BA2xx Communication Protocol

Part 1

Table of Contents

1. Overview

The purpose of this document is to describe the protocol for the serial interface between the BA2xx and the host system. This protocol allows for the transmission of waveform data and the setting and reading of various system parameters.

2. Abbreviations and Terminology

host	The host is the user device that sends commands to and receives data from the sensor.
CKS	Checksum - each data packet has a checksum for error-detection
CMD	Command byte identifier - each interface command has a unique identifier
NBF	Number of bytes to follow - number of bytes remaining in a data packet
exp	Expired
ETCO2	End tidal carbon dioxide
RS-232	An asynchronous serial communication standard.
TBD	To Be Determined.Value or property not yet decided.

3. General Terminology

3.1. Gas Compensations

The measurement of CO2 is affected by temperature, pressure, and gas compensations. The barometric pressure as well as the presence of O2, N2O, helium, and anesthetic agents in the gas mixture needs to be compensated for by the BA2xx in order to achieve its stated accuracy. The BA2xx provides instrument settings to allow the host to communicate these operating conditions. The instrument settings for these parameters should be set when initially connecting to the BA2xx and whenever there is a change in the conditions at the patient airway.

In the BA2xx, the temperature of the gas in the airway also effects the CO2 measurement. The mainstream BA2xx assumes the gas is at a temperature of 35℃. Therefore under normal patient use, it is not necessary to adjust this setting. When performing bench tests with bottled gas at a room temperature or when using the side stream BA2xx, it is necessary to adjust the instrument setting for the gas temperature to achieve the maximum accuracy for the BA2xx.

3.2. Zeroing

The BA2xx is compatible with a variety of different airway adapters as described in the BA2xx product specification. The BA2xx Zero allows for the BA2xx to accommodate the optical characteristics of each of the different adapter types. A zero should be performed whenever the type of adapter being used with the BA2xx is changed. For optimal accuracy, a zero should also be performed whenever the BA2xx is connected to the host system.

Before performing a zero, the BA2xx should be removed from the patient circuit and the airway adapter type to be used in the circuit should be inserted into the BA2xx. Care should be taken ensure that the airway adapter is clear of any residual CO2 gas. The maximum elapsed time for a zero is 30 seconds. The typical time for a zero is 15 – 20 seconds.

Several conditions may also request that a zero be performed. These requests stem from changes in the airway adapter that may indicate that the sensor is not in optimal measuring condition. When this occurs, the airway adapter should be checked to ensure optical occlusions such as mucus have not obscured the adapter window. If occlusions are found, the airway adapter should be cleaned or replaced.

3.3. No Breaths Detected

The “No Breaths Detected” timeout is the maximum time allowed from the detection of one breath to the next breath. Therefore, if the time between breaths exceeds the time out period, the status condition “No Breaths Detected” will be set (see Appendix A). Upon reaching the time-out period, the No Breaths Detected bit will be set in the status byte and the ETCO2 value, respiration rate and inspired CO2 value will be set to zero.
At start-up or following a zero, three breaths need to be detected before this timer is activated. To clear the “No Breaths Detected” time-out condition three breaths are needed, a zero must occur, or the Reset No Breaths Detected command must be issued.
It is important to note that the BA2xx is not an apnea monitor. The software cannot discriminate between the patient no longer breathing and a sensor that is disconnected from the patient circuit.

4. Communication Setup

Serial transmission is performed using the RxD, TxD and GND lines of a standard RS-232 interface.
Transmission characteristics for the serial data to and from the sensor:

- Rate : 19200 bps
- Format: 1 start bit ,8 data bits, 1 stop bit, no parity checking.
- Flow control: No hardware or software flow control (handshaking) is used.

4.1. Transmission Format

Data is transmitted and received as data packets in the following format:

CMD - NBF - [data_byte0 ... data_byten] – CKS

Where :

CMD - command byte, range 80h – FFh
NBF- number of bytes to follow this byte, including the checksum
data_byte0 ... data_byten - data bytes 0 thru n
CKS – checksum

Commands are expected to be transmitted in a timely fashion. No more than 500 msec may

pass between the time the first byte is received and the time the last byte of a command is received. If more than 500 msec passes between the first and last bytes of a command, a time-out will occur and the module will record a NACK error (No ACKnowledgement error).

The command structure is formatted in such a way to allow easy synchronizing of command bytes. All command bytes have the most significant bit set to 1. Therefore, the range of all available command bytes is 80h to FFh. All other bytes in the data packet, including the checksum are not allowed to have their MSB bit set, and therefore fall in the range of 0h - 7Fh.

Command Byte: 1xxxxxxxb
NBF byte: 0xxxxxxxb
1 byte data: 0xxxxxxxb
2 byte data: 0xxxxxxxb 0xxxxxxxb
3 byte data: 0xxxxxxxb 0xxxxxxxb 0xxxxxxxb
4 byte data: 0xxxxxxxb 0xxxxxxxb 0xxxxxxxb 0xxxxxxxb
5 byte data1: 0000xxxxb 0xxxxxxxb 0xxxxxxxb 0xxxxxxxb 0xxxxxxxb
checksum byte: 0xxxxxxxb

where x signifies either a 0 or 1.

Multiple byte data is encoded as follows:

2 Byte data xxxxABCD EFGHIJKL
transmitted as 0xxABCDE 0FGHIJKL

3 Byte data xxxxABCD EFGHIJKL MNOPQRST
transmitted as 0xABCDEF 0GHIJKLM 0NOPQRST

The ranges of the data types are as follows:

The range for 5-byte data values is truncated to 32 bits as shown by the ranges specified in Table 2.

Table 2: Data Types and Ranges

Data Type	Unsigned Range	Signed Range
Command Byte	80 - FFh	NA
5 Byte Data	0 – 4,294,967,295	-2,147,483,648, +2,147,483,647
4 Byte Data	0 - 268,435,455	-134,217,728, +134,217,727
3 Byte Data	0 - 2,097,151	-1,048,576, +1,048,575
2 Byte Data	0 - 16383	-8192, +8191
1 Byte Data	0 - 127	-64, +63
Checksum Byte	0 - 7Fh	NA
NBF Byte	0 - 7Fh	NA

The formula for extracting multiple byte data (both signed and unsigned) from a command string is defined as needed in the command definitions.

4.1.1. Packet Checksums

The checksum is computed by the formula:

CKSUM = (not (CMD + NBF + [data byte0] + ... [data byten]) + 1) & 7Fh

thus forming the checksum byte, so when added,

((CMD + NBF + [data byte1] + ... [data byten]) & 7Fh) + CKSUM) & 7Fh = 0

It is strongly recommended that the checksum byte of each packet be computed to verify a valid packet.

Example:

The Get Revision Command is CAh 02h 00h CHKSUM (refer to section 8.7 for description)

CHKSUM = not (CAh + 02h+ 00h)
= not (CCh)
= 34h

Therefore, the full command is CAh 02h 00h 34h

The following C language source code can be used to compute the checksum for outgoing packets:

```
enum Boolean {FALSE, TRUE};
void AddChecksum(short num, char buf[])
{
    char checksum;
    short i;
    checksum = 0;
    for (i = 0; i < num; i++)
    checksum = (char) (checksum + buf[i]);
    checksum = (char) ((-checksum) & 0x7F);
    buf[num] = checksum;
}
```

The following C language source code can be used to verify the checksum for incoming packets:

```
Boolean IsChecksumValid(char *lpBuf, short num)
{
    char checksum;
    checksum = *lpBuf;
    lpBuf++;
    num--;
    for (; num > 0; num--) {
    if (*lpBuf >= 0x80)
    return FALSE; // invalid data byte
    checksum = (char) (checksum + *lpBuf);
    lpBuf++;
    }
    checksum &= 0x7F;
    if (checksum == 0)
    return TRUE;
    else return FALSE; // invalid checksum
```

}

4.1.2. About the Packet's Number of Bytes to Follow

All command packets contain a NBF (number of bytes to follow this byte in the command packet) byte. This byte allows ease of decoding a command and future command enhancement. If enhancements to any commands are needed, the present command structure does not need to be changed. Enhancements will be incorporated by increasing the NBF for the command and adding additional bytes to the end of the present command to incorporate the change. This method allows for backward compatibility and future flexibility. The example below illustrates decoding a packet. The host recognizes an existing command that has two parameters, data_byte1 and data_byte2. Instead of hard-coding the location of the checksum byte as the fifth byte in the packet, the host should use the NBF byte. In the existing command, the NBF byte = 3; so after reading the first two bytes (CMD and NBF), the host should read the next three bytes and the checksum byte is the last byte read. Where this really comes into play is when the host encounters the enhanced version of the command. In the enhanced version, an additional byte is added to the data bytes, data_byte3. The host is not expecting this byte to be there (it is only familiar with the existing command with two data bytes). If the host uses the NBF byte as a guide to the checksum, it can still interpret the packet and validate its integrity. The host will only use data_byte1 and data_byte2 and will ignore the unknown data_byte3, but it can correctly calculate the checksum for the packet.

Example:

Existing Command:

Assume there is an existing command with the following format:

CMD - NBF - data_byte1 - data_byte2 – CKSUM

where NBF = 3.

Enhanced Command:

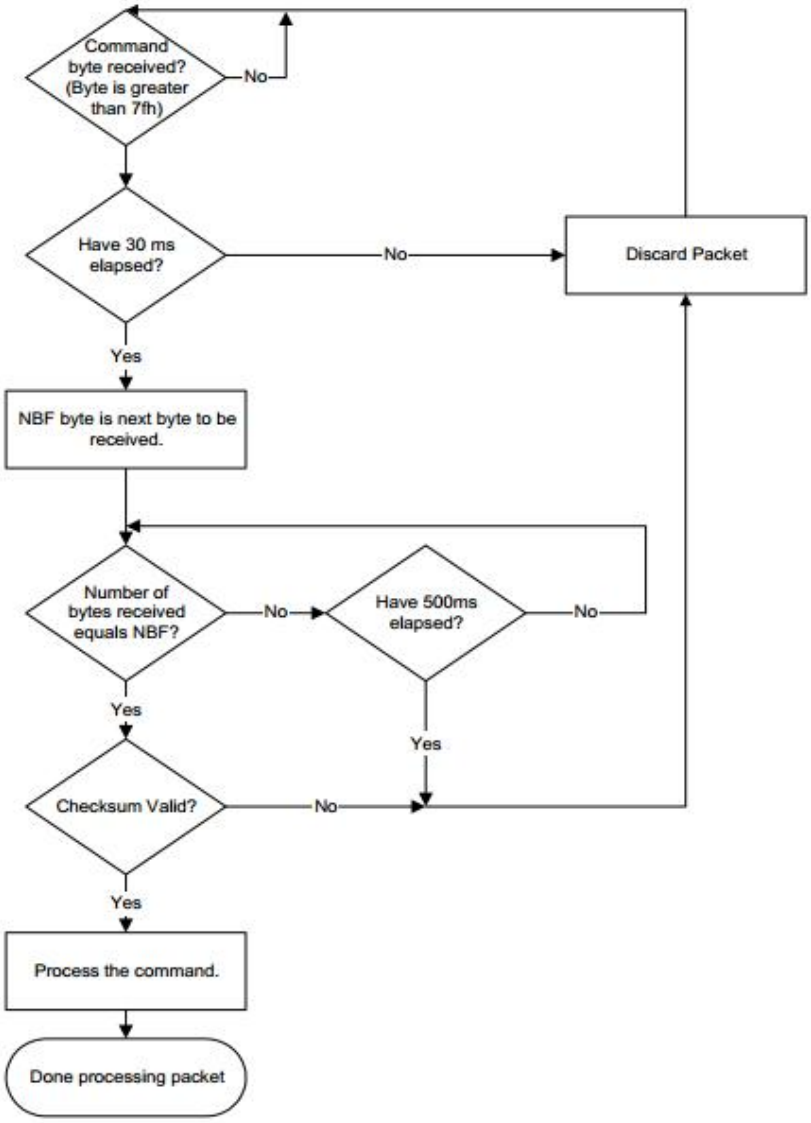
In the enhanced command, an additional byte is needed in the data packet.

CMD - NBF - data_byte1 - data_byte2 - data_byte3 - CKSUM

where NBF = 4, with the additional data byte3.

4.1.3. Processing a command packet

The following flow chart describes the proper sequence for receiving and decoding a data packet sent from the BA2xx.



The host should wait to receive a command byte. If the byte greater than 7Fh it is a command byte, otherwise the host should discard this byte. Once the command byte is received, the next byte will be the Number of Bytes (NBF) to follow byte. The host should receive the NBF byte within 30 ms of receiving the command byte; if it does not then the host should discard the packet. The host should wait up to 500ms to receive the remaining bytes in the packet based on the NBF count. Once all the bytes in the packet have been received, the checksum byte should be used to ensure the integrity of the packet. If the packet is valid it can now be decoded based on the Command Byte.

5. Communicating with the BA2xx via the Serial Interface

5.1. Startup

The BA2xx requires approximately five seconds from power up for internal initialization. The Stop Continuous Mode command (command C9) should be sent until a valid response (non-NACK) is received. Once this response is received, the BA2xx startup has completed and the BA2xx is ready to respond to all commands via the serial interface.

5.2. Initialization of the BA2xx

For optimal CO2 accuracy, the host must initialize several settings on startup or whenever the BA2xx is initially connected to the host. The Get/Set Instrument Settings Command (Command 84h) should be used to input the following environmental conditions:

1) Barometric Pressure (Setting Byte Identified ISB #1)

2) Gas Compensations (O2, N2O, He, anesthetic agents) (Setting Byte Identified ISB #11)

3) These settings should also be updated whenever there is a change in these operating conditions. When the BA2xx is first connected or powered on, the “compensation not yet set” is set in the instrument status bytes (see Appendix A). This status message is cleared whenever the BA2xx receives both a barometric pressure and gas compensation setting via the Get/Set Instrument Settings command.

5.3. Receiving Real-time Waveforms and Data Parameters

The serial interface commands CO2 Waveform /Data Mode (Command 80h) is used to start waveform/data transmission. These waveform mode packets will continue to be transmitted from the BA2xx until the waveform/data mode is stopped using the Stop Continuous Mode command (Command C9h). The data rate for the continuous waveform modes is 100 Hz, therefore a host can expect a new waveform packet approximately every 10 msec. Every waveform response packet will contain waveform data. The waveform modes may also transmit one of a list of available data parameters in the waveform packet (refer to the DPI tables in the waveform command definitions). Several of these additional data parameters are transmitted at a fixed data rate of once per second. Some are transmitted whenever an event occurs (such as the breath detected DPI). The majority of waveform mode packets will not contain a DPI nor its associated data parameter. No more than one data parameter is transmitted per waveform data packet.

The format of a CO2 Waveform/Data Mode packet is as follows:

80h - NBF - WFB1 – WFB2 - [DPI - DB1 .. DBn] – CKS

where 80h is the command identifier, NBF is the number in the packet, WFB1 and WFB2 are the real time waveform samples, DPI is the data parameter identifier, and DB1 thru DBn are the

data parameter values. The braces ([ ]) around the data parameter bytes indicate that the transmission of these bytes are optional; some packets may have these bytes and some may not. The NBF byte can be used to determine if the waveform packet contains a data parameter.

5.3.1. Transmission of Data Parameters

Data parameters are transmitted after an event has occurred. The breath detected data parameter is transmitted whenever the BA2xx detects a breath. The Extended CO2 Status/Error data parameter is transmitted every second (100 packets). The calculated parameters are transmitted once a second. Only one data parameter is transmitted per CO2 Waveform/Data Mode packet. Each data parameter has an associated unique data parameter identifier (DPI). The host uses the DPI value to determine which parameter is being transmitted. In the waveform packet, the data parameter bytes follow the waveform sample bytes. The number of bytes used to encode the data parameter value varies depending on the parameter; refer to the waveform mode commands for details on the size of each data parameter. For example, a waveform packet that has samples for CO2 waveforms and has data parameter number #3 (Respiration Rate), will consist of the following bytes:

80h - 7 – SYNC - WFB1 – WFB2 - 3 – DB1 – DB2 – CKS

The decoding of the DPI parameter is dependent on the parameter; again, refer to the CO2 Waveform/Data Mode for details on the decoding the parameter bytes. Many parameters, however, consist of two bytes (high-byte and low-byte). In the example above, to decode the value of Respiration Rate Flow, the following formula should be used:

Respiration Rate = (DB1 * 128) + DB2

When designing the host interface to the BA2xx 5, the host communications routine should be robust enough to interpret any future changes which may be added to commands. For backward compatibility, additional DPI numbers may be added to an existing command. Since the existing host software will not know how to interpret any future DPIs, it should ignore the DPI and its associated data while still decoding the data contained in the remainder of the command packet.

5.3.2. Validity of Data Parameters

BA2xx errors can cause the CO2 waveform and selected patient parameters to be set to default values. The table below lists the BA2xx errors and the effect they have on the given parameters.

Affected Parameters	Values Sent for Affected Parameters	Affected Parameters
ETCO2 Inspired CO2 Respiratory Rate	0	Compensations Not Set
		Zero in Progress
		Zero Error
		Zero Required

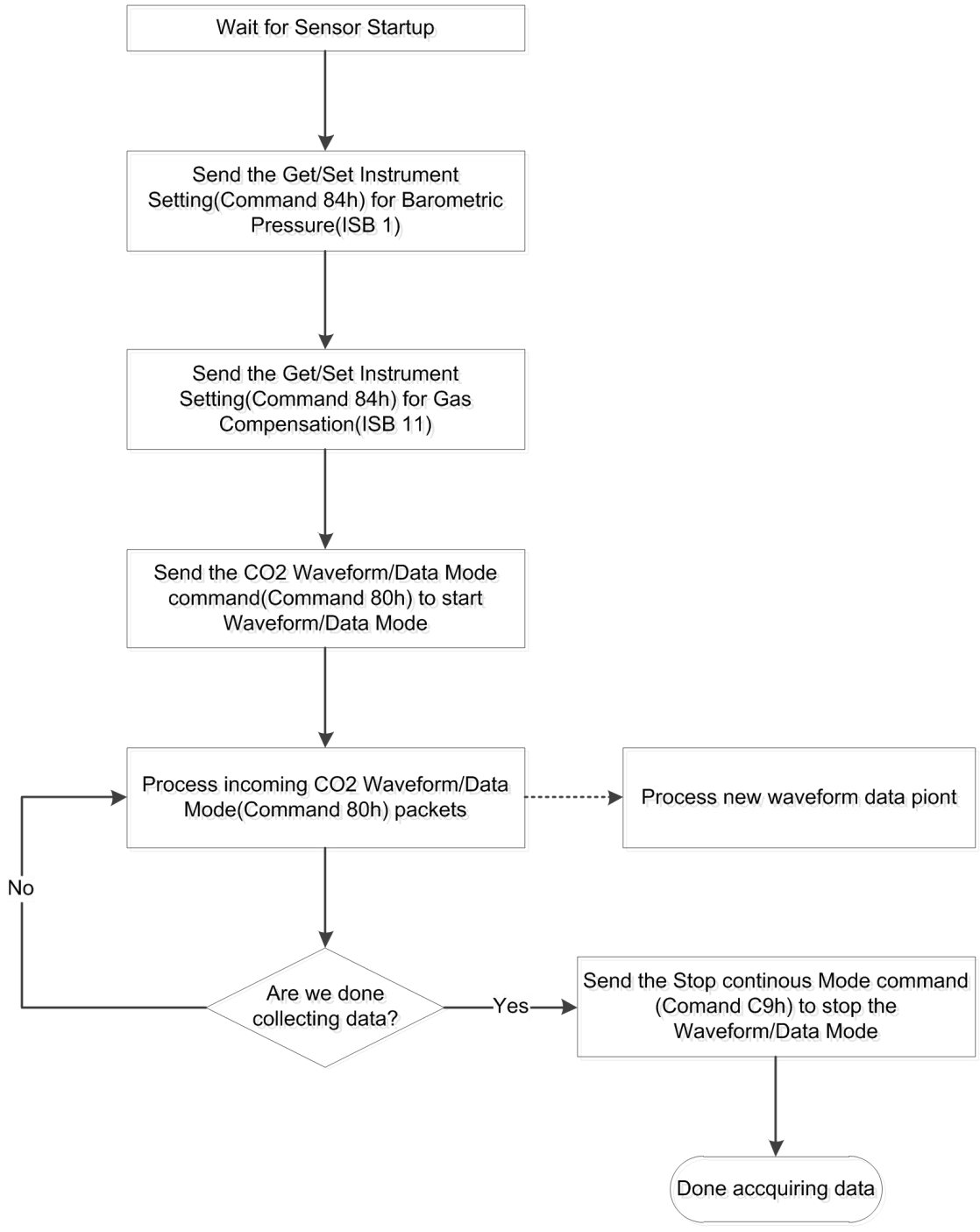
		EEPROM Not Formatted
		Source Current Drift
CO2 Waveform	-10.00	Check Airway Adapter
		Source Current Drift
		Source Current Limit Error
		Zero in Progress
		Zero Error

5.4. Stopping Waveform/Data Mode

The Stop Continuous Mode command (Command C9h) is used to end the continuous stream of Waveform/Data Mode packets.

6. General Usage Overview

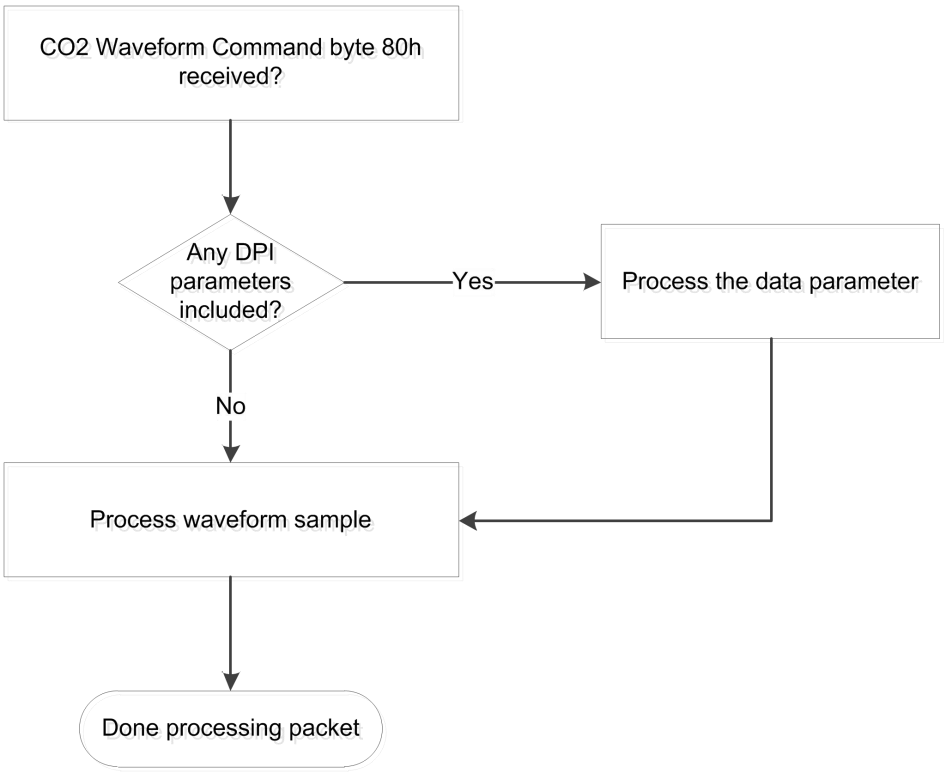
The flow-chart below illustrates the command sequence that should be used to communicate with the BA2xx using the Waveform/Data Mode.



The first step in starting the data acquisition is setting the Barometric Pressure and gas compensations using the Get/Set Instrument Settings command. The Start Waveform/Data Mode command is used to start the transmission of waveform and parameter data; the Waveform Mode byte is saved from the response packet. Then the host enters a data

acquisition loop where incoming CO2 Waveform/Data Mode packets are received and processed. When data acquisition is complete, the host sends the Stop Continuous Mode command to stop the transmission of data. To ensure the most accurate measurements of the BA2xx, the Get/Set Instrument Settings should be used whenever the conditions change.

The flow-chart below illustrates the steps that should be taken to properly interpret CO2 Waveform/Data Mode packets.



When the host receives a CO2 Waveform/Data Mode packet, it should first check the validity of the packet. Any included data parameters can then be decoded. The host should then process the waveform sample that is transmitted. When the packet processing is complete, the host should prepare to process the next packet.

7. Command Reference

There are two basic types of commands: continuous response commands and single response commands. Continuous response commands return a continuous stream of data, whereas single response commands return a single response. To stop a continuous response command from transmitting, use the command Stop Continuous Mode (Command C9h). While the continuous command is transmitting data, all single response commands are still valid. Due to the high throughput required by the continuous mode commands, single response command transmissions sent while a continuous response command is transmitting should be kept to a minimum. A continuous response command is given transmission preference over single response commands in certain situations. When the module's 256-byte serial transmit buffer is 75% full, only the continuous command will be transmitted to prevent buffer overrun. The response to the single response command is not transmitted. Under normal circumstances, this should not happen; if this is happening, limit the number of commands transmitted or transmit the commands over a longer time period. The following pages describe the current available commands.

7.1. CO2 Waveform/Data Mode (Command 80h)

Command:

80h - NBF - 0 – CKS

Command Type:

Continuous Response Command

Response:

80h - NBF – SYNC - CO2WB1 - CO2WB2 - [DPI – DPB1 - DPBn] – CKS

Definitions:

80h - command byte

NBF - number of bytes to follow

SYNC - Synchronization counter which increments with each packet sent. Counter starts at 0 and rolls over to zero when it reaches 127. This byte can be used to detect missed packets.

CO2WB1, CO2WB2 - CO2 Waveform x100[1].

DPI - Data Parameter Index (Valid DPI's are defined below) The DPI is sent only when necessary.

DPB1, DPBn - These bytes are sent only as necessary. These bytes contain the DPI data and the number of bytes can vary from zero to five bytes.

CKS - Checksum byte

Description:

The CO2 Waveform Mode command is sent at a rate of 100 Hz. This command is used to transmit CO2 waveforms and data. Whenever the CO2 waveform cannot be computed, the minimum CO2 value (-10.00) is sent as a waveform "penlift". This corresponds to

CO2WB1 & CO2WB2 both being zero. Additionally, the value for 0 in all units (mmHg, kPa, and %) is the same absolute byte value for CO2WB1 & CO2WB2.

The CO2 waveform can be decoded as follows:

Units	Range	Resolution	Conversion
mmHg	-9.99 to 150.00 mmHg	0.01 mmHg	(((128 * CO2WB1) + CO2WB2) – 1000)/100
kPa	-9.99 to 20.00 kPa	0.01 kPa	(((128 * CO2WB1) + CO2WB2) – 1000)/100
Percent	-9.99 to 19.70 %(for CO2)	0.01 %	(((128 * CO2WB1) + CO2WB2) – 1000)/100

[1] CO2 values are transmitted in the current CO2 units which can be set and retrieved using the Get/Set Instrument Setting (ISB # 7). Units cannot be changed while this command is active.

The DPI byte contains patient parameter data. The types of DPI are summarized in the table below and described in detail in Appendix B.

Command 80h DPI Parameter Table

DPI	Number Bytes	Description	Calculation
1	5	CO2 Status	See Appendix A
2	2	ETCO2 x10[1]	ETCO2 = (DPB1 * 128) + DPB2
3	2	Resp Rate	RespRate = (DPB1 * 128) + DPB2
4	2	InspCO2 x10[1]	Insp CO2 = (DPB1 * 128) + DPB2
5	0	Breath Detected Flag	Breath has been detected when this DPI is sent.
7	2	Hardware Status	Only sent when nonzero. See Appendix A

Refer to the DPI Reference in Appendix B for a detailed description of the DPIs.

7.2. BA2xx Zero Command (Command 82h)

Command:

82h -NBF – CKS

Command Type:

Single Response Command

Response:

82h - NBF - ZSB – CKS

Definitions:

82h - command byte
NBF - number of bytes to follow
ZSB - Zero status byte (see table below)
CKS – checksum

Description:

This command is used to initiate a BA2xx zero. A zero is used to correct for differences in airway adapter types. The BA2xx zero must be performed free of any CO2. Refer to section 4.2, BA2xx Zeroing for a detailed description.

ZSB	Description
0	BA2xx Zero started
1	BA2xx not ready. See the errors in Appendix A and remedy the BA2xx error before attempting another zero. This is caused by one of the following errors: • In sleep mode • BA2xx temperature not stable • BA2xx sensor faulty
2	BA2xx Zero in already progress (started).
3	BA2xx Zero attempted and breaths have been detected in the last 20 seconds. Remove Sensor and Airway Adapter from presence of CO2 and wait for Breath Detected Status bit, in Appendix A, to clear before attempting another Zero.

7.3. Get/Set Sensor Settings (Command 84h)

Command:

84h - NBF - ISB - [DB1 - … - DBN] – CKS

Command Type:

Single Response Command

Response:

84h - NBF - ISB - DB1 - … - DBN – CKS

Definitions:

84h - command byte

NBF - number of bytes to follow

ISB - Sensor Setting Byte Identifier (see table below)

DB1 … DBN - Data bytes used to set and return the value of a particular sensor setting

CKS – checksum

Description:

This command is used to get and set the various sensor settings in the BA2xx module. When the command is transmitted to the BA2xx without the optional DB1 … DBN data bytes, the current value of the specified sensor setting is transmitted in the command's response string. This corresponds to "getting" the current value of the sensor setting. If the DB1 … DBN data bytes are transmitted from the host, the sensor setting is set to that value. This corresponds to "setting" the specified sensor setting, and this new value is transmitted in the command's response string. A host must request a valid setting byte identifier - ISB. If an invalid ISB is requested, the command's response string will return with the ISB = 0 and no returned data bytes.

Examples:

- This example command string gets the current ETCO2 Time Period:

  84h - 02h - 05h – 75h

  The BA2xx's response would be:

  84h - 03h – 05h - 01h – 73h (current ETCO2 Time Period is 1 breath)

- This example command string sets the current ETCO2 Time to 10 seconds:

  84h - 03h - 05h - 0Ah – 6Ah

  The BA2xx's response would be:

  84h - 03h - 05h - 0Ah – 6Ah (current ETCO2 Time Period is 10 seconds)

The table below lists the ISB byte identifiers and the corresponding sensor settings.

ISB	Number of Bytes	Sensor Setting Description
0	0	Invalid Instrument or Parameter Setting Number of Data Bytes = 0
1	2	Barometric Pressure

		Default: 760 mmHg. Resolution: 1 mmHg (400-850 mmHg) Conversion: Barometric Pressure = (128 * DB_1) + DB_2 DB_1 = (Barometric Pressure / 128) & 7Fh DB_2 = (Barometric Pressure) & 7Fh Notes: This setting is used to set current Barometric Pressure.
4	2	Gas Temperature Default: 35.0 ℃ Resolution: 0.1 ℃ (0.0 – 50.0℃) Conversion: Gas Temperature ℃ = (128 * DB_1 + DB_2) / 10 Notes: This setting is used to set temperature of the gas mixture. This setting is useful when bench testing using static gasses where the temperature is often room temperature or below.
5	1	Current $ETCO_2$ Time Period Default: 10 Seconds. Conversion: $ETCO_2$ time period = DB_1 = 1 1 breath = 10 10 seconds = 20 20 seconds Notes: This setting is used to set the calculation period of the $ETCO_2$ value. The end-tidal CO_2 value is the highest peak CO_2 value of all end of expirations (end of breaths) over the selected time period. If less than two breaths exist in the selected time period, the value will be the maximum $ETCO_2$ value for the last two breaths.
6	1	No Breaths Detected Timeout Default: 20 seconds. Resolution: 1 second (10 to 60 seconds) Conversion: No Respiration Timeout = DB_1 Notes: This setting is used to set the no breaths detected time-out. This time-out is the time period in seconds following the last detected breath at which the BA2xx will signal no breaths detected.
7	1	Current CO_2 Units Default: mmHg Conversion: CO_2 units = DB_1 = 0 CO_2 units are mmHg = 1 CO_2 units are KPa = 2 CO_2 units are percent (%) Note: Continuous waveform mode commands (the CO_2 Waveform Mode command [command 80h] MUST NOT be active when this command is used otherwise this command will be ignored and the setting will remain unchanged. If any continuous waveform mode command is active, the following must occur: 1) The continuous waveform mode command must first be stopped

		with the Stop Continuous Mode Command (command C9h). 2) The CO_2 Units Setting can be changed with this command. 3) Then the continuous waveform command may be restarted.
8	1	Sleep Mode Default: Normal Operating Mode Conversion: Sleep mode setting = DB_1 = 0 Normal Operating Mode = 1 Sleep mode = 2 Sleep mode
9	1	Zero Gas Type Default: zero on room air Conversion: zero gas = DB_1 = 0 zero on N_2 = 1 zero on room air Notes: When performing a zero on room air, this setting should be set to room air (the default). Only change to nitrogen (N_2) when performing a zero on 100% N_2 gas; this is provided for use in a laboratory environment.
11	4	Get/Set Gas Compensations DB_1 = O_2 Compensation Default: 16 % Conversion: O_2 compensation = DB_1 Resolution: 1 % (0 – 100 %) DB_2 = Balance gas Default: 0 (room air) Conversion: balance gas = DB_2 = 0, room air = 1, N_2O = 2, Helium DB_3, DB_4 = Anesthetic agent x10 Default: 0.0 % Conversion: Anesthetic agent = [(DB_3 * 2^7) + DB_4] / 10 Resolution: 0.1 % (0.0 – 20.0 %) Notes: Use this setting to correct for the compensation of the gas mixture administered to the patient. Anesthetic agent is ignored when the balance gas is set to helium. Example: An oxygen value of 40%, balance N_2O with 3.5 % anesthetic agent would correspond to the following data byte values: DB_1 = 40, DB_2 = 1, DB_3 = 0, and DB_4 = 35
18	10	Get Sensor Part Number (Get Only) Conversion: DB_1 - DB_{10} are ASCII characters Notes: The sensor part number is a string of 10 ASCII characters that is not NULL terminated.
19	1	Get OEM ID (Get Only)

		Conversion: ID = DB_1 Notes: The ID is a 7-bit identifier which is set at the factory to a unique value for each OEM.
20	5	Get Sensor Serial Number (Get Only) Conversion: Serial Number $= (DB_1 * 2^{28}) + (DB_2 * 2^{21}) + (DB_3 * 2^{14}) + (DB_4 * 2^7) + DB_5$ Notes: This is a 32-bit number that is unique for each BA2xx.
21	3	Get Hardware Revision Number (Get Only) Conversion: DB_1 – DB_3 are ASCII characters. Note: This command returns the current hardware revision level. The revision level is a string of 3 ASCII characters that is not NULL terminated.
23	5	Get Total Use Time (Get Only) Resolution: 1 minute Update frequency: every 5 minutes Conversion: minutes of use $= (DB_1 * 2^{28}) + (DB_2 * 2^{21}) + (DB_3 * 2^{14}) + (DB_4 * 2^7) + DB_5$ Notes: This is the total time the sensor has been in service. The usage time is sent in 1-minute units, however the usage time is updated once every 5 minutes so the usage time will not return contiguous values, as numbers between the five minute updates will be skipped.
24	5	Get Last Zero Time (Get Only) Resolution: 1 minute Update frequency: every 5 minutes Conversion: minutes of use $= (DB_1 * 2^{28}) + (DB_2 * 2^{21}) + (DB_3 * 2^{14}) + (DB_4 * 2^7) + DB_5$ Notes: This is the total time that has elapsed with the sensor in service since the last BA2xx Zero. The time is sent in 1-minute units, however the value is updated once every 5 minutes so values between the five minute updates will be skipped.

7.4. NACK Error (Command C8h)

Response:

C8h - NBF - CEB – CKS

Definitions:

C8h - command identifier
NBF - number of bytes to follow
CEB - command error byte (see Table below)
CKS - checksum byte

Description:

The communications protocol has built-in command error checking. The following

NACK Errors

CEB	NACK Error	Description
0	Bootcode	Waiting for bootloader – Startup only
1	Invalid Command	This occurs whenever a command other than the defined commands is received. It can also occur when a command byte (byte > 80h) is expected and the byte is < 80h.
2	Checksum Error	This occurs whenever an improper checksum is received.
3	Time-out Error	This occurs whenever more than 500 msec elapses between the first and last bytes of a command.
4	Invalid Byte count	This occurs whenever the byte count is less than thenumber of bytes expected for a particular command.
5	Invalid Data Byte	This occurs whenever a non-command byte is expected and a command byte (byte with MSB=1) is encountered.
6-10	System Faulty	This occurs when the system is in a non-functional state due to a system fault. All commands will be ignored. Contact Service
11-19	Not used	Reserved for future use.
20-24	System Faulty	This occurs when the system is in a non-functional state due to a system fault. All commands will be ignored. Contact Service

During normal operation, command errors should not occur. In cases where one of these errors is encountered, the CO2 module will respond by sending the appropriate NACK response.

If system faulty errors are encountered, the BA2xx is in a non-functional state and all commands will be rejected. Check that the sensor is properly plugged in. Reinsert or reset the sensor if necessary. If the error persists, return the sensor to the factory for servicing

7.5. Stop Continuous Mode (Command C9h)

Command:

C9h - NBF – CKS

Command Type:

Single Response Command

Response:

C9h - NBF – CKS

Definitions:

C9h – command
NBF - number of bytes to follow
CKS – checksum

Description:

This command is used to stop the data transmission of a continuous response command. The response is sent as soon as the current process is halted. Any data packet currently being sent will be sent in its entirety before the current continuous response is halted. If the waveform mode command is not active, the Stop Continuous Mode command will send the appropriate response but the command has no effect.

7.6. Get Software Revision (Command CAh)

Command:

CAh - NBF - RF – CKS

Command Type:

Single Response Command

Response:

CAh - NBF - RF - SRB0 - SRB1 - .... - SRBn – CKSUM

Definitions:

CAh - command byte
NBF - number of bytes to follow
RF - revision format: (valid range: 0 – 3, see Revision Strings table below)
SRB0 - SRBn - Maximum of 35 bytes containing ASCII characters.
CKSUM – checksum

Description:

This command returns the current software revision level. The revision level is a string of a maximum of 35 ASCII characters that is not NULL terminated; its length is NBF - 2. The byte RF describes which revision string is requested. The table below describes the available revision string requests.

Revision Strings

RF Byte	Revision Strings
0	Full Software Revision This string describes the current main software revision

7.7. Reset No Breaths Detected Flag (Command CCh)

Command:

CCh - NBF- CKSUM

Command Type:

Single Response Command

Response:

CCh - NBF- CKSUM

Definitions:

CCh - command byte
NBF - number of bytes to follow
CKSUM – checksum

Description:

This command is used to force the system to clear the No Breaths Detected flag. When this command is issued, the status bit for No Breaths Detected is cleared and the system enters a state similar to initial startup. Note that all DPI parameters are also reset. This command can be sent even if the status flag for No Breaths Detected is not set.

7.8. Reset BA2xx (Command F8h)

Command:

F8h - NBF- CKSUM

Command Type:

Single Response Command

Response:

None

Definitions:

F8h - command byte
NBF - number of bytes to follow
CKSUM – checksum

Description:

This command is used to cause a system watchdog reset in the sensor. When this command is issued, the system enters an infinite loop and a watchdog timer resets the system one second later.

Appendix A: Status/Error Bytes

A.1 CO2 Status/Error

The status of the system is encoded in DPI #1 of the CO 2 Waveform Data Mode packet (command 80h).

The status of the system can be determined though two methods: the Extended Status Bytes and the Prioritized Status byte.

The Extended Status Bytes contain the full status information for the BA2xx at the time of transmission. This allows for multiple messages to be displayed by the host simultaneously. For each status condition, a value of '1' signifies the given condition exists. This allows for multiple messages to be displayed by the host simultaneously.

The Prioritized Status Error Byte can be used to simplify the messaging for the host. With this status, only the highest priority message will be transmitted. The host can then use the suggested Message/ Response to indicate to the user how to proceed. There is no correlation between the value byte of the status and its priority in the list.

- The No Respiration Timeout and Breaths Detected flags are only available through the Extended Status Byte and should be monitored.

- Values for the Prioritized Status Error Byte not specified in Table A.4 are reserved for future use by manufacturer. Host software should ignore any undefined Prioritized Status Error Bytes not specified in the table.

There are several levels of status message:

1) Hardware Error – These are severe latched errors that usually require contacting Service for correction.

2) Correctable Error– These messages are used to indicate that a condition exists that may impair the sensor from correctly calculating CO2 values. These messages often require user interaction, such as a BA2xx Zero, to remove the error.

3) Informative – These messages relay to the host the current state of the BA2xx. The can be used by the host to decide if any action is required or to give further detail to an error message.

A.4 CO2 Status/Error

Priority	Value	Suggested Message/Response	Status Type	Description
1 Highest	01h	*"Sensor Over Temp"* Make sure sensor is not exposed to extreme heat (heat lamp, etc.). If error persists, return sensor to factory for servicing	Hardware Error	The sensor temperature is greater than 40 °C.
2	02h	*"Sensor Faulty"* Check that the sensor is properly plugged in. Reinsert or reset the sensor if necessary. If error persists, return sensor to factory for servicing	Hardware Error	One of the following conditions exist: ● Sensor Source Current Failure ● EEPROM Checksum Faulty ● Hardware Error
3	03h	*No message* The host must set the Barometric Pressure and compensations to clear this error; no user intervention should be required.	Correctable Error	Barometric Pressure and/or gas compensations (N2O, O2, helium, and anesthetic agent) have not been set since power on. For CO2 to be calculated with the stated accuracy, these values should be set whenever the Sensor is plugged in
4	04h	Reversed		
5	05h	*"Zero In Progress "*	Informative	A Sensor Zero is currently in progress.
6	06h	*"Sensor Warm Up "* This error condition is normal at startup. This error should clear when the warm up is complete.	Informative	One of the following conditions exist: ● Sensor under temperature ● Temperature not stable ● Source Current unstable
7	0Ah	*"Check Sampling Line"* Check that the sampling line is not occluded or kinked	Correctable Error	This error occurs whenever the pneumatic pressure is outside the expected range
8	07h	*"Zero Required"* To clear, check airway adapter and clean if necessary. If this does not correct the error, perform an adapter zero. If you must adapter zero more than once, a	Correctable Error	One of the following conditions exist: ● Zero Required ● Zero Required: Zero Error

		possible hardware error may exist.		
9	08h	"*CO2 Out of Range"* If error persists, perform a zero.	Correctable Error	The value being calculated is greater than the upper CO2 limit (150 mmHg, 20.0 kPa, or 19.7 %). The maximum value output is the upper CO2 limit
10 Lowest	09h	*"Check Airway Adapter"* To clear, clean airway adapter if mucus or moisture is seen. If the adapter is clean, perform a Sensor zero	Correctable Error	Usually caused when the airway adapter is removed from the Sensor or when there is an optical blockage on the windows of the airway adapter. May also be caused by failure to perform Sensor zero to when adapter type is changed.

Byte 1: Bit	Status/Error	Status Type	Description	Suggested Message/Response
7	Sync	NA	Always zero	NA
6	No Breaths Detected	Informative	This message results whenever the "no breaths detected" timeout condition occurs.	“CO2 No Breaths Detected”
5	In Sleep Mode	Informative	This bit is set when the Sensor has been placed in sleep mode.	“ Sensor in Sleep Mode”
4	CO2 Sensor NOT Ready to Zero	Informative	This bit is set if the CO2 sensor is not ready for a Sensor Zero. If the Zero Required (Byte 2, Bit 1) is set, and this bit is set, one or more of the following conditions may exist: • Breaths detected (Byte 2, Bit 2) • Temperature is not stable (Byte 2, Bits 1,0) • Source Current unstable (Byte 2, Bits 5) • In sleep mode. (Byte 2, Bits 5)	No Message
3	CO2 Out of Range	Correctable Error	The value being calculated is greater than the upper CO2 limit (150 mmHg, 20.0 kPa, or 19.7 %). The maximum value output is the upper CO2 limit.	“*CO2 Out of Range”* If error persists, perform a zero.
2	Breaths Detected	Informative	Breaths have been detected by the Sensor within the last 20 seconds while a Sensor zero was attempted.	No message. Disconnect airway adapter from patient circuit and wait for status to clear before attempting a Sensor Zero.
1	Check Adapter	Informative	Usually caused when the airway adapter is removed from the Sensor or when there is an optical blockage on the windows of the airway adapter. May also be caused by failure to perform a Sensor zero when the adapter type is changed	*“Check Airway Adapter”* To clear, clean airway adapter if mucus or moisture is seen. If the adapter is clean, perform a Sensor zero
0	Negative CO2 Error	Informative	This error occurs when the calculated CO2 is less than zero for a period of time. This can be caused by a Sensor that was zeroed with CO2 in the airway or by an optical blockage of the airway adapter.	*“Check Airway Adapter”* To clear, check airway adapter and clean if necessary. If this does not correct the error, perform an adapter zero

Byte 2: Bit	Status/Error	Status Type	Description	Suggested Message/Response
7	Sync	NA	Always zero	NA

6,5	reversed	NA	Reversed bits	NA
4	Compensation Not Yet Set	Correctable Error	Barometric Pressure or gas compensations have not been set since power on. For CO2 to be calculated with the highest accuracy, these values should be set whenever the Sensor is plugged in or the gas compensations change. When this error bit is active, the Sensor will send end tidal CO2 (ETCO2), Inspired CO2 , and respiratory rate as zeroes.	"Sensor not initialized" Set the Barometric Pressure and gas compensations to clear this error.
3,2	Sensor Calibration Status	Informative	00 – No Zeroing in Progress	No message
			01 – Zeroing in Progress - A Sensor Zero is currently in progress.	*"Zero In Progress "*
			10 – Zero Required - A Sensor Zero is required for the Sensor, for one of the following reasons: • Check Adapter (Byte 1, Bit 1) • Negative CO2 error (Byte 1, Bits 1, 0)	*"Zero Required"* See cause of status bit to see if additional messages should be displayed
			11 – Zero Required: Zero Error - An error was found during Sensor zero. This could occur if the airway adapter is occluded or CO2 gas is present in the adapter.	*"Zero Required"* Perform adapter zero to correct. If you must adapter zero more than once, a possible hardware error may exist
1,0	Sensor Temperature Status	Informative	0 – Stable at Operating Temperature – The sensor temperature is stable and ready for operation	No message
			01 – Below Operating Temperature - The sensor has not reached operating temperature. This condition is typical at power on or after inserting a sensor into the host system.	*"Sensor Warm Up "* If calibration is required while this error is set, the message "Wait for Sensor" can be shown.
			10 - Above Operating Temperature -The internal temperature of the sensor is above the operating temperature.	*"Sensor Over Temp"* Make sure sensor is not exposed to extreme heat (heat lamp, etc.). If error persists, return sensor to factory for servicing
			11 – Temperature Unstable	*"Sensor Warm Up "* If calibration is required while this error is set, the message "Wait for Sensor" can be shown.

Byte 3: Bit	Status/Error	Status Type	Description	Suggested Message/Response
7	Sync	NA	Always zero	NA
6	EEPROM Checksum Faulty	Hardware error	The calibration values in the sensor EEPROM failed the checksum test.	*"Sensor Faulty"* Check that the sensor is properly plugged in. Reinsert or reset the sensor if necessary. If error persists, return sensor to factory for servicing
5	Hardware Error	Hardware error	The sensor has detected a hardware error	*"Sensor Faulty"* Check that the sensor is properly plugged in. Reinsert or reset the sensor if necessary. try to plug the sensor into a different host power supply as this error may result from a faulty sensor or a faulty host supply. If error persists, return sensor to factory for servicing.
4-0	Not used	NA	Reserved for future use (1)	NA

Byte 4: Bit	Status/Error	Status Type	Description	Suggested Message/Response
7	Sync	NA	Always zero	NA
6-4	Not used	NA	Reserved for future use (1)	
3	Pump Off	Informative	This bit is set when the sampling pump is off	No message
2	Pneumatic System Error	Correctable error	This error occurs whenever the pneumatic pressure is outside the expected range	*"Check Sampling Line"* Check that the sampling line is not occluded or kinked.
1	Pump Life exceeded	Informative	This bit is set when the manufacturer stated pump life has been exceeded. Service may be required if Pneumatic System Error is present and can no longer be cleared.	No message
0	Sidestream adapter not detected	Informative	There is no sidestream sampling set connected to the sidestream sensor.	*"Sample Line Disconnected"* Connect a sampling adapter to clear.

A.6 Hardware Status Bytes

Byte 1: Bit	Status/Error	Status Type	Description	Suggested Message/Response
7	Sync	NA	Always zero.	NA
6	Pulse Width Watchdog Error	Hardware Error	A problem has been detected in the pulse width watchdog	*"Sensor Faulty"* Check that the sensor is properly plugged in. Reinsert or reseat the sensor. If error persists, return sensor to factory for servicing.
5	Pulse Width Range Error	Hardware Error	The calculated width of the pulse is out of range.	
4	Source Voltage Range Error	Hardware Error	The calculated source voltage is out of range.	
3	Bias Voltage Range Error	Hardware Error	The calculated bias voltage is out of range.	
2	Volt Voltage Range Error	Hardware Error	The calculated 5 volt voltage is out of range.	
1	Heater Thermistor Error	Hardware Error	Error detected in the heater thermistors.	
0	Software Fault	Hardware Error	A software error or fault has been detected.	

Byte 2: Bit	Status/Error	Status Type	Description	Suggested Message/Response
7	Sync	NA	Always zero.	NA
6	Program RAM checksum error	Hardware Error	The checksum of the program segment stored in RAM does not match the value calculated after loading.	*"Sensor Faulty"* Check that the sensor is properly plugged in. Reinsert or reseat the sensor. If error persists, return sensor to factory for servicing.
5	Main Flash checksum error	Hardware Error	The main flash checksum is corrupt. The checksum value stored in the flash does not match the calculated checksum.	
4	CO2 Warm-up Period Exceeded	Hardware Error	The BA2xx CO2 warm-up period has been exceeded.	
3-0	Not used	NA	Reserved for future use (1)	

NOTE (1) :Do not assume the state of this bit is a particular value. It may be used by manufacturer for internal use.

Appendix B: DPI Reference

B.1 Data Parameter Index (DPI)

DPI = 1 CO2 Status/Errors

Description: This parameter returns a comprehensive bitwise summary of the current CO2 status. For more information on the CO2 status, see Appendix A.
Updated: once a second
Length = 5 bytes
Conversion: Extended Status Byte 1 = DB1
Extended Status Byte 2 = DB2
Extended Status Byte 3 = DB3
Extended Status Byte 4 = DB4
Prioritized Status Byte = DB5

DPI = 2 End-Tidal CO2 (ETCO2)

Description: The end-tidal CO2 value is the highest peak CO2 value of all end of expirations (end of breaths) over the selected time period ETCO2 . If less than two breaths exist in the selected time period, the value will be the maximum ETCO2 value for the last two breaths. The method is determined by the ETCO2 time period setting in the instrument settings (refer to the Get/Set Sensor Settings Command in section 8.3)
Resolution: 0.1 (0 – 150.0 mmHg, 0 – 20.0 kPa, or 0 – 19.7 %)
Updated: once a second
Length = 2 bytes
Conversion =((128 * DB1) + DB2) / 10

DPI = 3 Respiratory Rate, Total (Freq t)

Description: This parameter returns the average respiratory rate for the last eight breaths.
Resolution: 1 br/min (0 - 150 br/min)
Updated: once a second
Length = 2 bytes

Conversion = (128 * DB1) + DB2

DPI = 4 Inspired CO2

Description: The inspired CO2 is the minimum average value of CO2 over the inspiratory period. An inspired CO2 is reported if the level of CO2 greater than or equal to 3 mmHg is detected. The inspired CO2 value reported is a 20 second average.

Resolution: 0.1 (3 – 50.0 mmHg, 0.4 – 6.6 kPa, or 0.4 – 6.6 %)
Updated: once a second
Length = 2 bytes
Conversion = ((128 * DB1) + DB2) / 10

DPI = 5 Breath Detected

Description: This parameter is sent to indicate that the BA2xx has detected a breath. This parameter is sent at the end of expiration of each breath and has no data bytes. It is sent only once per breath.
Updated: Each breath
Length = 0 bytes

Appendix C:Specifications

Module	BA210
Transducer Type	Mainstream CO_2 Sensor
Principle of Operation	Non-dispersive infrared(NDIR), dual wavelength, no moving parts
Initialization Time	Capnogram displayed in less than 5 seconds，full specifications within 2 minutes
CO_2 Measurement Range	0 – 114 mmHg, 0 – 15 %, 0 – 15.2 kPa(760mmHg)
Rise Time	Less than 60 ms - Adult Reusable or Single-Patient-Use Airway Adapter Less than 60 ms - Infant Reusable or Single-Patient-Use Airway Adapter
CO_2 Resolution	0.1mmHg 0 to 59 mmHg 0.25 mmHg 60 to 114mmHg
CO_2 Accuracy	0 – 40mmHg ±2 mmHg 41 – 76 mmHg ±5% of reading 77 – 114 mmHg ±8% of reading Above 80 BPM ±12% of reading
CO_2 Stability	Short term drift: Drift over four hours shall not exceed 1 mmHg maximum Long term drift: Accuracy specification will be maintained over a 120-hour period
Sampling Frequency	100 Hz
Respiration Rate Range	2 to 150 Breaths Per Minute (BPM)
Respiration Rate Accuracy	±1 breath
Compensations (Supplied)	Barometric pressure: 400 mmHg to 800 mmHg Operator selectable O2, N2O, He and agent compensation
Calibration	No routine user calibration required. An airway adapter zero is required when changing
Airway Adapters	Single-patient-use or reusable, < 5 cc dead space (adult), < 1 cc dead space infant
Voltage Requirements	5.0 VDC ±5%
Power Consumption	1.2Watts typical (Steady State) Up to 1.8 Watts maximum on power up (Warm up)
Interconnection	Connector Type: Lemo Redel 8-pin plastic connector Sensor Plug: PAB.M0.8GL.AC39GZ Host Receptacle: PKB.M0.8GL.L Pin out: 1. VA 5.0VDC 2. Shield Shield 3. DGND Digital return 4. VSRC 5.0VDC 5. TxD Serial data from Sensor 6. RxD Serial data from Host 7. AGND Analog return 8. SYNC Unused
Data Interface	RS232 , bi-directional, 19200 baud rate, standard N-8-1.
Data Output	CO_2 gas concentration (mmHg), End-tidal CO_2, Inspired CO_2, Respiratory Rate. Gas and barometric pressure compensated when supplied by host.
Temperature and Humidity	Operating: 0 to 45℃, 10 to 90% RH, non-condensing

	Storage: -40 to 70℃, <90% RH, non-condensing
Water Resistance	IPX4 – Splash-proof (sensor head only)