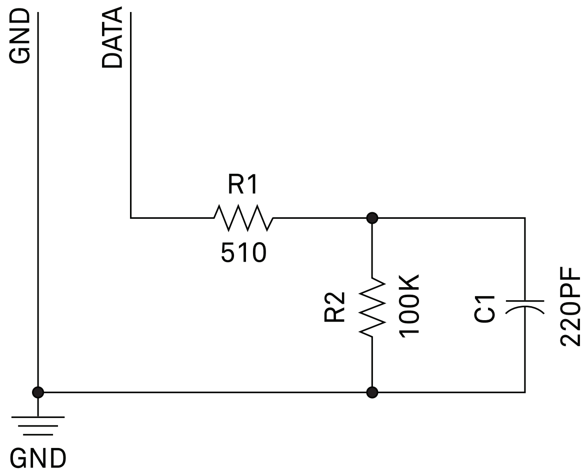

Figure 2 Equivalent circuit diagram

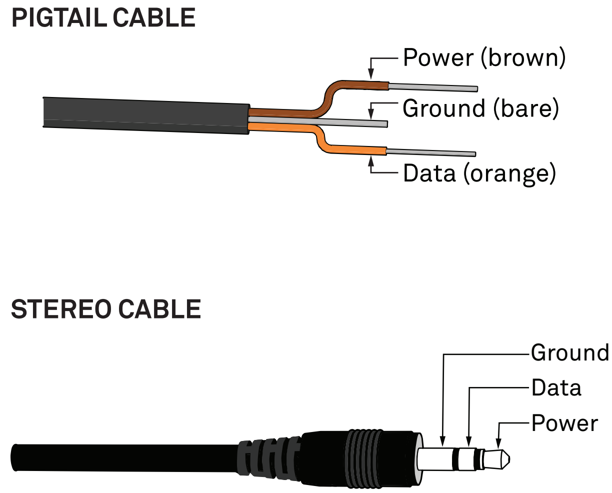

Figure 3 Connection types

⚠ SAFETY PRECAUTIONS

METER sensors are built to the highest standards, but misuse, improper protection, or improper installation may damage the sensor and possibly void the warranty. Before integrating sensors into a sensor network, follow the recommended installation instructions and implement safeguards to protect the sensor from damaging interference.

SURGE CONDITIONS

Sensors have built-in circuitry that protects them against common surge conditions. Installations in lightning-prone areas, however, require special precautions, especially when sensors are connected to a well-grounded third-party logger.

Visit metergroup.com for articles containing more information.

CABLES

Improperly protected cables can lead to severed cables or disconnected sensors. Cabling issues can be caused by many factors, including rodent damage, driving over sensor cables, tripping over the cable, not leaving enough cable slack during installation, or poor sensor wiring connections. To relieve strain on the connections and prevent loose cabling from being inadvertently snagged, gather and secure the cable travelling between the ATMOS 22 and the data acquisition device to the mounting mast in one or more places. Install cables in conduit or plastic cladding when near the ground to avoid rodent damage. Tie excess cable to the data logger mast to ensure cable weight does not cause sensor to unplug.

SENSOR COMMUNICATIONS

METER digital sensors feature a 3-wire interface following SDI-12 protocol for communicating sensor measurements.

SDI-12 INTRODUCTION

SDI-12 is a standards-based protocol for interfacing sensors to data loggers and data acquisition equipment. Multiple sensors with unique addresses can share a common 3-wire bus (power, ground, and data). Two-way communication between the sensor and logger is possible by sharing the data line for transmit and receive as defined by the standard. Sensor measurements are triggered by protocol command. The SDI-12 protocol requires a unique alphanumeric sensor address for each sensor on the bus so that a data logger can send commands to and receive readings from specific sensors.

Download the SDI-12 Specification v1.3 to learn more about the SDI-12 protocol.

DDI SERIAL INTRODUCTION

The DDI serial protocol is the method used by the METER family of data loggers for collecting data from the sensor. This protocol uses the data line configured to transmit data from the sensor to the receiver only (simplex). Typically, the receive side is a microprocessor UART or a general-purpose IO pin using a bitbang method to receive data. Sensor measurements are triggered by applying power to the sensor. When the ATMOS 22 is set to address `0`, a DDI serial string is sent on power up, identifying the sensor.

INTERFACING THE SENSOR TO A COMPUTER

The serial signals and protocols supported by the sensor require some type of interface hardware to be compatible with the serial port found on most computers (or USB-to-serial adapters). There are several SDI-12 interface adapters available in the marketplace; however, METER has not tested any of these interfaces and cannot make a recommendation as to which adapters work with METER sensors. METER data loggers and the ProCheck hand-held device can operate as a computer-to-sensor interface for making on-demand sensor measurements. For more information, please contact Customer Support.

METER SDI-12 IMPLEMENTATION

METER sensors use a low-impedance variant of the SDI-12 standard sensor circuit (Figure 2). During the power-up time, sensors output some sensor diagnostic information and should not be communicated with until the power-up time has passed. After the power up time, the sensors are compatible with all commands listed in the SDI-12 Specification v1.3 except for the continuous measurement commands (`aR0`–`aR9` and `aRC0`–`aRC9`) and the concurrent measurement commands (`aC`–`aC9` and `aCC0`–`aCC9`). `M` and `R` command implementations are found on page 7.

Out of the factory, all METER sensors start with SDI-12 address `0` and print out the DDI serial startup string during the power-up time. This can be interpreted by non-METER SDI-12 sensors as a pseudo-break condition followed by a random series of bits.

The ATMOS 22 will omit the DDI serial startup string (sensor identification) when the SDI-12 address is nonzero. Changing the address to a nonzero address is recommended for this reason.

ATMOS 22 INTERNAL MEASUREMENT SEQUENCE

While powered up, the ATMOS 22 takes wind and air temperature measurements every 10 s and logs the values internally. Orientation is measured every 60 s and also logged internally. The `aR4!` command will output instantaneous measurements of these parameters and must be used at intervals of 10 s or greater for the response to be returned within the SDI-12 specification of 15 ms.

The `aM!`, `aR0!`, and `aR3!` commands (and subsequent `D` commands when necessary) will compute and output the averages or maximums of these measurements (and derived measurements) and reset internal averaging counters. Hence, it is not necessary to oversample the ATMOS 22 and compute averages and maximums in external data acquisition systems. Less frequent sampling has the additional benefit of decreasing data acquisition system and ATMOS 22 power consumption. **The `aR3!` command must be used at intervals of 10 s or greater for the response to be returned within the SDI-12 specification of 15 ms.**

SENSOR BUS CONSIDERATIONS

To use the ATMOS 22 in a bus configuration it is necessary to leave the data line idle for at least 6 s between commands being issued to the ATMOS 22, in addition to any other timing considerations. **Failure to leave the line idle can result in the ATMOS 22 outputting error codes instead of valid data.**

SDI-12 CONFIGURATION

Table 1 lists the SDI-12 communication configuration.

Table 1 SDI-12 communication configuration

Baud Rate	1200
Start Bits	1
Data Bits	7 (LSB first)
Parity Bits	1 (even)
Stop Bits	1
Logic	Inverted (active low)

SDI-12 TIMING

All SDI-12 commands and responses must adhere to the format in Figure 4 on the data line. Both the command and response are preceded by an address and terminated by a carriage return line feed combination and follow the timing shown in Figure 5.

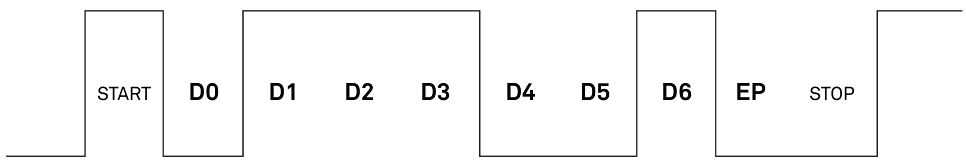

Figure 4 Example SDI-12 transmission of the character `1` (`0x31`)

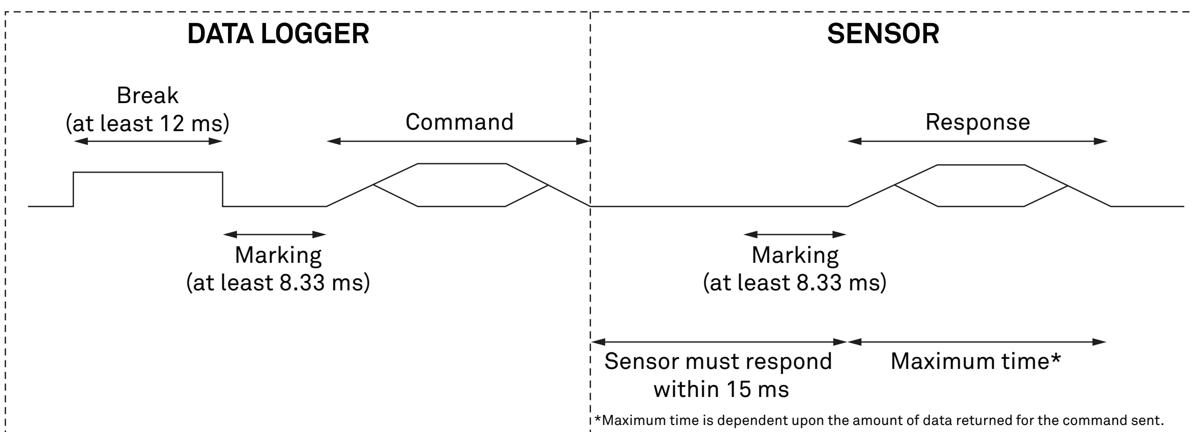

Figure 5 Example data logger and sensor communication

COMMON SDI-12 COMMANDS

This section includes tables of common SDI-12 commands that are often used in an SDI-12 system and the corresponding responses from METER sensors.

IDENTIFICATION COMMAND (`aI!`)

The Identification command can be used to obtain a variety of detailed information about the connected sensor. An example of the command and response is shown in Example 1, where the command is in **bold** and the response follows the command.

Example 1 **1I!**113METER␣␣␣ATM22␣100631800001

Parameter	Fixed Character Length	Description
1I!	3	Data logger command Request to the sensor for information from sensor address 1.
1	1	Sensor address Prepended on all responses, this indicates which sensor on the bus is returning the following information.
13	2	Indicates that the target sensor supports SDI-12 Specification v1.3
METER␣␣␣	8	Vendor identification string (METER and three spaces ␣␣␣ for all METER sensors)
ATM22␣	6	Sensor model string This string is specific to the sensor type. For the ATMOS 22, the string is ATM22␣.
100	3	Sensor version This number divided by 100 is the METER sensor version (e.g., 100 is version 1.00).
631800001	≤13, variable	Sensor serial number This is a variable length field. It may be omitted for older sensors.

CHANGE ADDRESS COMMAND (aAB!)

The Change Address command is used to change the sensor address to a new address. All other commands support the wildcard character as the target sensor address except for this command. All METER sensors have a default address of 0 (zero) out of the factory. Supported addresses are alphanumeric (i.e., a–z, A–Z, and 0–9). An example output from a METER sensor is shown in Example 2, where the command is in **bold** and the response follows the command.

Example 2 **1A0!**0

Parameter	Fixed Character Length	Description
1A0!	4	Data logger command. Request to the sensor to change its address from 1 to a new address of 0.
0	1	New sensor address. For all subsequent commands, this new address will be used by the target sensor.

ADDRESS QUERY COMMAND (?!)

While disconnected from a bus, the Address Query command can be used to determine which sensors are currently being communicated with. Sending this command over a bus will cause a bus contention where all the sensors will respond simultaneously and corrupt the data line. This command is helpful when trying to isolate a failed sensor. Example 3 shows an example of the command and response where the command is in **bold** and the response follows the command. The question mark (?) is a wildcard character that can be used in place of the address with any command except the Change Address command.

Example 3 **?!**0

Parameter	Fixed Character Length	Description
?!	2	Data logger command. Request for a response from any sensor listening on the data line.
0	1	Sensor address. Returns the sensor address to the currently connected sensor.

COMMAND IMPLEMENTATION

The following tables list the relevant Measurement (M) and Continuous (R) commands and subsequent Data (D) commands when necessary.

MEASUREMENT COMMANDS IMPLEMENTATION

Measurement (M) commands are sent to a single sensor on the SDI-12 bus and require that subsequent Data (D) commands are sent to that sensor to retrieve the sensor output data before initiating communication with another sensor on the bus.

Please refer to Table 2 and Table 3 for an explanation of the command sequence and see Table 8 for an explanation of response parameters.

Table 2 aM! command sequence

Command	Response
This command reports average or maximum values.	
aM!	atttn
aD0!	a+<windSpeed>+<windDirection>+<gustWindSpeed>
aD1!	a±<airTemperature>

NOTE: The measurement and corresponding data commands are intended to be used back to back. After a measurement command is processed by the sensor, a service request a <CR><LF> is sent from the sensor signaling the measurement is ready. Either wait until ttt seconds have passed or wait until the service request is received before sending the data commands. See the SDI-12 Specifications v1.3 document for more information.

Table 3 aM1! command sequence

Command	Response
This command reports instantaneous values.	
aM1!	atttn
aD0!	a±<xOrientation>±<yOrientation>+<nullValue>

NOTE: The measurement and corresponding data commands are intended to be used back to back. After a measurement command is processed by the sensor, a service request a <CR><LF> is sent from the sensor signaling the measurement is ready. Either wait until ttt seconds have passed or wait until the service request is received before sending the data commands. See the SDI-12 Specifications v1.3 document for more information.

CONTINUOUS MEASRUEMENT COMMANDS IMPLEMENTATION

Continuous (R) measurement commands trigger a sensor measurement and return the data automatically after the readings are completed without needing to send a D command.

The aR3! and aR4! commands must be used at intervals of 10 s or greater for the response to be returned within 15 ms as defined in the SDI-12 standard.

Please refer to Table 4 through Table 7 for an explanation of the command sequence and see Table 8 for an explanation of response parameters.

Table 4 aR0! measurement command sequence

Command	Response
This command reports average or maximum values.	
aR0!	a+<windSpeed>+<windDirection>+<gustWindSpeed>±<airTemperature>±<xOrientation>± <yOrientation>+<nullValue>±<NorthWindSpeed>±<EastWindSpeed>

NOTE: This command does not adhere to the SDI-12 response timing. See METER SDI-12 Implementation for more information.

Table 5 **aR1!** measurement command sequence

Command	Response
This command reports instantaneous values.	
aR1!	a±<xOrientation>±<yOrientation>±<nullValue>

NOTE: This command does not adhere to the SDI-12 response timing. See METER SDI-12 Implementation for more information.

Table 6 **aR3!** measurement command sequence

Command	Response
This command reports average or maximum values. Do not issue this command more frequently than every 20 s or else the ATMOS 22 measurements may be compromised. Please see ATMOS 22 Internal Measurement Sequence for more details.	
aR3!	a<TAB><NorthWindSpeed> <EastWindSpeed> <gustWindSpeed> <airTemperature> <xOrientation> <yOrientation> <nullValue><CR><sensortype><Checksum><CRC>

NOTE: This command does not adhere to the SDI-12 response format. However it does adhere to SDI-12 timing if it is sent at intervals ≥10 s. See METER SDI-12 Implementation for more information.

Table 7 **aR4!** measurement command sequence

Command	Response
This command reports instantaneous values.	
aR4!	a<TAB><NorthWindSpeed> <EastWindSpeed> <gustWindSpeed> <airTemperature> <xOrientation> <yOrientation> <nullValue><CR><sensortype><Checksum><CRC>

NOTE: This command does not adhere to the SDI-12 response format. However it does adhere to SDI-12 timing if it is sent at intervals ≥10 s. See METER SDI-12 Implementation for more information.

PARAMETERS

Table 8 Parameter Descriptions

Parameter	Unit	Description
±	—	Positive or negative sign denoting sign of the next value
a	—	SDI-12 address
n	—	Number of measurements (fixed width of 1)
nn	—	Number of measurements with leading zero if necessary (fixed width of 2)
ttt	s	Maximum time measurement will take (fixed width of 3)
<TAB>	—	Tab character
<CR>	—	Carriage return character
<LF>	—	Line feed character
<NorthWindSpeed>	m/s	Wind speed from the northerly direction (negative values denote southerly direction) (average since the last measurement or instantaneous value depending on SDI-12 command used)
<EastWindSpeed>	m/s	Wind speed from the easterly direction (negative values denote westerly direction) (average since the last measurement or instantaneous value depending on SDI-12 command used)
<windSpeed>	m/s	Combined wind speed magnitude of the <NorthWindSpeed> and <EastWindSpeed> (average since the last measurement or instantaneous value depending on SDI-12 command used)
<gustWindSpeed>	m/s	Maximum measured <windSpeed> since the last measurement

Table 8 Parameter Descriptions (continued)

Parameter	Unit	Description
`<windDirection>`	°	Wind heading clockwise from north reference (average since the last measurement or instantaneous value depending on SDI-12 command used)
`<airTemperature>`	°C	Air temperature (average since the last measurement or instantaneous value depending on SDI-12 command used)
`<xOrientation>`	°	X orientation angle (0 is level) (last measured value)
`<yOrientation>`	°	Y orientation angle (0 is level) (last measured value)
`<nullValue>`	—	This parameter is reported as 0. Previous firmware versions reported a compass heading, which has been removed.
`<sensortype>`	—	ASCII character denoting the sensor type For ATMOS 22, the character is the right square bracket] character
`<Checksum>`	—	METER serial checksum
`<CRC>`	—	METER serial 6-bit CRC

SERIAL CHECKSUM

These checksums are used in the continuous commands R3 and R4. The legacy checksum is computed from the start of the transmission to the sensor identification character, excluding the sensor address.

Legacy checksum example input is `<TAB>0.26 1.27 0.37 23.1 3.2 4.8 0<CR>\Hg` and the resulting checksum output is H.

```
uint8_t LegacyChecksum(const char * Response)
{
      uint16_t length;
      uint16_t i;
      uint16_t sum = 0;

      // Finding the length of the response string
      length = strlen(response);

      // Adding characters in the response together
      for( i = 0; i < length; i++ )
      {
          sum += response[i];

          if(response[i] == '\r')
          {
              // Found the beginning of the meta data section of the response
              break;
          }
      }

      // Include the sensor type into the checksum
      sum += response[++i];

      // Convert checksum to a printable character
      sum = sum % 64 + 32;

      return sum;
}
```

The more robust CRC6, if available, utilizes the CRC-6-CDMA2000-A polynomial with the value 48 added to the results to make this a printable character and is computed from the start of the transmission to the legacy checksum character, excluding the senor address.

CRC6 checksum example input is `<TAB>0.26 1.27 0.37 23.1 3.2 4.8 0<CR>\Hg` and the resulting checksum is the character g.

```
unit8_t CRC6_Offset (const char *buffer)
{
      uint16_t byte;
      uint16_t i;
      uint16_t bytes;
      uint8_t bit;
      uint8_t crc = 0xfc;  // Set upper 6 bits to 1's

     // Calculate total message length -- updated once the meta data section is found
     bytes = strlen(buffer);

      // Loop through all the bytes in the buffer
      for(byte = 0; byte < bytes; byte++)
      {
       // Get the next byte in the buffer and XOR it with the crc
       crc ^= buffer[byte];

          // Loop through all the bits in the current byte
          for(bit = 8; bit > 0; bit--)
          {
              // If the uppermost bit is a 1...
              if(crc & 0x80)
              {
                  // Shift to the next bit and XOR it with a polynomial
                  crc = (crc << 1) ^ 0x9c;
              }
              else
              {
                  // Shift to the next bit
                  crc = crc << 1;
              }
          }

          if(buffer[byte] == '\r')
          {
             // Found the beginning of the meta data section of the response
             // both sensor type and legacy checksum are part of the crc6
             // this requires only two more iterations of the loop so reset "bytes"
             // bytes is incremented at the beginning of the loop, so 3 is added
             bytes = byte + 3;
          }
      }

      // Shift upper 6 bits down for crc
      crc = (crc >> 2);

      // Add 48 to shift crc to printable character avoiding \r \n and !
      return (crc + 48);
}
```

CUSTOMER SUPPORT

Customer service representatives are available for questions, problems, or feedback Monday through Friday, 8 am–5 pm Pacific time.

Email: support.environment@metergroup.com
sales.environment@metergroup.com

Phone: +1.509.332.5600

Fax: +1.509.332.5158

Website: metergroup.com

If contacting METER by email, please include the following information:

Name
Address
Phone number
Email address
Instrument serial number
Description of problem

NOTE: For ATMOS 22 Ultrasonic Anemometers purchased through a distributor, please contact the distributor directly for assistance.

REVISION HISTORY

The following table lists document revisions.

Revision	Date	Compatible Firmware	Description
02	1.6.2018	1.03	Removed compass output Removed 20 s limitation of `aM!`, `aR0!`, and `aR3!`
01	11.15.2017	1.00	Added information for DS-2 update to ATMOS 22
00	10.27.2017	1.00	Initial release