PVC*uni* and PVC*duo*

Process and Vacuum Controllers

COMMUNICATIONS HANDBOOK

- **MODBUS-RTE Protocol and Parameter Address Listing**. Read/write of multiple parameters per message. Floating point data transfer
- **QueBus Protocol and Parameter Mnemonic Listing**. Easy to use ASCII-based protocol. Read/write of multiple parameter per message. Various error checking options.

Table of Contents

1 Serial Communications

*Computer control over the RS232/RS485 bus of all aspects of controller operation is available via 2 different protocols: a binary **MODBUS RTE**-based protocol and **QueBUS,** an ASCII-based protocol . Both provide fast data throughput, including reading and writing of multiple parameters with a single message exchange*

1.1 Introduction

Controller parameters are accessible over the built-in RS232 and 485 serial communications interfaces. 2 protocol types are available:

- A robust binary-level **MODBUS** RTE-based protocol offering high speed, data throughput and integrity. Multiple sequential 32 bit parameters can be read and written in each data exchange allowing floating point resolution data transfer. Little and Big Endian data formats are supported.
- **QueBUS** is an easy to use ASCII-based protocol. Multiple, arbitrarily-selected parameters can read/written to in each data exchange. Various levels of error checking are available to simplify development.

In addition, VacTools freeware provides a stand-alone graphical application for monitoring and controlling vacuum systems via multiple PVC controllers with full data logging and data presentation. VacTools can interface with third party user software via the vtde API.

1.2 Hardware

- RS232 and RS485 (3 wire) interfaces are included as standard. Both are "multi-drop" allowing several controllers to share the same port.
- Two serial port RJ45 sockets allow ease of daisy-chaining.
- Baud rate: 2400, 4800, 9600, 19200, 38400, 57600.
- Parity options: None, Even, Odd.
- 8 bits, 1 start bit, 1 stop bit.
- Address range 01 to 99.

2 MODBUS protocol implementation

2.1 Introduction

For more information on the industry standard MODBUS protocol, please refer to the "MODBUS Application Protocol Specification" available on-line from MODBUS.ORG.

MODBUS is a powerful, multi-application protocol. The implementation of MODBUS used in the PVC*uni* and PVC*duo* controllers is selected to match transfer of the 32 bit parameters underpinning the firmware; as such, many of the features of MODBUS (such as register and coil transfers) are not used. Key features of the implementation:

- Only one Function Code - hexadecimal 17h (decimal 23) - is used. Function code 17h allows both writing and reading of multiple sequential registers within a single message transaction. No other function codes ("coils" etc) are implemented.
- To facilitate floating point access, **all** controller parameters (including integer and flag parameters) are 32 bit, i.e. take up 2 MODBUS registers (2x 16bit MODBUS register WORDs, or 4x 8 bit BYTES). **Thus, all controller parameter addresses are even (i.e. address 0, 2, 4, 6…)**
- Data transfer can be set to **LITTLE ENDIAN** (least significant byte first) or **BIG ENDIAN** (most significant byte first). In the examples given, LITTLE ENDIAN representation is assumed, unless stated differently.
- Up to 16 parameters can be read and/or written within a single message.
- The "INVALID" data value allows parameters in the middle of a multiple write command to be ignored.

2.2 Messages and Frames

Communication from the PC/PLC (the client) to the controller (the server) is ALWAYS initiated by the client sending a message. The controller CANNOT initialize a communication.

A MODBUS message sent by the client or server is referred to as a "Frame". It comprises:

Frame start	Device Address	Function Code	Data Package	CRC	End Indicator
(3.5 bytes)	1 byte	1 byte	n bytes	2 bytes	(3.5 bytes)
	01h-63h (1-99)	17h (decimal 23)			

- Frame start is assumed when there is no bus activity for more than 3.5 times the time required to transmit an 8-bit BYTE. Once started, messages MUST be sent contiguously, i.e. without interruption of the data flow.
- The first byte is the address 1-99 (01h-63h) of the controller.
- The second byte is the function code, and is ALWAYS 17h.
- The Data Package containing the parameter addresses to be written/read, the data and the amount of data.
- The 2 BYTE Cyclical Redundancy Check (CRC) is a highly efficient method of error detection and MUST be included.

2.3 The Structure of the CRC

The CRC detects transmission errors, offering 100% detection for single-, double- and odd-numbered bit errors, 100% on burst errors of 16 bits or less and >>99.9969% detection on burst errors in excess of 16 bits. ***If the controller device fails to verify the CRC, no reply is sent***.

The CRC is performed on all bytes within the frame and is appended to the message. The algorithm used to generate the CRC is:

1. Create a 16 bit (WORD) CRC register and load with FFFFh.
2. Exclusive OR the first byte with the CRC register.
3. Shift the CRC register one bit to the right.
4. If the over-flow bit is set, exclusive OR the CRC register with A001h. If not, no operation.
5. Repeat 3 and 4 until 8 shifts have been performed.
6. Repeat steps 2 to 6 until all bytes have been exclusive OR'ed and shifted 8 times.
7. Append the lower BYTE of the CRC register to the message
8. Append the upper BYTE of the CRC register to the message.

The following 'C' code function calculates the CRC for nMessLen characters of the message lpMess:

```
// Generates crc check bytes and places these in lpCRC
```

```
// The initial message is pointed to by lpMess and the total message
//  length (including address and function code bytes) are indicated
//  in nMessLen
int zet_calc_modbuscrc
   (unsigned char* lpMess, int nMessLen, unsigned char* lpCRC)
{
 int                      j;
 unsigned short int       wCarry, wNext, wCRC;
 // Initialise the CRC register…
 wCRC = 0xFFFF;
 while (nMessLen--)
 {
  // Cast the byte to a 16 bit value…
  wNext = (unsigned short)lpMess;
  // Exclusive OR it with the CRC register…
  wCRC ^= wNext;
  // Shift loop 8 times…
  for (j=0; j<8; j++)
  {
   // Save bottom bit…
   wCarry = wCRC & 0x0001;
   // Shift CRC register to the right by 1 bit…
   wCRC >>= 1;
   // If the saved bottom bit was set, EXOR with A001h…
   if (wCarry)
    wCRC ^= 0xA001;
  }
  // Next byte…
  lpMess++;
 }
 // Set upper byte of CRC register to second CRC byte…
 lpCRC[1] = (unsigned char)(wCRC >> 8);
 // Set the low byte of the CRC register to the first CRC byte…
 lpCRC[0] = (unsigned char)(wCRC);
 // Return the number of extra bytes…
 return 2;
}
```

As an example of using such a function, to append the CRC to a message:

```
 unsigned char      sMess[128];  // buffer for the message
 int         nMessLen;    // Stores the message length so far
 …
 … Code to generate the message
 …
 nMessLen += zet_calc_modbuscrc(sMess, nMessLen, &sMess[nMessLen]);
 // The nMessLen parameter will have been extended by 2 to indicate
 //  addition of 2 bytes
 …
 … Send the message
 …
```

To use the function to verify a CRC message…

```
 unsigned char      sMess[128];  // buffer for the message
 unsigned char      sCRC[2];     // buffer for the CRC
 int         nMessLen;    // Stores the message length received (including the returned CRC)
 …
 … Read the message into sMess
 … Store the number of character read in nMessLen
 … Perform any checks (such as returned unit address and function code, is length valid for
the number of parameters sent? etc.)
 …
 zet_calc_modbuscrc(sMess, nMessLen-2, sCRC);
 // Note use of nMessLen-2 since we do not want to include
 //  received back CRC in the CRC generation process
 if ((sMess[nMessLen-2] == sCRC[0]) && (sMess[nMessLen-1] == sCRC[1]))
 {
  // SUCCESS!!!!
 }
 else
 {
  // ERROR PROCESSING!!!!
 }
```

2.4 Data Package: Message Structure from the Client (the Command)

A communication is initiated by the PC/PLC sending data to and/or requesting data from the controller. Note: when constructing the data package, **BYTE order must correspond to the selected ENDIAN representation - see section 2.5.1**.

2.4.1 Reading Data:

After the address and function code, 4 BYTES indicate what data is to be **read**:

Address of first Parameter to be read		Number of Registers to be read	
Upper byte	Lower Byte	Always 00h	yyh

Note:

- Since parameters use 2 registers, the number of registers to be read is twice the number of parameters and is therefore also ALWAYS even.
- If no bytes are to be read, all 4 bytes MUST be included and set to zero.

2.4.2 Writing Data:

The above is followed by any data to be written to the controller. The first 2 BYTES of this section are the Address of the first Parameter to be written to. This is followed by 2 BYTES for the number of registers (twice the number parameters) to be written, and 1 BYTE for the number of bytes of data to follow. [Note that in this implementation of MDOBUS, this means that the number of bytes value is always twice the number of registers value.] Finally, the data BYTES are appended.

Address of first Parameter to be written to		Number of Registers to write		Number of data Bytes (n)	Data
Upper byte	Lower byte	Always 00h	yyh	1 BYTE	*n* BYTES

Note:

- Since parameters use 2 registers, the number of registers to be written is twice the number of parameters and is therefore also ALWAYS even.
- If no Parameters are to be written, all five BYTES MUST still be included in the data package but all set to zero.
- The number of data bytes value will ALWAYS be four times number of parameters (or 2 times the number registers) that are to be written.

2.4.3 The "Invalid" Parameter Value - FFFFFFFFh

When sending multiple parameters, any parameters that are **NOT** to be changed can be should with the "Invalid" parameter value of FFFFFFFFh. This tells the controller to ignore this parameter data; by design, no parameter has this as a "valid" value.

2.4.4 Data Package: Message Structure From the Server (the Response)

On receiving a transmission:

- The controller checks the first byte against its own address. If this agrees...
- Checks the next BYTE is Function Code 23 (17h). If not, sends error message (section 2.4.5). If it agrees...
- Checks the integrity of the message by verifying the received CRC value. If this agrees...
- Looks to see if any parameters are to be written. If so, ignores any parameter values set to the invalid parameter value FFFFFFFFh (see 2.4.3). If an invalid value for a parameter is received, an error message is generated - see section 2.4.5.
- When all parameters have been written, looks to see if any parameters are to be read - these can be the same or different parameters to those written. If so, these are compiled and appended to the response.
- Calculates a CRC.

A message is then returned that comprises:

Device Address	Function Code	Num of Data BYTES being returned	Data	CRC
1 BYTE	1 BYTE	1 BYTE	*n* BYTES	2 BYTES
01h-63h (1-99)	17h (decimal 23)	(0,4,8,12...)		

Note:
- The Client programme should check the address corresponds to that sent and that the second BYTE is 17h, and validity of the returned CRC.
- Since each Parameter comprises of 2x 16 bit registers, each which is 2x 8 bit BYTES, the number of data BYTES value is 0 or a multiple of 4. This value should be four times the number of parameters requested.
- Each parameter comprises of 4 BYTES in the Data section.
- The returned BYTE order depends on the selected ENDIAN representation 2.5.1.

2.4.5 Error Response

An error response message comprises 5 BYTES:

Device Address	Function Code + 80h	Error Code	CRC	
1 BYTE	97h	01h or 02h	MSB	LSB

The first BYTE is the address, followed by the (Function Code + 80h) i.e. a decimal value of 151. A single error code value follows and the message terminates in the CRC. The error codes are:

Code	Description
01h	Invalid Function Code, i.e. function code BYTE was NOT 17h
02h	Invalid Parameter Address or Value

2.5 Parameter Types

2.5.1 BIG and LITTLE ENDIAN Data

All parameters within the controller are 32 bit, i.e. take up 4 bytes. When transferring data, between computer and controller, the 4 bytes can be transferred with the most significant byte first (BIG endian) or with the least significant byte first (LITTLE endian). For example, the order for transmission of the 32 bit hexadecimal value 12345678h would be:

BIG endian: 12h (first), then 34h, then 56h then 78h
LITTLE endian: 78h (first), then 56h, then 34h, then 12h

When interpreting multi-byte values transferred using binary protocols (such as MODBUS), it is important that both parties use the same data transfer type (endianism) to ensure successful reconstruction of the data. For example, if different transfer types are used, the message "Hello" would be interpreted as "olleH"!

The PVC provides the ability to transfer data either as BIG or LITTLE endian, as selected in the ***Protocol*** Parameter. Nevertheless, the user still needs to reconstruct the original value. Various methods, depending on language, are available for transferring the to the correct value, for example:

To RECEIVE a 32 bit floating point number returned in a byte array pdata:

```
// define a float and pointer to a type byte...
float               fVal;  // 32 bit floating point
unsigned char*      p;

// set the pointer to fVal...
p = (unsigned char*)&fVal;

if (BIG_ENDIAN)
{
 *pS = *(pData+3);
 *(pS+1) = *(pData+2);
 *(pS+2) = *(pData+1);
 *(pS+3) = *pData;
}
else // LITTLE_ENDIAN
{
 *pS = *pData;
 *(pS+1) = *(pData+1);
 *(pS+2) = *(pData+2);
 *(pS+3) = *(pData+3);
}
```

To SEND a 32 bit integer value in a variable nVal, set it into a byte array pointed to by pdata:

```
int         nVal = the_data_to_transmit;
// define a pointer to a type byte...
unsigned char*      p;

// set the pointer to nVal...
p = (unsigned char*)&nVal;

if (BIG_ENDIAN)
{
 *(pData+3) = *ps;
 *(pData+2) = *(pS+1);
 *(pData+1) = *(pS+2);
 *pData = *(pS+3);
}
else // LITTLE_ENDIAN
{
 *pS = *pData = *pS;
 *(pData+1) = *(pS+1);
 *(pData+2) = *(pS+2);
 *(pData+3) = *(pS+3);
}
```

2.5.2 String Parameters

As memory locations are 32 bit, these can store up to 4 eight-bit ASCII characters. For example the *Unit ID* parameter is a 4 character parameter, and gauge names are each 3 characters.

2.5.3 Integer Parameters

These store 32 bit integer values. They can either be signed integers (-2147483648 to +2147483647) or unsigned integer types (0 to 4294967295).

2.5.4 Floating Point Parameters

These hold a 32 bit floating point value in standard IEEE format (as employed in most high level languages):

SIGN	EXPONENT	FRACTION
Bit 31	Bit 30 to Bit 23	Bit 22 to Bit 0
±	2^{7} to 2^{0}	2^{-1} to 2^{-23}

2.5.5 Composite Parameters

Composite Parameter divide a 32 bit unsigned integer parameter into nibbles (4 bit) and/or byte (8 bit) sections, each section storing a particular facet of the parameter. For example, the *Trip Settings* parameter holds information about its allocation, power-up condition and operating state. The highest bit in each section is used for validation (VALID bit). If the parameter is read, the upper bit will be set. If the parameter is being written, the VALID bit for that section must be set to instruct the controller to write the new value into the section; otherwise the new value will be ignored. This method allows the user to only write those parts of the composite parameter that require being changed.

For example, the *Ion Gauge Settings* Parameter (section 2.8) has several nibble sections (filament control and input filter) and a byte wide section for Emission set value. The highest bit in each section is the validation bit; it indicates that a valid value is present for reading, and it needs to be set to indicate to the controller that the value needs to be replaced.

Referring to the *Ion Gauge Settings* parameter, the filament number section occupies nibble 000**X**0000h; the value 00080000h is the validation bit, with the bits 00070000h representing the selected data value. Thus, if reading this parameter, you receive 008**9**B080h, the value 9 indicates that the filament number VALID flag is set and the value is 1 (which means always use filament 1).

To change this value to use filament auto-selection, send 00080000h. This tells the controller that the filament number VALID flag is set, so the value (0 = auto-select) is to be set. ***More importantly, by sending 0 in all the other nibbles, it tells the controller NOT to change those section values***.

2.6 Parameter Listing

The controller has 256 off 32-bit memory (parameter) locations. These are directly accessed by the MODBUS protocol where the the MODBUS parameter address is 2x the memory location. Thus all MODBUS addresses are even.

The following table lists and describes the MODBUS addresses for accessing parameters:
MODBUS Parameter Address gives the decimal and hexadecimal value of the parameter address.
Parameter Name relates the parameter to its function.
Type indicates whether the parameter is an integer, composite, string or floating point type. Also indicated is whether the parameter can be read only ('R'), can be written to as well as read ('W') or can only be changed from the unit front panel ('M').
Description provides a detailed breakdown of the parameter function.

2.7 System Global, Communications Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Address	*Name*		Description
0/000h	*Unit ID*	IR	**UNI**: 75435650h [little endian ASCII representation of "PVCu"). **DUO**: 64435650h (little endian ASCII representation of "PVCd")
2/002h	*Firmware Version*	IR	**UNI**: 7569***mmnn***h **DUO**: 646F***mmnn***h where ***mm*** is major version and ***nn*** is minor revision
6/006h	*Comms Address*	IM	Unit Communications Address. Range 01 to 99. Used by both MODBUS and QueBUS
8/008h	*Comms Settings*	CM CM CM	**BAUD: X**0000000h. 8h VALID flag OR'ed with: 0h = 4800; 1h = 9600; 2h = 19200; 3h = 38400; 4h = 57600 **PARITY**: 0**X**000000h. 8h VALID flag OR'ed with: 0h = None; 1h = ODD; 2h = EVEN **PROTOCOL**: 00000**X**00h. 8h VALID flag OR'ed with: 0h = MODBUS Little Endian; 1h = MODBUS Big Endian; 2h = QueBUS no error checking; 3h = QueBUS with Check Sum; 4h = QueBUS with CRC
10/00Ah	*Unit Name*	SW	3 character unit name.
14/00Eh	*General Status*	CR CR CR CR CR	**FAN**: 0**X**000000h. 8h VALID flag OR'ed with any combination of: 1h = Fan Error has occurrred; 2h = Fan is on (full speed); 4h = Fan is on (slow speed) **HIGH VOLTAGE STATUS**: 00**X**00000h. 8h VALID flag OR'ed with combination of: 1h = No High Voltage; 2h = ion gauge 1 filament potential failed; 4h = ion gauge filament potential failed. **EMISSION OVER-CURRENT**: 000**X**0000h. 8h VALID flag OR'ed with combination of: 1h = ion gauge 1 over-current; 2h = ion gauge 2 over-current **A/D STATUS**: 0000**X**000h. 8h VALID flag OR'ed with: 1h if A/D converter has failed **UNIT TEMPERATURE STATUS**: 00000**X**00h. 8h VALID flag OR'ed with: 1h = Under temperature error; 2h = Over temperature error.
18/012h	*User ID*	IW	Can be used as required by user as identification or unit data storage. Stores a number between 0 and 999999.
36/024h	*Display Settings*	CW CW CR	**COMMUNICATIONS INDICATOR**: 000**X**0000h. 8h VALID flag OR'ed with: 1h = On **DISPLAY SAVER**: 0000**X**000h. 8h VALID flag OR'ed with: 1h = On **STATUS MODE PAGE**: 000000**XX**h. 80h VALID flag OR'ed with: 0h = General Text Page; 1h = Bar Graph Page; 2h = Ion gauge 1 Large Page; 3h = Ion gauge 2 large page (duo only); 4h = Both ion gauges page (duo only); 5h = Modules large page; 6h = Bake-out page; 7h = Trips and digital input page; 8h = Timers page
38/026h	*Display Saver Time*	IW	1 to 999 minutes
44/02Ch	*Unit Operating Time*	IR	Number of hours the unit has been operating
46/02Eh	*Menu Escape Time*	IW	20 to 999 seconds
64/040h	*Global Settings*	CW CW CM	**BARGRAPH MAX EXPONENT**: **XX**000000h. 80h VALID flag OR'ed with exponent indicator (EI). EI is calculated by adding decimal 16 to the exponent value and converting to hex. For example, for 1e+3: 3 + 16 = 19 = 13h. Range is 1e-9 to 1e+6 (values of 07h to 16h). **BARGRAPH MIN EXPONENT**: 00**XX**0000h. 80h VALID flag OR'ed with exponent indicator (EI). EI is calculated by adding decimal 16 to the exponent value and converting to hex. For example, for 1e10: -10 + 16 = 6 = 06h. Range is 1e-16 to 1e0 (values of 00h to 10h). **PARAMETER UNLOCK**: 0000**X**000h. 8h VALID flag OR'ed with: 0h = Locked; 1h Unlocked. Note: this parameter protects calibration parameters so only manually accessible **PRESSURE UNITS**: 0x000000**X**0h. 8h VALID flag OR'ed with: 0h = mBar; 1h = Torr; 2h = Pascal.
186/0BAh	*Heatsink Temperature*	FR	Measured heat-sink temperature

Address	*Name*		Description
292/124h	*Internal Temp Power*	FR	[Diagnostics]
294/126h	*Internal Temperature*	FR	[Diagnostics]

2.8 Ion Gauge Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Ion Gauge 1	Ion Gauge 2	*Name*		Description
16/010h	12/00Ch	*Ion gauge name*	IW	3 character name for the ion gauge
24/018h	416/1A0h	*Degas Ramp Time*	IW	1 to 499 minutes
26/01Ah	418/1A2h	*Degas Hold Time*	IW	1 to 499 minutes
28/01Ch	420/1A4h	*Auto-degas level*	CW	**EMISSION**: 000000**XX**h. 80h VALID flag OR'ed with: 0Dh = Degas Low, 0Eh = Degas Medium, 0Fh = Degas High. (See ion gauge emissions parameter for details)
40/028h	422/1A6h	*Degas Remaining Time*	IR	Number of minutes remaining
42/02Ah	424/1A8h	*Ion gauge operating time*	IW	Number of hours the ion gauge has been operating This parameter either reads the time or resets the time to zero (e.g. after filament change)
96/060h	428/1ACh	*Minimum ion gauge emission*	CW	**EMISSION**: 000000**XX**h. 80h VALID flag OR'ed with a value between 01 (0.06mA) and the current Maximum ion gauge emission value. (See ion gauge *Settings* parameter for details)
98/062h	430/1AEh	*Maximum ion gauge emission*	CW	**EMISSION**: 000000**XX**h. 80h VALID flag OR'ed with a value between current Minimum ion gauge emission value and 0Ch (10mA). (See ion gauge *Settings* parameter for details)
100/064h	432/1B0h	*Minimum emission power*	IW	Range is 0 to current setting of Maximum power value
102/066h	434/1B2h	*Maximum mission power*	IW	Range is current setting of Minimum power value to 99
104/068h	436/1B4h	*Emission PID Prop Band*	IW	1 to 99AU. Default value is 25.
106/06Ah	438/1B6h	*Emission PID Damp (ID)*	IW	1 to 99AU. Default value is 25.
126/07Eh	414/19Eh	*Status2*	CR	**INHIBIT STATUS**: 000000**XX**h. 80h VALID flag OR'ed with: 2h = Digital input inhibited
136/088h	412/19Ch	*Status*	CR	**LAST ERROR**: **XX**000000h. Reports why previous ion gauge operation failed. The data is cleared at next attempt to start the ion gauage. 80 VALID flag OR'ed with combination of: 01h = filament drive failed; 02h = emission control failed; 04h = ion gauge interlock failed (cable unplugged); 08h = maximum power exceeded during normal operation; 10h = Overpressure detected; 20h = Failed due to digital input interlock; 40h = Failed due to fan error operation detected.
			CR	**HIGH VOLTAGE FAIL**: 00**X**00000h. 8h VALID flag OR'ed with 1 if failed
			CR	**MINLIMIT PRESSURE**: 0000**X**000h. 8h VALID flag OR'ed with 1 if measured pressure is below measurement limit
			CR	**CURRENT EMISSION**: 000000**XX**h. 80h VALID flag OR'ed with currently operating emission value. (See ion gauge settings parameter for details).

Ion Gauge 1	Ion Gauge 2	***Name***		**Description**
140/08Ch	408/198h	*Settings2*	CW	**DECIMAL PLACES**: 00**X**00000. 8h VALID flag OR@ed with 1 for 2 decimal places, else 1 decimal place
			CW	**GAUGE UNITS**: 000**X**0000h. Determines whether to use pressure units for the gauge or measured current options. 8h VALID flag OR'ed with: 0h = Measured values are in currently selected pressure units; 1h = Direct measurement of the current; 2h = Measurement of the current normalized to 10mA equivalent
			CW	**DI4 INTERLOCKING**: 0000**X**000h. 8h VALID flag OR'ed with: 1h = Interlocked. Ion gauge trips off if the DI fails; 2h = Emission On. Ion gauge remains on for as long as DI4 is on; 4h = Emission Toggle. Ion gauge toggles on and off after pulse - see section xxx
			CW	**DI3 INTERLOCKING**: 00000**X**00h. see DI4 values above
			CW	**DI2 INTERLOCKING**: 000000**X**0h. see DI4 values above
			CW	**DI1 INTERLOCKING**: 0000000**X**h. see DI4 values above
142/08Eh	410/19Ah	*Settings*	CW	**FILAMENT POWER-UP RAMP**: 00**X**00000h. 8h VALID flag OR'ed with: 0h = Standard Ramp, 1h = Slow Ramp, 2h = Fast Ramp, 3h = Learn Start-up Power
			CW	**FILAMENT NUMBER**: 000**X**0000h. 8h VALID flag OR'ed with: 0h = Auto; tries filament 1 first and if fails tries filament 2; 1 = Filament 1; 2 = Filament 2
			CW	**INPUT FILTER**: 0000**X**000h. 8h VALID flag OR'ed with a value between 0 and 7. Factory set to 4 - should not need adjusting. Please also see *Low Pass Filter* parameter
			CW	**EMISSION SET-POINT**: 000000**XX**h. 80h VALID flag OR'ed with: 00h = OFF; 01h = 0.06mA; 02h = 0.1mA; 03h = 0.15mA; 04h = 0.25mA; 05h = 0.4mA; 06h = 0.6mA; 07h = 1mA; 08h = 1.5mA; 09h = 2.5mA; 0Ah = 4mA; 0Bh = 6mA; 0Ch = 10mA; 0Dh = Degas Low; 0Eh = Degas Medium; 0Fh = Degas High; 10h = Auto-Emission. Note: set values must conform to operational conditions such as Min and Max Emissions settings and degas start-up restrictions.
150/096h	392/188h	*Setpoint Emission Value*	FR	The current setpoint emission current in mA
152/098h	394/18Ah	*Measured Emission Value*	FR	The current measured emission current in mA
154/09Ah	396/18Ch	*Measured Value*	FR	The current measured value. If the ion gauge is set to display as current, the measured value is in amps; otehrwise the value is in the currently selected pressure units
156/09Ch	398/18Eh	*Gauge Sensitivity*	FW	The gauge sensitivity in /mBar. Range 0.1 to 99.9 /mBar
158/09Eh	400/190h	*Low Pass Filter*	FW	Pressure measurement filter. Range 0.1 to 9.0 seconds
188/0BCh	402/192h	*Gas Sensitivity*	FW	Gas senstivity factor. Range 0.01 to 99.00. Default is 1.00 for Nitrogen.
190/0BEh	382/17Eh	*Emission Start Power*	FW	The initial power value ramped to on ion gauge start-up. Range is *Minimum Power* to *Maximum Power* parameters
204/0CCh	406/196h	*Degas Suspend Limit*	FW	Before Degas: The measured pressure must be below this value to enabled starting degas. During Degas: If the measured pressure rises above this value during degas ramp time, the emission ramp is suspended until the pressure recovers
206/0CEh	404/194h	*Filament Power Value*	FR	The output filament drive power. Range: 0 to *Maximum Power*.

2.9 Dual (wide-range) Gauge Mode Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Ion Gauge 1	Ion Gauge 2	***Name***		**Description**
68/044h	440/1B8h	*Settings/Status*	CW	**ENABLE**: **X**0000000h. 8h VALID flag OR'ed with: 0h = Disabled; 1h = Enabled. Note: If no secondary gauge is assigned, this parameter will revert to disabled. See also power-up options.
			CW	**SECONDARY GAUGE PROTECT**: 0**X**000000h. 8h VALID flag OR'ed with: 0h = Protection off; 1h = Protection on
			CW	**SECONDARY GAUGE ASSIGNMENT**: 0x00**X**00000. 8h VALID flag OR'ed with: 0h = off; 1h = Slot 1 gauge assigned; 2h = Slot 2 gauge assigned.
			CW	**POWER UP OPTIONS**: 0x000**X**0000h. 8h VALID flag OR'ed with: 0h = Wide range operation is disabled on power up; 1h = If enabled at power down, wide range operation starts immediately on power up; 2h = Allow wide-range operation once only - once ion gauge started, wide-range operation is disabled
			CR	**STATUS**: 0x00000**XX**0h. 80h VALID flag OR'ed with combination of: 01h = Wide range operation in progress; 02h = In delay period before ion gauge start; 04h = Ion gauge starting; 10h = Digital input inhibited; 20h = Wide range operation is inhibited due to disable state
			CR	**NUMBER OF FAILED ATTEMPTED STARTS**: 0 to 9.
70/046H	442/1BAh	*Dual Gauge Emission*	CW	**ION GAUGE EMISSION VALUE**: 000000**XX**h. 80h VALID Flag OR'ed with: Required operating emission value (see ion gauge *Settings* emission values). The value can be set to any fixed value between the *Minimum Emission* and *Maximum Emission* parameters, or to Auto-Emission
192/0C0h	384/180h	*Ion Gauge ON Pressure*	FW	The secondary gauge pressure at which to attempt to start the ion gauge.
194/0C2h	386/182h	*Ion Gauge Delay Time*	FW	Delay time after reaching *Ion Gauge On Pressure* before starting the ion gauge. 0 to 499 seconds
196/0C4h	388/184h	*Number of Start Attempts*	FW	Number of attempts to try to start the ion gauge. 1 to 9
198/0C6h	390/186h	*Ion Gauge OFF Pressure*	FW	The ion gauge pressure at which th ion gauge turns off and reverts to secondary gauge pressure indication.

2.10 Modules Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Module 1	Module 2	***Name***		**Description**
22/016h	20/014h	*Module Name*	IW	3 character name for the ion gauge
48/030h	56/038h	*LO Calibration*	IM	Low calibration value. 5000 = neutral. Range 0 to 12000
50/032h	58/03Ah	*HI Calibration*	IM	High calibration value. 5000 = neutral. Range 0 to 12000
60/03Ch	62/03Eh	*Module Status*	IR	
134/086h	66/042h	*Module Type*	CR	**TYPE**: 000000**XX**h. 80h VALID flag with one of the following: 00h = No module present; 02h = 'V' VG Pirani module; 03h = 'K' thermocouple module (only for Module 2); 04h = 'E' VSP52x module; 05h = 'U' Universal module; 06h = 'F' VSP84x module; 07h = 'W' Universal module ***The following relate to 'W' Modules:*** **INPUT PULL-UP**: 00**X**00000h. 8h VALID flag OR'ed with: 0h = Pull-up off; 1h = Pull-up on **FUNCTION**: 000**X**0000h. 8h VALID flag OR'ed with: 0h = Logarithmic; 1h = Linear **OUTPUT POWER**: 0000**X**000h. 8h VALID flag OR'ed with: 0h = Off; 1h = On (returns to off on unit power up); 2h = Always On (on as soon as unit is powered up)
144/090h	148/094h	*Measured value*	FR	Measured value in currently selected pressure units (or temperature if 'K' module in slot 2
296/128h	240/0F0h	*Minimum Pressure*	FW	**'U' and 'W' modules only**: The minimum pressure corresponding to the *Minimum Input* Parameter
298/12Ah	242/0F2h	*Maximum Pressure*	FW	**'U' and 'W' modules only**: The maximum pressure corresponding to the *Maximum Input* Parameter
300/12Ch	244/0F4h	*Minimum Input*	FW	**'U' and 'W' modules only**: The minimum intput corresponding to the *Minimum Pressure* Parameter

Module 1	Module 2	***Name***		**Description**
302/12Eh	246/0F6h	*Maximum Input*	FW	**'U' and 'W' modules only**: The maximum intput corresponding to the *Maximum Pressure* Parameter

2.11 Thermocouple Input Parameters

Please also refer to Bake-out Parameters.

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Address	***Name***		Description
52/034h	*LO Calibration*	IM	Low calibration value. 5000 = neutral. Range 0 to 12000
54/036h	*HI Calibration*	IM	High calibration value. 5000 = neutral. Range 0 to 12000
146/092h	*Measured Temperature*	FR	Measured temperature value

2.12 Trips Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Address	***Name***		Description
80/050h	*Trip 1 Settings*	CW	**POWER UP OPTIONS**: **X**0000000h. 8h VALID flag OR'ed with: 0h = Retains inhibited or over-riden when powered off; 1h = Inhibit or Override state removed on power up; 2h = Powers up in inhibit state; 5h = Powers up in override state
82/052h	*Trip 2 Settings*		
84/054h	*Trip 3 Settings*	CW	**ALLOCATION**: 00**XX**0000h. 80h VALID flag OR'ed with: 00h = Unallocated; 01h = Ion gauge 1 pressure; 02h = Slot 1 pressure; 03h = Bake-out zone 1; 04h = Slot 2 pressure; 05h = Wide range ion gauge 1 pressure; 06h = Ion gauge 1 is on; 07h = Ion gauge 1 is degassing; 08h = Bake-out zone 2; 09h = Timer 1; 0Ah = Timer 2; 0Bh = Ion gauge 2 pressure*; 0Ch = Wide range ion gauge 2 pressure*; 0Dh = Ion gauge 2 is on*; 0E = Ion gauge 2 is degassing*. * = PVC*duo* only
86/056h	*Trip 4 Settings*		
88/058h	*Trip 5 Settings*		
90/05Ah	*Trip 6 Settings*		
92/05Ch	*Trip 7 Settings*	CW	**ACTION**: 00000**X**00h. 8h VALID flag OR'ed with: 0h = Trip is on when allocation value is less than *Trip Point* Parameter; 1h = Trip is on when allocation value is greater that *Trip Point* Parameter
		CW	**STATE**: 0000000**X**h. 8h VALID flag OR'ed with: 0h = Trip is off; 1h = Trip is on; 2h = Trip is in Inhibit; 5h = Trip is in Override (and on) To put trip in normal trip mode, write either 0 or 1 Note: The status of all 7 trips can be read using the *Trip 1-7 Status* Parameter
128/080h	*Trips 1-7 Status*	CR	**TRIP 1 STATUS**: 0000000**X**h.
		CR	**TRIP 2 STATUS**: 000000**X**0h.
		CR	**TRIP 3 STATUS**: 00000**X**00h.
		CR	**TRIP 4 STATUS**: 0000**X**000h.
		CR	**TRIP 5 STATUS**: 000**X**000Xh.
		CR	**TRIP 6 STATUS**: 00**X**0000Xh.
		CR	**TRIP 7 STATUS**: 0**X**00000Xh.
			For each nibble: 8h VALID flag OR'ed with: 0h = Trip is off; 1h = Trip is on; 2h = Trip is in Inhibit; 5h = Trip is in Override (and on) Note: To write a trip condition, use individual *Trip Settings State* parameters
160/0A0h	*Trip 1 Point*	FW	Point at which trip operates. Note comparison is numerically against the allocated parameter value in which units apply. See also *Trip Settings* and *Trip Hysteresis* Range 1E-16 to 1E6
162/0A2h	*Trip 2 Point*	FW	
164/0A4h	*Trip 3 Point*	FW	
166/0A6h	*Trip 4 Point*	FW	
168/0A8h	*Trip 5 Point*	FW	
170/0AAh	*Trip 6 Point*	FW	
172/0ACh	*Trip 7 Point*	FW	
174/0AEh	*Trip Hysteresis*	FW	To avoid trip "chatter" when the allocated value is close to the Trip point, the point at which the trip turns off again is raised by (*trip hysteresis value*) * (*trip point value*). Range 1.1x to 99.0x

2.13 Digital Inputs Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Address	*Name*		Description
112/070h	*Digital Input 1 Settings*	CW	**POWER UP OPTIONS**: **X**0000000h. 8h VALID flag OR'ed with: 0h = Retains inhibited or over-riden when powered off; 1h = Inhibit or Override state removed on power up; 2h = Powers up in inhibit state; 5h = Powers up in override state
114/072h	*Digital Input 2 Settings*	CW	**INVERT**: 000000**X**0h. 8h VALID flag OR'ed with: 0h = Normal: Digital input is **on** when power applied; 1h = Inverted: Digital input is **off** when power applied.
116/074h	*Digital Input 3 Settings*	CW	**STATE**: 0000000**X**h. 8h VALID flag OR'ed with: 0h = DI is off; 1h = DI is on; 2h = DI is in Inhibit; 5h = DI is in Override (and on) To put digital input in normal mode, write either 0 or 1 Note: The status of all 4 digital inputs can be read using the *Digital Inputs 1-4 Status* Parameter
118/076h	*Digital Input 4 Settings*		
130/082h	*Digital Inputs 1-4 Status*	CR CR CR CR	**DI 1 STATUS**: 0000000**X**h. **DI 2 STATUS**: 000000**X**0h. **DI 3 STATUS**: 00000**X**00h. **DI 4 STATUS**: 0000**X**000h. For each nibble: 8h VALID flag OR'ed with: 0h = DI is off; 1h = DI is on; 2h = DI is in Inhibit; 5h = DI is in Override (and on) Note: To write a DI condition, use individual *Digital Input Settings State* parameters

2.14 Analogue Outputs Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

AOut 1	AOut 2	*Name*		Description
94/05Eh	132/084h	*Settings*	CW	**FUNCTION**: **X**0000000h. 8h VALID flag OR'ed with: 0h = Output proportional to log10 of pressure. 1h = Output linearly related to pressure
			CW	**FAIL MODE**: 0**X**000000h. 8h VALID flag OR'ed with: 0h = Fail to max output voltage. 1h = Fail to min output voltage
			CW	**ALLOCATION**: 00**XX**0000h. 80h VALID flag OR'ed with: 00h = Output set to *Minimum Output* value; 01h = Output set to *Maximum Output* value; 02h = Ion gauge 1 pressure; 03h = Wide range ion gauge 1 pressure; 04h = Module 1 pressure; 05h = Module 2 pressure; 06h = Thermocouple temperature; 07h = Set to level sent to *Output* parameter over comms; 08h = OFF: output is at 0V; 09h = Ion gauge 2 pressure*; 0Ah = Wide range ion gauge 2 pressure*. * PVC*duo* only
			CW	**DI4 INTERLOCKING**: 0000**X**000h. 8h VALID flag OR'ed with: 1h = Interlocked. AO goes to fail mode state if DI fails;
			CW	**DI3 INTERLOCKING**: 00000**X**00h. see DI4 values above
			CW	**DI2 INTERLOCKING**: 000000**X**0h. see DI4 values above
			CW	**DI1 INTERLOCKING**: 0000000**X**h. see DI4 values above
176/0B0h	280/118h	*Minimum Output Voltage*	FW	Minimum output voltage (in *milliVolts*) corresponding to the *Minimum Pressure* Parameter
178/0B2h	282/11Ah	*Maximum Output Voltage*	FW	Maximum output voltage (in *milliVolts*) corresponding to the *Maximum Pressure* Parameter
180/0B4h	284/11Ch	*Minimum Pressure*	FW	Minimum pressure corresponding to the *Minmum Output Voltage* Parameter
182/0B6h	286/11Eh	*Maximum Pressure*	FW	Maximum pressure corresponding to the *Maxmum Output Voltage* Parameter
184/0B8h	288/120h	*Output value*	FR	Output
248/0F8	250/0FAh	*Calibration Value*	FM	Full scale calibration

2.15 Bake-out Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Zone 1 is associated with the built-in thermocouple. For 2 zone operation, a 'K' module must be present in slot 2.

Zone 1	Zone 2	*Name*		Description
72/048h		*Settings/Status*	CR	**CURRENT STEP**: **X**0000000h. 8h VALID flag OR'ed current step (1 to 6). If bake-out not running, value is 0.
			CW	**ION GAUGE 1 AUTO-DEGAS**. 0**X**000000h. 8h VALID flag OR'ed with: 0h = Off; 1h = Ion gauge 1 degases automatically on successful completion of bake-out
			CW	**ION GAUGE 2 AUTO-DEGAS**. 00**X**00000h. 8h VALID flag OR'ed with: 0h = Off; 1h = Ion gauge 2 degases automatically on successful completion of bake-out
			CR	**BAKE-OUT OPERATION**: 00000**X**00h. 8h VALID flag OR'ed with: 0h = No action; 1h = Start Zone 1 only; 2h = Stop Operating Bake-out; 3h = Start Zone 2 only*; 4h = Start bake-out for zones 1 AND 2*. * Only if 'K' module fitted in slot 2. NOTE: Changes to this nibble are actioned at next bake-out cycle and the value is then reset to 0.
			CR	**STATUS**. 000000**XX**h. 80h VALID flag OR'ed with combination of: 01h = Bake-out Operating; 02h = Trips are digital input inhibited; 04h = Trips ion are gauge inhibited; 08h = Trips are user inhibited; 10h = Trips are on; 20h = The bake-out timer is suspended (and trips inhibited); 40h = The bake-out was terminated (the presence of bits 02h, 04h or 08h provide reason for termination).
74/04Ah		*Interlock Settings*	CW	**USER INTERLOCK**: 0**X**000000h. 8h VALID flag OR'ed with: 0h = OFF; 1h = Inhibit; 2h = Suspend.
			CW	**ION GAUGE 2 INTERLOCK** (PVC*duo* only): 00**X**00000h. 8h VALID flag OR'ed with:
			CW	**ION GAUGE 1 INTERLOCK**: 000**X**0000h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 4 INTERLOCK**: 0000**X**000h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 3 INTERLOCK**: 00000**X**00h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 2 INTERLOCK**: 000000**X**0h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 1 INTERLOCK**: 0000000**X**h. 8h VALID flag OR'ed with: 0h = OFF; 1h = Inhibit; 2h = Suspend; 3h = Terminate bake-out
200/0C8h	202/0CAh	*Peak Temperature*	FR	Peak Temperature at the running or last Bake-out. Reset at start of each bake-out
208/0D0h	256/100h	*Step 1 Temperature*	FW	Temperature at the end of Step 1
210/0D2h	258/102h	*Step 2 Temperature*	FW	Temperature at the end of Step 2
212/0D4h	260/104h	*Step 3 Temperature*	FW	Temperature at the end of Step 3
214/0D6h	262/106h	*Step 4 Temperature*	FW	Temperature at the end of Step 4
216/0D8h	264/108h	*Step 5 Temperature*	FW	Temperature at the end of Step 5
218/0DAh	266/10Ah	*Step 6 Temperature*	FW	Temperature at the end of Step 6
220/0DCh		*Temperature Hystersis*	FW	To avoid power switching "chatter" where setpoint and measured temperature are close, the trip off temperature is increased by the hystersis value. Range 0.0 to 9.0°C. (Default = 0.5°C).
222/0DEh		*Ion Gauge 1 Trip Level*	FW	Bake-out ion gauge 1 trip level (if ion gauge 1 interlock is set)
270/10Eh		*Ion Gauge 2 Trip Level*	FW	Bake-out ion gauge 2 trip level (if ion gauge 1 interlock is set) PVC*duo* only
224/0E0h		*Step 1 Time*	FW	Step 1 Duration in hours
226/0E2h		*Step 2 Time*	FW	Step 2 Duration in hours
228/0E4h		*Step 3 Time*	FW	Step 3 Duration in hours
230/0E6h		*Step 4 Time*	FW	Step 4 Duration in hours
232/0E8h		*Step 5 Time*	FW	Step 5 Duration in hours
234/0EAh		*Step 6 Time*	FW	Step 6 Duration in hours
236/0ECh		*Remaining Time*	FR	The remaining time of the current bake-out. If bake-out not operating, value is 0

2.16 Timers Parameters

Decimal/Hexadecimal parameter address.
C=Composite Parameter; S=3 character string parameter; I=Integer value; F=Float value.
R=Read only; W=Read&Write; M=Comms read only/manual write

Timer 1	Timer 2	***Name***		Description
304/130h	320/140h	*Cycle Time*	FW	Cycle Time; the total off + on duration in hours
306/132h	322/142h	*Off Time*	FW	Off time before the on time, in hours
308/134h	324/144h	*Current Time*	FR	The current time in timer operation
310/136h	326/146h	*Restart Time*	FW	The time in the cycle at which a
312/138h	316/13Ch	*Ion gauge 1 Trip Level*	FW	Level at which ion gauge 1 trip interlock occurs
314/13Ah	318/13Eh	*Ion gauge 2 Trip Level*	FW	Level at which ion gauge 2 trip interlock occurs
448/1C0h	452/1C4h	*Interlocks*	CW	**USER INTERLOCK**: 0**X**000000h. 8h VALID flag OR'ed with: 0h = OFF; 1h = Inhibit; 2h = Suspend
			CW	**ION GAUGE 2 INTERLOCK** (PVC*duo* only): 00**X**00000h. 8h VALID flag OR'ed with:
			CW	**ION GAUGE 1 INTERLOCK**: 000**X**0000h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 4 INTERLOCK**: 0000**X**000h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 3 INTERLOCK**: 00000**X**00h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 2 INTERLOCK**: 000000**X**0h. 8h VALID flag OR'ed with:
			CW	**DIGITAL 1 INTERLOCK**: 0000000**X**h. 8h VALID flag OR'ed with: 0h = OFF; 1h = Inhibit; 2h = Suspend; 3h = Terminate Timer
450/1C2h	454/1C6h	*Settings/Status*	CW	**TYPE**: **X**0000000h. 8h VALID flag OR'ed with: 0h = One-shot; 1h = Cyclical
			CW	**OPERATION**: 0**X**000000h. 8h VALID flag OR'ed with: 0h = No action; 1h = Start at off time; 2h = Start at on time; 3h = Jump to value in *Current Time* parameter; 4h = Stop Timer
			CR	**STATUS**. 000000**XX**h. 80h VALID flag OR'ed with combination of: 01h = Timer Operating; 02h = Trips digital input inhibited; 04h = Trips ion gauge inhibited; 08h = Trips User inhibited; 10h = Trips are on; 20h = The timer is suspended (and trips inhibited); 40h = The bake-out has been terminated

3 QueBUS Protocol (QueBUS, QueBUS+CS & QueBUS+CRC)

3.1 Features

- Simple to use ASCII character based protocol.
- Up to 16 parameters can be read and/or written to during each communications exchange (subject to total messages not exceeding 240 bytes).
- 3 optional levels of error checking (none, check-sum or cyclic redundancy check) - see section 3.3.

3.2 Messages

Communication from the PC/PLC (the client) to the controller (the server) is ALWAYS initiated by the client sending a message. The controller CANNOT initialize a communication.

3.3 Error Checking Check-sum and CRC

Many comms protocols include error checking in the form of 1 or 2 bytes additional to the message that is used to verify correct receipt of a message. The ***Protocol Selection*** Parameter provides three options for use of QueBUS:

- **QueBUS-NoCheck**: No check bytes are added, so no error checking is performed. This can be useful, for example, during comms interface development.
- **QueBUS+CS**: A 2 BYTE check-sum is appended to the end of the message. This simple medium-efficacy algorithm for generating the check-sum is described in section 3.4.
- **QueBUS+CRC**: This 2 BYTE check algorithm to the message provides extremely effective error detection. The algorithm used is the same as for MODBUS communications and is described in section 2.3. This provides the highest level of integrity.

Note that if using the check sum or CRC, the 2 BYTEs generated can have any of the 256 possible BYTE values, including special characters that have specific meaning to QueBUS (see 3.5). However, because of the structure of QueBUS and the message data termination character ('!'), the occurrence of special characters within 2 characters of the end of the message provides a context for their presence.

3.4 The QueBUS-CS Check-sum

The QueBUS-CS check-sum is based on the Fletcher algorithm (e.g. https://en.wikipedia.org/wiki/Fletcher's_checksum) and comprises of 2 BYTES derived from ALL the message characters. The code sample below clarifies the procedure:

1. Set two 16bit (or larger) variables to 0.
2. For each message character:
   - Add the character to the first check-sum variable, then apply modulo FFh (255) (remainder after dividing by 255) to first check-sum BYTE.
   - Add first check-sum variable to the second variable and take modulo FFh (255) to the second variable
3. When all characters have been processed, cast the variables to the 2 check-sum BYTEs

C SAMPLE CODE:

```
int zpvc_QBChecksum(unsigned char* lpMess, int nMessLen, unsigned char* lpCS)
{
 int            i, CS0, CS1;
 // Initialize to 0...
 CS0 = CS1 = 0;
 // Loop through the message characters...
 for (i=0; i<nMessLen; i++)
 {
  // Add character to CS0, them modulo to keep in range...
  CS0 = (CS0 + lpMess[i]) % 0xFF;
  // Add first checks sum to second and modulo to keep in range
  CS1 = (CS1 + CS0) % 0xFF;
 }
 // Cast the integers to the check sum characters...
 lpCS[0] = (unsigned char)CS0;
 lpCS[1] = (unsigned char)CS1;
 return 2;  // return length of checksum.
}
```

3.5 "Special" QueBUS characters

The QueBUS protocol reserves several characters for specific functions; aside from arising in the CS or CRC, they can only appear for these purposes:

- '**>**' (ASCII character 62/3Eh): the start of a new message ***from the computer to the controller***. If is followed by 2 BYTEs representing the address of the controller to be addressed (e.g. "03" for address 3)
- '**<**' (ASCII character 60/3Ch): the start of the response ***from the controller to the computer***. If is followed by 2 BYTEs indicating the address of the responding controller (e.g. "15" for address 15)
- '**!**' (ASCII character 33/21h): the end of a message. If the no check-sum or CRC is selected, it is the last character in the message. If check-sum or CRC options are selected, it is followed by the 2 check-sum or CRC BYTEs.
- '**?**' (ASCII character 63/3Fh): the start of a parameter package requesting data. It is followed by a 2 BYTE mnemonic.
- '**#**' (ASCII character 35/23h): the start of a parameter package writing data. It is followed by a 2 character mnemonic and the data string.
- '*****' (ASCII character 42/2Ah): a parameter package error has occurred. Followed by the error code (see 14.4.7)

In the rest of a QueBUS message (except error checking bytes), only the following characters are allowed: 'A' to 'Z', 'a' to 'z', '0' to '9', '.', '+', '-', and ' ' (space).

3.6 QueBUS Message Structure

3.6.1 Message Structure

Sending messages to the controller
- **1 Start BYTE**: Message to the controller ALWAYS starts with '**>**' (ASCII character 62/3Eh).
- **2 Address BYTEs**: In the range "01" to "99".
- **1 or more multi-BYTE Send Parameter Packages** - see section 3.6.2
- **1 Termination BYTE**: Always '!' (ASCII character 33/21h)
- **2 *OPTIONAL* Checksum or CRC BYTEs** - see section 3.3 or 3.4

Response from controller
- **1 Start BYTE**: Message to the controller ALWAYS starts with '**<**' (ASCII character 60/3Ch).
- **2 Address BYTEs**: In the range "01" to "99".
- **1 or more multi-BYTE Response Parameter Packages** - see section 3.6.2
- **1 Termination BYTE**: Always '!' (ASCII character 33/21h)
- **2 *OPTIONAL* Checksum or CRC BYTEs** - see section 3.3 or 3.4

3.6.2 Parameter Packages

Each message from the computer or returned from the controller can contain multiple **Parameter Packages**, each of which is a command to the controller or returns information from the controller.

Parameter Package from computer to controller comprises:
- **1 Command BYTE**: To read a parameter value: '**?**' (3Fh). To write a parameter value: '**#**' (23h)
- **2 BYTE Parameter Mnemonic** as listed in section 3.7 onwards. First character is ALWAYS upper case. Second character can be upper or lower case.
- **Multi-BYTE Data** - if writing data to the controller

Examples: **?Ev** requests the measured emission value; **?Iv** requests the measure pressure value; **#EE10h** (hex value) or **#EE16** (decimal value) sets the Emission Current to Auto-emission; **#Te3.2e-7** sets Trip 5 pressure value to 3.2e-7.

Each package has a response package included in the reply from the controller, in the order in which the

Response Parameter Package
- **Echo of the 1 Command BYTE**
- **Echo 2 BYTE Parameter Mnemonic**
- **Multi-BYTE Response Data**.

  Multi-BYTE Response to a '?' Read Command: If the parameter mnemonic was recognised, the required data string. If an error occurred, a 2 BYTE error code - see section 3.6.3

  Multi-BYTE Response to a '#' Write Command: If the parameter mnemonic was recognized and the value accepted, no data bytes. If an error occurred, a 2 BYTE error code - see section 3.6.3

Responses to the Examples above: **?Ev02.35** indicating emission current is 2.35mA; **?Iv3.57e-07** indicating pressure is 3.57e-7 pressure units; **#EE** indicates emission value correctly set; **#Te** indicates Trip value correctly set.

Notes:
- Each parameter package has a MAXIMUM length of 15 characters. If a longer package is sent, further characters are ignored.
- Only the following characters are allowed for data in Parameter Packages: '0' to '9', 'A' to 'Z', 'a' to 'z', '-', '+', '.' and ' ' (space). Inclusion of other characters when sending a data package will result in an error code for the package.
- Numerical data can be sent in a variety of formats, e.g. as decimal integers (e.g. 12), hexadecimal integers with 'h' or 'H' suffix (e.g. 3ABh), floating point (e.g. 11.35) or exponential formats (2.35e-08). Only 'E' or 'e' (for exponential format), 'h' or 'H' (terminator for hexadecimal value), '0' to '9', '-', '+', '.' and ' ' (space) characters are permitted. Space characters are ignored. Presence of any other character will result in an error.
- Numerical data is returned in a format suitable to the parameter and generally complying to that shown on the instruments display.
- If requesting data, any BYTEs additional to the 3 command and mnemonic BYTEs are ignored.

3.6.3 Error Codes

- ***R**: The mnemonic was not recognized, or an attempt to write to a read-only parameter was made, or the data was corrupted
- ***O**: The data value was out of parameter range.
- ***D**: Attempt to write a parameter but no data present in the package.

Example 1: **?ZO*R** returned as ZO is not a recognized mnemonic
Example 2: **#EE*O** returned in response to #EE25 as 25 is outside the range of Emission setting
Example 3: **#Ea*D** returned in response to a command #Ea without any data

3.6.4 Hexadecimal Status Commands

For reasons of efficiency, a few commands are provided that read status information about several related aspects of controller operation compacted into a single 8 digit hexadecimal string, followed by '**h**' - hexadecimal indicator.. A hexadecimal format allows simple detection of individual bits representing, for example, an operating state and division into 4 bit nibbles or 8 bit bytes allows ease of interpretation.

Each digit represents a 4-bit "nibble" with a value from 0 to F (0 to 15). For example, the command **?ZT** returns an 8 digit hexadecimal string in which the bottom seven 4-bit nibbles provide the state of all 7 trips; similarly **?ZD** returns the state of all 4 digital inputs in the bottom four nibbles.

Some parameters also provide "status" information where different combinations of bits within the value provide conditional information about the parameter. For example, if an operation (e.g. bake-out, timer...) is inhibited, different combinations of bits indicate whether this is due to a digital input, or ion gauge or user.

NOTES:
- All hexadecimal status commands are **READ ONLY**. They cannot be written to.
- The first character for all hexadecimal Status Mnemonics is always '**Z**'.
- The data always comprises 8 characters, each character being a 4 bit nibble.
- The data will always have '**h**' appended to indicated hexadecimal format.
- Nibbles (4 bit sections) that are currently unused are returned as 'x'. For example, the command **?ZD** may respond with "xxxx1210h" where the 'x' character indicates the nibbles are unused; users wanting to perform string to hex number conversion, replace each occurrence of 'x' with '0'.

3.6.5 Putting it all together

As an example of using the protocol, consider wanting to read the measured ion gauge and Slot pressures, change the state of Trip 4 to override and read the states of all trips and digital inputs:
>01?Iv?Xv?Yv#TV5?ZT?ZD!
If check-sum (section 3.4) or CRC (section 2.3) are required, append.

Sample response is:
<01?Iv5.04E-09?Xv3.59E+00?Yv1.11E-04#TV?ZTx0100111?ZDxxxx0011!
followed by the check-sum or CRC if required.

- The first 3 data packages, contain the requested pressures from the ion gauge and Slots.
- #TV without any data indicates that the change to override to trip 4 has been accepted.
- However, note that in the response to the trip status ?ZT command, trip 4 still has a value of '0' (off) as the change has not been implemented before the request for the status has occured. Any subsequent call to ?ZT would however, show a value of '5' for trip 4.

3.7 Parameter Mnemonics

The controller has 256 32-bit memory (parameter) locations. Unlike the MODBUS protocol, QueBUS cannot access raw binary data from the controller (see section 2.6). Individual parameter values are transferred as ASCII strings, each identified by a 2 character mnemonic. In some cases, composite data (such the status of all 7 trips) can be read as a single parameter.

The following tables list and describe the QueBUS commands mnemonics and their type.

3.8 System Global, Communications Parametersa

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

	Name		**Description**
QU	*Unit ID*	R	**UNI**: returns "PVCu". **DUO**: returns "PVCd"
QV	*Firmware Version*	R	***mm.nnn*** where ***mm*** is major release and ***nnn*** minor revision reference
Nu	*Unit Name*	W	3 character unit name; user-defined
QA	*Comms Address*	M	2 character address, e.g. "04" for address 4
QB	*Comms BAUD*	M	1 character. 0 = 4800; 1 = 9600; 2 = 19200; 3 = 38400; 4 = 57600.
QC	*Comms Protocol*	M	1 character. 0 = MODBUS Little Endian; 1 = MODBUS Big Endian; 2 = QueBUS no error checking; 3 = QueBUS with Check Sum; 4 = QueBUS with CRC
QD	*Comms Parity*	M	1 character. 0 = None; 1 = ODD; 2 = EVEN
QE	*Comms Indicator*	W	1 character. 0 = 0ff, 1 = On
QI	*User ID*	W	Up to 6 characters. Stores a user ID number between 0 and 999999
QK	*Bar Graph Minimum Exp*	W	Range: 0 (=1e-16) to *Bar Graph Maximum Exp* parameter
QL	*Bar Graph Maximum Exp*	W	Range: *Bar Graph Minimum Exp* parameter to 22 (=1e+6)
QN	*Display Saver*	W	1 character. 0 = 0ff, 1 = On
QP	*Pressure Units*	W	1 character: 0 = mBar; 1 = Torr; 2 = Pascal
Qe	*Internal Temp*	R	[Diagnostics]
Qh	*Heatsink Temperature*	R	Measured heatsink temperature.
Qs	*Display Saver Time*	W	Up to 3 characters. Value 1 to 499 minutes
Qt	*Menu Escape Time*	W	20 to 999 seconds
Qv	*Internal Temp Setpoint*	R	[Diagnostics]
Qw	*Internal Power*	R	[Diagnostics]
Qx	*Unit Operation Time*	R	Returns the number of hours the unit has been operating
ZS	*General Status*	H	8 characters hexadecimal: "x**JKLMN**xxh". Nibbles indicated by 'x' are not used. **J** FAN Status. Combination of flags: 1h = Fan Error; 2h = Fan On; 4h = Slow speed. **K** High Voltage Status. Combination of flags: 1h = High voltage supply failed; 2h = Ion gauge 1 filament potential failed; 4h = Ion gauge 2 filament potential failed **L** Emission over-current occurred: 1h = Ion gauge 1 over-current; 2h = Ion gauge 2 over-current **M** A/D converter: 1h = A/D failed **N** Temperature: 1h = Under temperature (sensor fail); 2h = Over temperature

3.9 Ion Gauge Parameters

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

IG 1	**IG 2**	***Name***		**Description**
Ni	Nj	*Ion gauge name*	W	3 character name for the ion gauge

IG 1	IG 2	***Name***		**Description**
EA	FA	*Auto-degas Emission Level*	W	The degas level used for automated processes, for example, auto degas at end of bake-out. Values are based on **Emission Set Value** parameter: 13=Degas Low, 14=Degas Medium, 15= Degas High
EE	FE	*Emission Set Value*	W	Sets/reads the setpoint emission value. 0=OFF; 1=0.06mA; 2=0.1mA; 3=0.15mA; 4=0.25mA; 5=0.4mA; 6=0.6mA; 7=1mA; 8=1.5mA; 9=2.5mA; 10=4mA; 11=6mA; 12=10mA; 13=Degas Low; 14=Degas Medium; 15= Degas High; 16=Auto-emission. Note that setting a value that is not between min and max emission range, or a degas level if ion gauge not operating or in valid pressure regime returns an error value (*E).
EN	FN	*Minimum Emission*	W	The minimum emission for fixed and auto-emission. Values are based on **Emission Set Value** parameter and can vary from 1=0.06 to value set in **Maximum Emission** parameter
EX	FX	*Maximum Emission*	W	The maximum emission for fixed and auto-emission. Values are based on **Emission Set Value** parameter and can vary from the value set in the **Minimum Emission** parameter to 12=10mA.
Es	Fs	*PID Setpoint Emission (mA)*	R	The current internal setpoint emission value used by the controller for emission stabilization in mA. If the emission set value has been changed, an emission ramp is started.
Ev	Fv	*Measured Emission (mA)*	R	The measured emission current in mA.
Eo	Fo	*Start-up Power*	W	Power ramped up to during ion gauge start-up. Range is **Minimum** to **Maximum Power** parameter settings with extended resolution to 2 decimal places.
Ep	Fp	*Minimum Power (%)*	W	Minimum power value used by the PID emission stabilization loop. Range is 0% to value set in **Maximum Power** parameter
Eq	Fq	*Maximum Power (%)*	W	Maximum power value used by the PID emission stabilization loop. Range is value set in **Minimum Power** parameter to 99%
Ew	Fw	*Current output power (%)*	R	Reads the current output power %. 2 decimal place resolution
Ey	Fy	*Emission Prop Band*	W	Proportional Band term value used by PID control of Emission. Range 1 to 99 arbitrary units.
Ez	Fz	*Emission Damp*	W	Integral and Differential term value used by PID control of Emission. Range 1 to 99 arbitrary units.
ID	JD	*Digital In 1*	W	DI1 control of ion gauge: 0=OFF; 1=Interlock - ion gauge trips off if interlock fails; 2=Emission On - ion gauge on when DI is on; 3=Emission Toggle - a brief on/off pulse turns ion gauge on if off, and off if on.
IE	JE	*Digital In 2*	W	
IF	JF	*Digital In 3*	W	
IG	JG	*Digital In 4*	W	
II	JI	*Input Filter*	W	Digital input filter. Value 0 to 7.
IM	JM	*Filament Number*	W	0=AUTO - attempts to power up filament 1 first, and if that fails, attempts filament 2; 1=Filament 1; 2=Filament 2 [Cannot be changed if ion gauge is on]
IN	JN	*Filament Ramp Type*	W	0=Standard Ramp (2%/sec); 1=Slow Ramp (0.2%/sec); 2=Fast Ramp (10%/sec); 3=Perform Start Ramp Learn process at Slow Ramp speed
IR	JR	*Pressure Display Resolution*	W	0=1 decimal place; 1=2 decimal places
IU	JU	*Display Units*	W	Selects whether ion gauge reading is reported in currently selected pressure units or in amps: 0=Pressure Units; 1=Direct measurement in amps; 2=Emission-corrected measurement in amps.
Ia	Ja	*Degas Ramp Time*	W	1 to 499 minutes
Ib	Jb	*Degas Hold Time*	W	1 to 499 minutes
If	Jf	*Low Pass Filter*	W	Display filter value. 0.1 to 9.0 seconds.
Ig	Jg	*Gas Sensitivity Factor*	W	Gas sensitivity for current measurements. 0.01 to 99.00.
Il	Jl	*Degas Trip Pressure*	W	Before Degas: The measured pressure must be below this value to enable degas start. During Degas: If the measured pressure rises above this value during degas ramp time, the emission ramp is suspended until the pressure recovers
Is	Js	*Gauge Sensitivity*	W	Sensitivity factor for the ion gauge (in units of /mBar). Range 0.1 to 99.9
Iv	Jv	*Current Gauge Reading*	R	Current ion gauge value. In selected pressure units or amps depending on I**on Gauge Units** parameter setting.

IG 1	IG 2	***Name***		**Description**
ZI	ZJ	*Status 1*	H	8 characters hexadecimal: "**HHK**x**M**x**SS**h". Nibbles indicated by 'x' are not used. **HH** Error states: One or more of the following bits set: 40h=Fan Error; 20h=emission failed due to digital input; 10h=emission failed due to gauge over pressure; 08h=emission failed due to insufficient power maintain emission; 04h=interlock failed - ion gauge not connected; 02h=general emission fail occurred; 01h=emission failed due to filament error. All flags are automatically reset when emission start is retried **K** High voltage error: 1h=high voltage failed **M** Electrometer Detection: 1h=the measured ion current is below the minimum level for the electrometer - this is the same as the "<MinLim" display messages **SS** Current aimed at emission current. Value is as per the **Emission Set Value** parameter in the range 00h to 10h.
ZK	ZL	*Status 2*	H	8 characters hexadecimal: "xxxxxx**SS**h". Nibbles indicated by 'x' are not used. **SS** Current inhibited states. 01h = Digital input inhibited.

3.10 Dual (wide-range) Gauge Mode Parameters

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

IG 1	IG 2	***Name***		**Description**
GA	HA	*Secondary Gauge Selection*	W	Selects the secondary gauge: 0 = No allocation; 1 = Slot 1 selected; 2 = Slot 2 selected
GE	HE	*Emission Level*	W	The emission value that will be used when starting the ion gauge. This is in the range **Minimum** to **Maximum Emission** parameter (see section 3.9), or can be set to Auto-emission.
GG	HG	*Enable DGM*	W	Enables DGM operation, if a secondary gauge is selected 0 = Disabled; 1 = Enabled
GH	HH	*Secondary gauge Protection*	W	Enables secondary gauge pressure monitoring during ion gauge operation 0 = Disabled; 1 = Enabled
GW	HW	*DGM Options*	W	0 = Disable at power up; 1 = Enable at power up (if secondary gauge selected and DGM is enabled); 2 = Perform DGM once as pump-down action
Ga	Ha	*Number of attempts to start ion gauge*	W	Value 1 to 9
Gd	Hd	*Delay Time*	W	Time required for secondary gauge pressure to be below **Ion gauge On** parameter value before starting the ion gauge Value 0.0 to 499.0 seconds
Gf	Hf	*Ion Gauge Off Pressure*	W	Pressure at which ion gauge is turned off and secondary gauge takes over
Gn	Hn	*Ion Gauge On Pressure*	W	Secondary gauge Pressure at which ion gauge is turned on
ZG	ZH	*DGM Status*	R	8 characters hexadecimal: "**GHJK**x**SST**h". Nibbles indicated by 'x' are not used. **G**: DGM: 0 = Disabled; 1 = Enabled **H**: Secondary Gauge protection: 0 = Disabled; 1 = Enabled **J**: DGM Secondary Gauge Allocation: 0 = No allocation; 1 = Slot 1 selected; 2 = Slot 2 selected **K**: DGM Options: 0 = Disable at power up; 1 = Enable at power up (if secondary gauge selected and DGM is enabled); 2 = Perform DGM once as pump-down action **SS**: Operating Flags: Combination of the following bits: 01h = DGM Operating; 02h = In delay period before ion gauge start; 04h = Ion gauge starting; 08h = Ion gauge measurement operating; 10h = Digital input inhibited **T**: Number of completed attempts at starting ion gauge this transition.

3.11 Modules Parameters

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

Slot 1	Slot 2	***Name***		**Description**
Nm	Nn	*Slot Name*	W	3 character name for the ion gauge
XF*	YF*	*'U/W' Module Function*	W	Conversion function from input voltage to pressure 0=Logarithmic; 1=Linear

Slot 1	Slot 2	***Name***		**Description**
XP*	YP*	*'W' Module power on/off*	W	Turns power on and off to connected device 0=off; 1=on
XT	YT	*Slot Type*	R	0=Empty; 2='V' module for VG Pirani; 3='K' module for second thermocouple (dual-zone bake-out - slot 2 only); 4='E' module for VSP52x Pirani; 5='U' universal module; 6='F' module for VSP84x Pirani; 7='W' universal module
XU*	YU*	*'W' Module Input pull-up*	W	Turns pull-up on and off to connected device 0=off; 1=on
Xh	Yh	*High Calibration*	M	Module high calibration value
Xl	Yl	*Low Calibration*	M	Module low calibration value
Xn*	Yn*	*'U/W' Minimum Input*	W	Minimum input voltage corresponding to minimum pressure value
Xp*	Yp*	*'U/W' Minimum Pressure*	W	Minimum pressure corresponding to minimum input voltage value
Xq*	Yq*	*'U/W' Maximum Pressure*	W	Maximum pressure corresponding to maximum input voltage value
Xv	Yv	*Measured value*	R	Current measured value
Xx*	Yx*	*'U/W' Maximum Input*	W	Maximum input voltage corresponding to maximum pressure value

* 'W' module only

3.12 Thermocouple Input

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

	Name		**Description**
Kh	*High Calibration*	M	High calibration value.
Kl	*Low Calibration*	M	Low calibration value.
Kv	*Measured Temperature*	R	Measured temperature value. Returns 999.0 if sensor break - a value > 990 should be interpreted as thermocouple error.

3.13 Trips Parameters

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

	Name		**Description**
TA	*Trip 1 Allocation*	W	0 = Unallocated; 1 = Ion gauge 1 pressure; 2 = Slot 1 pressure; 3 = Bake-out zone 1; 4 = Slot 2 pressure; 5 = Wide range ion gauge 1 pressure; 6 = Ion gauge 1 is on; 7 = Ion gauge 1 is degassing; 8 = Bake-out zone 2; 9 = Timer 1; 10 = Timer 2; 11 = Ion gauge 2 pressure*; 12 = Wide range ion gauge 2 pressure*; 13 = Ion gauge 2 is on*; 14 = Ion gauge 2 is degassing*. * = PVC*duo* only
TB	*Trip 2 Allocation*	W	
TC	*Trip 3 Allocation*	W	
TD	*Trip 4 Allocation*	W	
TE	*Trip 5 Allocation*	W	
TF	*Trip 6 Allocation*	W	
TG	*Trip 7 Allocation*	W	
TI	*Trip 1 Action*	W	0 = Trip is on when allocation value is less than *Trip Point* Parameter; 1 = Trip is on when allocation value is greater that *Trip Point* Parameter
TJ	*Trip 2 Action*	W	
TK	*Trip 3 Action*	W	
TL	*Trip 4 Action*	W	
TM	*Trip 5 Action*	W	
TN	*Trip 6 Action*	W	
TO	*Trip 7 Action*	W	
TS	*Trip 1 State*	W	0 = Trip is off; 1 = Trip is on; 2 = Trip is in Inhibit; 5 = Trip is in Override (and on). To put trip in normal trip mode, write either 0 or 1 Note: The status of all 7 trips can be read using the **Trips 1-7 Status** Parameter
TT	*Trip 2 State*	W	
TU	*Trip 3 State*	W	
TV	*Trip 4 State*	W	
TW	*Trip 5 State*	W	
TX	*Trip 6 State*	W	
TY	*Trip 7 State*	W	
Ta	*Trip 1 Pressure Value*	W	Pressure at which trip operates. 1.0E-16 to 1.0e+6
Tb	*Trip 2 Pressure Value*	W	

	Name		**Description**
Tc	*Trip 3 Pressure Value*	W	
Td	*Trip 4 Pressure Value*	W	
Te	*Trip 5 Pressure Value*	W	
Tf	*Trip 6 Pressure Value*	W	
Tg	*Trip 7 Pressure Value*	W	
Th	*Trip Hysteresis*	W	1.1x to 99.0x
Ts	*Trip 1 Power-up State*	W	0 = Retains inhibited or over-riden when powered off; 1 = Inhibit or Override state removed on power up; 2 = Powers up in inhibit state; 5 = Powers up in override state
Tt	*Trip 2 Power-up State*	W	
Tu	*Trip 3 Power-up State*	W	
Tv	*Trip 4 Power-up State*	W	
Tw	*Trip 5 Power-up State*	W	
Tx	*Trip 6 Power-up State*	W	
Ty	*Trip 7 Power-up State*	W	
ZT	*Trips 1 to 7 Status*	RH	Returns 8 digit hexadecimal number with 'h' suffix (e.g. "x**0001111**h"); Most significant nibble 'x' not used. Each nibble indicates status where: 0 = trip off; 1 = trip on; 2 = trip inhibit; 5 = trip override

3.14 Digital Input Parameters

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

	Name		**Description**
DI	*DI 1 Invert*	W	0 = normal action (on when power applied to input); 1 = Inverted;
DJ	*DI 2 Invert*	W	
DK	*DI 3 Invert*	W	
DL	*DI 4 Invert*	W	
DO	*DI 1 Power-up State*	W	0 = Retains inhibited or over-riden when powered off; 1 = Inhibit or Override state removed on power up; 2 = Powers up in inhibit state; 5 = Powers up in override state
DP	*DI 2 Power-up State*	W	
DQ	*DI 3 Power-up State*	W	
DR	*DI 4 Power-up State*	W	
DS	*DI 1 State*	W	0h = DI is off; 1h = DI is on; 2h = DI is in Inhibit; 5h = DI is in Override (and on) To put Digtal input in normal mode, write either 0 or 1 Note: The status of all 4 digital inputs can be read using the *Digital Inputs 1-4 Status* Parameter
DT	*DI 2 State*	W	
DU	*DI 3 State*	W	
DV	*DI 4 State*	W	
ZD	*Digital Inputs 1 to 4 Status*	RH	Returns 8 digit hexadecimal number with 'h' suffix (e.g. "xxxx**5111**h"); Most significant 4 nibbles 'x' are not used. Each nibble indicates status where: 0 = trip off; 1 = trip on; 2 = trip inhibit; 5 = trip override

3.15 Analogue Output Parameters

R=Read only; W=Read&Write; M=Comms read only/manual writep
H = Hexadecimal Read-only Status Parameter

AO 1	**AO 2**	***Name***		**Description**
OA	PA	*Analogue Output Allocation*	W	0 = Set to Minimum Voltage Output; 1 = Set to Maximum Voltage Output; 2 = Track IG1; 3 = Track IG1 in DGM wide-range mode; 4 = Track Slot 1 module (if present); 5 = Track Slot 2 module (if present); 6 = Track thermocouple; 7 = Set to communications based value; 8 = Set to off (OV output); 9 = Track Ion gauge 2*; 10 = Track IG2 in DGM wide-range mode*. * PVC*duo* only
OD	PD	*DI 1 Interlock*	W	0 = No interlock; 1 = AO goes to fail mode state if DI fails;
OE	PE	*DI 2 Interlock*	W	
OF	PF	*DI 3 Interlock*	W	
OG	PG	*DI 4 Interlock*	W	
OL	PL	*Fail Mode*	W	Determines Output when Fail Mode (e.g. if allocated DI interlocks fail or, if ion gauge allocated, ion gauge is off): 0 = On fail, output **Maximum Voltage** parameter value; 1 = On fail, output **Minimum Voltage** parameter value

AO 1	AO 2	***Name***		**Description**
ON	PM	*Translation Function*	W	Determines relationship between allocated device parameter and voltage output: 0 = Logarithmic; 1 = Linear
Oc	Pc	*Calibration value*	M	Analogue output calibration value
On	Pn	*Minimum Voltage*	W	Voltage that the **Minimum Pressure** parameter value corresponds to (in *milliVolts*)
Op	Pp	*Minimum Pressure*	W	Pressure that the **Minimum Voltage** parameter value corresponds to
Oq	Pq	*Maximum Pressure*	W	Pressure that the **Maximum Voltage** parameter value corresponds to
Ov	Pv	*Output Voltage*	R	Output voltage (V)
Ox	Px	*Maximum Voltage*	W	Voltage that the **Maximum Pressure** parameter value corresponds to (in *milliVolts*)

3.16 Bake-out Parameters

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

Zone 1	Zone 2	***Name***		**Description**
BA		*IG1 Auto-Degas*	W	0 = OFF; 1 = Ion gauge 1 degas automatically on successful completion of bake-out
BB*		*IG2 Auto-Degas*	W	0 = OFF; 1 = Ion gauge 2 degas automatically on successful completion of bake-out
BD		*DI 1 Interlock*	W	0 = OFF; 1 = Inhibit; 2 = Suspend; 3 = Terminate bake-out
BE		*DI 2 Interlock*	W	
BF		*DI 3 Interlock*	W	
BG		*DI 4 Interlock*	W	
BI		*IG1 Interlock*	W	
BJ*		*IG2 Interlock*	W	
BK		*User Interlock*	W	0 = OFF; 1 = Inhibit; 2 = Suspend
BO		*Bake-out Operation*	W	0 = No action; 1 = Start Zone 1 only; 2 = Stop Operating Bake-out; 3 = Start Zone 2 only*; 4 = Start bake-out for zones 1 AND 2*. * Only if 'K' module fitted in slot 2. NOTE: This setting is actioned at next bake-out cycle and the value is then reset to 0.
BZ		*Current Step*	R	0 if bake-out is not running. Otherwise the current step: '1' to '6'
Ba		*Step 1 Duration*	W	Step Duration in ***hh:mm*** format, e.g. "02:30" for 2 hours and 30 minutes. Values in the minutes field greater than 59 is pinned to 59. ***Warning incomplete format may be misinterpreted.***
Bb		*Step 1 Duration*	W	
Bc		*Step 1 Duration*	W	
Bd		*Step 1 Duration*	W	
Be		*Step 1 Duration*	W	
Bf		*Step 1 Duration*	W	
Bk	Ck**	*Step 1 Temperature*	W	Step zone temperature. Range 0.0 to 500.0°C
Bl	Cl**	*Step 2 Temperature*	W	
Bm	Cm**	*Step 3 Temperature*	W	
Bn	Cn**	*Step 4 Temperature*	W	
Bo	Co**	*Step 5 Temperature*	W	
Bp	Cp**	*Step 6 Temperature*	W	
Bs	Cs**	*Current Setpoint*	R	Current zone setpoint value
Bt		*Remaining Bake-out time*	R	Remaining bake-out time. Value is returned as ***hhh:mm:ss*** format.
Bu	Cu	*Peak Zone Temperature*	R	The peak zone temperature achieved in current running bake-out, or during previous bake-out
Bv	Cv	*Current Measured Zone Temperature*	R	The current zone temperature. For zone 1, this is the same as the **Thermocouple Measured** temperature. For zone 2, this is the same as **Module 2 Measured** value.
Bw		*IG1 Pressure Interlock Value*	W	If IG1 Interlock is set, the pressure at which the interlock operates
Bx*		*IG2 Pressure Interlock Value*	W	If IG2 Interlock is set, the pressure at which the interlock operates
Bz		*Bake-out Temperature Hysteresis*	W	To avoid power switching "chatter" where setpoint and measured temperature are close, the trip off temperature is increased by the hystersis value. Range 0.0 to 9.0°C. (Default = 0.5°C).

Zone 1	Zone 2	***Name***		**Description**
ZB		*Bake-Out Status*	RH	Returns 8 digit hexadecimal number with 'h' suffix (e.g. "xxxxxx**SS**h"); Most significant 6 nibbles 'x' not used. **SS**: Bottom 2 nibbles are a combination of the following bits: 01h = Bake-out Operating; 02h = Trips are digital input inhibited; 04h = Trips ion are gauge inhibited; 08h = Trips are user inhibited; 10h = Trips are on; 20h = The bake-out timer is suspended (and trips inhibited); 40h = The bake-out was terminated (the presence of bits 02h, 04h or 08h provide reason for termination).

* PVC*duo* only
** Only if 'K' module fitted in Slot 2.

3.17 Timer Parameters

R=Read only; W=Read&Write; M=Comms read only/manual write
H = Hexadecimal Read-only Status Parameter

Tmr 1	**Tmr 2**	***Name***		**Description**
UA	VA	*Timer Operation*	W	0 = No action; 1 = Start at off time; 2 = Start at on time; 3 = Jump to value in **Current Time** parameter; 4 = Stop Timer
UD	VD	*DI 1 Interlock*	W	0 = OFF; 1 = Inhibit; 2 = Suspend; 3 = Terminate Timer
UE	VE	*DI 2 Interlock*	W	
UF	VF	*DI 3 Interlock*	W	
UG	VG	*DI 4 Interlock*	W	
UI	VI	*IG1 Interlock*	W	
UJ*	VJ*	*IG2 Interlock*	W	
UK	VK	*User Interlock*	W	0 = OFF; 1 = Inhibit; 2 = Suspend
UO	VO	*Timer Type*	W	0 = One-shot; 1 = Cyclical
Uc	Vc	*Cycle Time*	W	Total Cycle Time in format ***hhh:mm:ss***. When writing, h, m and s fields must be present.
Ui	Vi	*IG1 Interlock Pressure*	W	Pressures at which ion gauge interlock fails
Uj*	Vj*	*IG2 Interlock Pressure*	W	
Uo	Vo	*On Time*	W	Duration of On Time in format ***hhh:mm:ss***. Cannot exceed **Cycle Time**. When writing, h, m and s fields must be present.
Ur	Vr	*Restart Time*	W	Restart Time in format ***hhh:mm:ss***. Cannot exceed **Cycle Time**. When writing, h, m and s fields must be present.
Ut	Vt	*Current Timer Time*	R	If Timer is off, this returns 0. If operating, the current time in the cycle.
ZU	ZV	*Timer Status*	R	Returns 8 digit hexadecimal number with 'h' suffix (e.g. "**T**xxxxx**SS**h"); **T**: Type: 0 = One-shot; 1 = Cyclical **SS**: Combination of bits indicating status: 01h = Operating; 02h = DI inhibited; 04h = IG inhibited; 08h = User inhibited; 10h = Trips allocated to timer are on; 20h = Timer is suspended; 40h = Timer was terminated