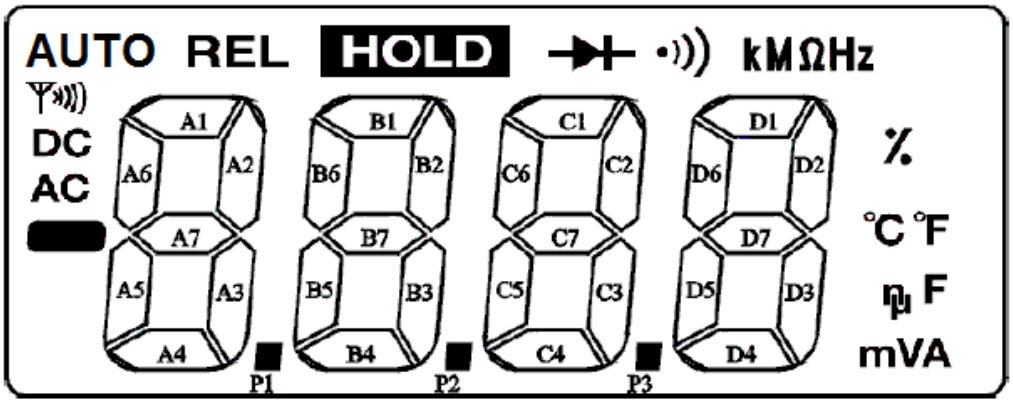
4000-COUNT DMM COMMUNICATIONS PROTOCOL

After connecting the meter and PC computer via interface cable & after initiating the COM port, the meter will automatically output 14-byte data of which format is shown below. The output rate is equal to the LCD display update rate - normally 3 times per second.

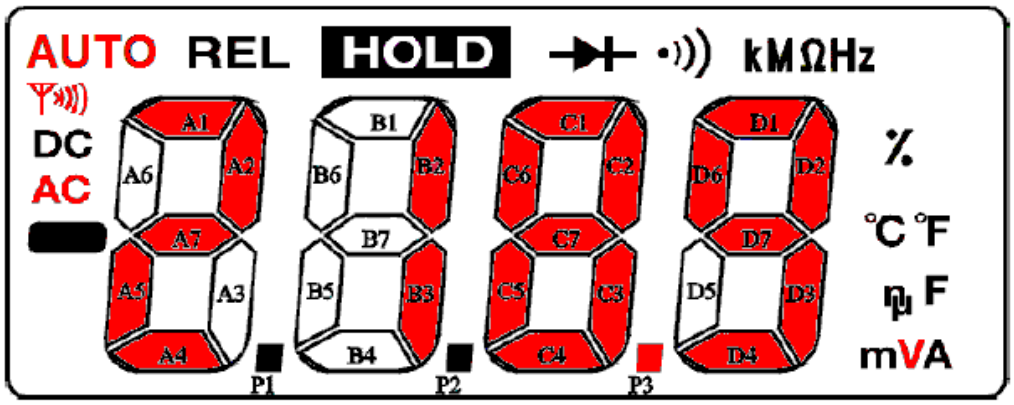
BAUD RATE 2400, N, 8, 1



Output Data set	Bit 7 ~ Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1st byte	0001	AC	DC	AUTO	
2nd byte	0010		A5	A6	A1
3rd byte	0011	A4	A3	A7	A2
4th byte	0100	P1	B5	B6	B1
5th byte	0101	B4	B3	B7	B2
6th byte	0110	P2	C5	C6	C1
7th byte	0111	C4	C3	C7	C2
8th byte	1000	P3	D5	D6	D1
9th byte	1001	D4	D3	D7	D2
10th byte	1010	u	n	k	
11th byte	1011	m	%	M	
12th byte	1100	F	Ω	REL	HOLD
13th byte	1101	A	V	Hz	
14th byte	1110			°C	°F

The QM1571 uses a wireless interface which connects to the computer's USB port. A USB driver must be installed and automatically allocates a COM port for each wireless interface connected. The COM port allocated can be changed using the Device Manager. The QM1571 outputs a 5 byte header which can be ignored; the remaining 14 bytes of data are as shown above. The main chip in the QM1571 is the Semic CS7721CN (this is also used in some other meters). An alternate chip may be used in some meters - the Fortune Semiconductor Corp FS9721-LP3 device.

Example of 14 byte data string when the meter reads - "AC 218.9V auto-ranging"…



Output Data set	Bit 7 ~ Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	*HEX format*
1st byte	0001	1	0	1	1	1BH
2nd byte	0010	0	1	0	1	25H
3rd byte	0011	1	0	1	1	3BH
4th byte	0100	0	0	0	0	40H
5th byte	0101	0	1	0	1	55H
6th byte	0110	0	1	1	1	67H
7th byte	0111	1	1	1	1	7FH
8 th byte	1000	1	0	1	1	8BH
9th byte	1001	1	1	1	1	9FH
10th byte	1010	0	0	0	0	A0H
11th byte	1011	0	0	0	0	B0H
12th byte	1100	0	0	0	0	C0H
13th byte	1101	0	1	0	0	D4H
14th byte	1110	0	0	0	0	E0H

The full output string in hexadecimal is:

XX-XX-XX-XX-XX-1B-25-3B-40-55-67-7F-8B-9F-A0-B0-C0-D4-E0

The first five bytes can be ignored. XX = don’t care.

To extract each digit, the 3 segment bits from the 2nd, 4th, 6th or 8th byte are combined with 4 segment bits from the next byte to produce a number between 0 and 127. The number can be used in a series of "if" statements or by a look-up table array, where the digit 0 = 7DH (125), 1 = 05H (5), 2 = 5BH (91), 3 = 1FH (31), 4 = 27H (39), 5 = 3EH (62), 6 = 7EH (126), 7 = 15H (21), 8 = 7FH (127) and 9 = 3FH (63).