Interface and Data Format Specification for sensors

(v3 sensor boards with Alpha sensor)

Waggle Group

Waggle Sensor Array

November 2016, Version 1.01

Contents

1 Physical Connections and Interfaces

'v3 sensor boards with alpha sensor' means a set of sensors which includs a v3.1 Metsense board, a v3.1 Lightsense board, a Chemsense board, and a alpha sensor.

Physical connections between sensor boards and alpha sensor are shown in the Figure 1. The metsense board is connected to the lightsense board and the chemsense board through CAT6 cable. The metsense and the lightsense deliver data through I2C communication, and chemsense board delivers data through serial3 communication. All sensor data from metsense, lightsense, and chemsense board are delivered to nodecontroller thourgh USB connect attached on lightsense board. Alpha sensor will be conncted to Airsense board using SPI communication line and a digital pin 1 of JP2. Then all the sensor data will be delivered through the USB connected on lighsense board.

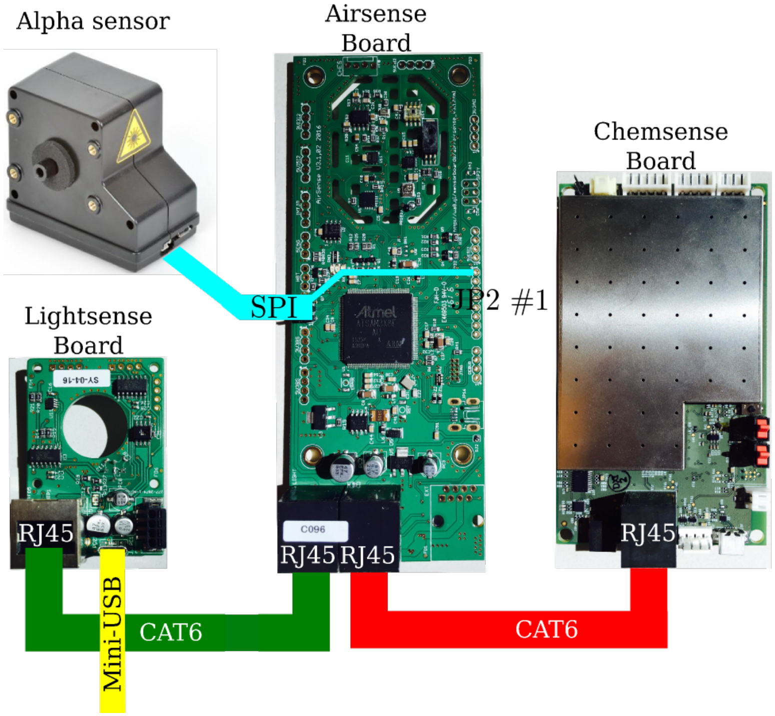

Figure 1: Connections between the sensor boards and the sensor

2 Data Transmission

The data from the sensor boards is sent as a formatted unit of data – a transmission packet. A transmission packet is composed of several data sub-packets, each of which carries information pertaining to the parameter. The transmission packet format and the data sub-packets are described here.

2.1 Transmission Packet

A transmission packet can be separated into 6 segments. The structure of the transmission packet relies on positions of Bytes and predefined values for those Byte segments. Table 1 below illustrates how the segments are organized in a transmission packet.

Preamble	**Seq. | Prot. Ver.**	**Data Length**	**Data**	**CRC**	**Postscript**
1st Byte	2nd Byte	3rd Byte	next Bytes up to 251 Bytes	Penultimate Byte	Final Byte

Table 1: Transmission Packet structure

The first segment is the start byte, or the preamble. The preamble is followed by the packet sequence number and protocol version, each of which are 4 bits long and are together packed into a single byte. Next, one byte field that reports the length of the data which follows it immediately. The data segment is followed by a single CRC byte, and finally the packet ends with a one byte postscript. Table 2 lists the packet and the static values, if any, for each of the segments.

Field	**Value**	**Segment**	**Length**
Preamble	0xAA	1	1 Byte
Packet Sequence Number	Variable	2	1 Nibble
Protocol version	0x00		1 Nibble
Length of data (not whole packet)	Variable	3	1 Byte
Data	Variable	4	Variable
CRC of data (not whole packet)	Variable	5	1 Byte
Postscript	0x55	6	1 Byte

Table 2: Transmission Packet Segments

2.2 Data Sub-packets

The data segment of the transmission packet is further separated into many sub-packets. Table 3 below shows the organization of a sub-packet. The sub-packet starts with a source identifier. One bit validity field and seven bits "length of the sub-packet" field are packed together as the next byte. The length field counts the number of bytes following it which make up the sub-packet. The validity bit is set to 1 if the sensor reading is valid and set to 0 if the sensor is dead, disabled, unconnected, unresponsive or if data could not be collected from the sensor in the time window. The size of the sub-packet is restricted to 127 Bytes by the seven bits length field.

The packet validity is initially 0, and it will be changed to 1 when each sub-packet gets sensor value from sensor. And after a transmission packet is trasmitted, the validity becomes 0 again. When validity is set to 0, if the sensor is dead, disabled, unconnected, unresponsive or data could not be collected, the particular invalid sub-packet is not packed into a transmission packet.

Source ID	**1-bit Validity [0: invalid, 1: valid]|| 7-bits Data Length**	**Data**
1 Byte	1 Byte	up to 127 Bytes

Table 3: Transmission Packet Segments

2.3 Data Packer CRC

To validate the data transmitted from the sensor board, a CRC value for the data is calculated and transmitted as part of the data packet. The Maxim 1-Wire CRC polynomial is used for calculating the CRC. On receiving the data packet, the CRC of the data packet is recalculated and compared with the value transmitted as part of the packet. If the two CRC values match, the transmission is error-free. The equivalent polynomial function of the CRC is shown in Equation 1.

$$CRC = x^8 + x^5 + x^4 + 1 \tag{1}$$

Further description of the Maxim 1-Wire CRC is available in Maxim Application Note 27. Below are the Python and C implementations of the CRC calculator. The CRC implementations below take a data Byte and the previous CRC as inputs, and return the new CRC as return value.

Python Code:

```
def calc_crc (data_Byte,CRC_Value)
    CRC_Value = ord(data_Byte) ^ CRC_Value
    for j in range(8):
    if (CRC_Value  & 0x01):
        CRC_Value  = (CRC_Value  >> 0x01) ^ 0x8C
    else:
        CRC_Value  =  CRC_Value  >> 0x01
return CRC_Value
```

C Code:

```
unsigned char  CRC_CALC (unsigned char data, unsigned char crc)
{
        unsigned char i;
        crc ^= data;
        for (i=0x00; i < 0x08; i++)
        {
                if (crc & 0x01) { crc = (crc >> 0x01)^0x8C; }
                else { crc =  crc >> 0x01; }
        }
        return(crc);
}
```

3 Data Formats

The data sent in each sub-packet is encoded in one or more formats. Currently we define eight formats for various types of data including integers, bytes, and floating point numbers. The numerical range of these representations is restricted to within the bounds of values that we expect from the various sensors and other sources. Thus the encoding schemes are specifically designed to effectively and efficiently encode the values expected in the sensor streams. The eight formats, and the encoding schemes are listed below in Table 4.

Format	Number of Bytes Used	Value Represented	Value Range
1	2	unsigned int_16 input	[0, 65535]
2	2	int_16 input	[−32767, 32767]
3	6	byte input[6]	[0x00, 0xFFFFFFFFFFFF]
4	3	unsigned long_24 input	[0, 16777215]
5	3	long_24 input	[−8388607, 8388607]
6	2	float input	[−127.99, 127.99]
7	4	byte input[4]	[0x00, 0xFFFFFFFF]
8	2	float input	[−0 – 31.999]

Table 4: Data formats

Format 1:

8 Data Bits	8 Least Significant Bits
Byte[0]	Byte[1]

Table 5: format 1

Format 2:

Sign Bit | 7 Most Significant Bits	8 Least Significant Bits
Byte[0]	Byte[1]

Table 6: format 2

Format 3:

Array[0]	Array[1]	Array[2]	Array[3]	Array[4]	Array[5]
Byte[0]	Byte[1]	Byte[2]	Byte[3]	Byte[4]	Byte[5]

Table 7: format 3

Format 4:

8 Most Significant Bits	Bits 15 – 8	8 Least Significant Bits
Byte[0]	Byte[1]	Byte[2]

Table 8: format 4

Format 5:

Sign Bit | 7 Most Significant Bits	Bits 15 – 8	8 Least Significant Bits
Byte[0]	Byte[1]	Byte[2]

Table 9: format 5

Format 6: The leading bit of the Byte 1 is always set to 0

Sign Bit| 7 bit representation of Integer part	0 | 7 bit representation of the Fractional part
Byte[0]	Byte[1]

Table 10: format 6

Format 7:

Array[0]	Array[1]	Array[2]	Array[3]
Byte[0]	Byte[1]	Byte[2]	Byte[3]

Table 11: format 7

Format 8: Only 3 fractional places are allowed

Sign Bit| 5 bit representation of Integer | 2 MSBs of fraction	8 LSBs of the fraction
Byte[0]	Byte[1]

Table 12: format 8

Sign bit:

Positive Number	0
Negative Number	1

Table 13: Sign bit

4 Sub-packets

As shortly explained in document section 2.2, data sub-packets are generated depending on its designated data format and length when data reading from each sensor if valid. The first byte of the sub-packet is sensor ID for each parameter, and the second byte means validity of the packet and length of the sensor data as shown in Table 3. Detail of sub-packet and sensor data will be explined in this section.

4.1 Parameters

The sensor boards output a set of parameters which are identified by a unique ID. Each parameter has a set of values associated with it which are encoded in an appropriate data format. The table below lists the various parameters produced by the sensor boards, the unique source ID used to identify them, the values produced by them and the format in which the value is encoded.

Each parameter and its values are composed into a sub-packet based on the format described in document section 2.2. In the case of parameters with 2 or more values, the encoded values are arranged in the sub-packets sequentially.

Table 14: Data sub-packet structure (each row is a "chunk")

Parameter	Source ID	Values	Formats
Firmware version	0xFD	Firmware version (HW/SW)	Bit mask
		Build time	
		Build git	
Airsense board			
Airsense/Lightsense MAC address	0x00	MAC Address	Format 3
TMP112	0x01	Temperature	Format 6
HTU21D	0x02	Temperature	Format 6
		relative humidity	
BMP180	0x04	Temperature	Format 6
		Pressure	Format 4
PR103J2	0x05	Temperature	Format 1
TSL250RD	0x06	Visible Light	Format 1
MMA8452Q	0x07	Acceleration in X	Format 6
		Acceleration in Y	
		Acceleration in Z	
		Vibration	

Table14 – continued from previous page

Parameter	Source ID	Values	Formats
SPV1840LR5H-B	0x08	RMS Sound Level	Format 1
TSYS01	0x09	Temperature	Format 6
Lightsense board			
HMC5883L	0x0A	Magnetic Field in Z	Format 8
		Magnetic Field in Y	
		Magnetic Field in Z	
HIH6130	0x0B	Temperature	Format 6
		relative humidity	
APDS-9006-020	0x0C	Ambient light intensity	Format 1
TSL260RD	0x0D	IR intensity	Format 1
TSL250RD	0x0E	Visible light intensity	Format 1
MLX75305	0x0F	Light	Format 1
ML8511	0x10	UV intensity	Format 1
TMP421	0x13	Temperature	Format 6
SPV1840LR5H-B	0x14	RMS Sound Level	Format 1
Chemsense board			
Total reducing gases	0x15	Raw Concentration	Format 5
Nitrogen dioxide	0x17		
Ozone	0x18		
Hydrogen sulphide	0x19		
Total oxidizing gases	0x1A		
Carbon monoxide	0x1B		
Sulfur dioxide	0x1C		
SHT25	0x1D	Temperature	Format 2
		relative humidity	
LPS25H	0x1E	Temperature	Format 2
		Pressure	Format 4
Si1145	0x1F	UV intensity	Format 1
		Visible light intensity	
		IR intensity	
Chemsense MAC address	0x20	MAC Address	Format 3
CO ADC temp	0x21	ADC temperature	Format 2
IAQ IRR ADC temp	0x22		

Table14 – continued from previous page

Parameter	Source ID	Values	Formats
O3 NO2 ADC temp	0x23	ADC temperature	Format 2
SO2 H2S ADC temp	0x24		
CO LMP temp	0x25		
Accelerometer	0x26	Acceleration in X	Format 2
		Acceleration in Y	
		Acceleration in Z	
		Vibration	Format 4
Gyro	0x27	Orientation in X	Format 2
		Orientation in Y	
		Orientation in Z	
		Orientation Index	Format 4
Alpha Sensor			
Histogram	0x28	Bin count	Raw reading
		Average Time	
		Sample flow rate	
		Temp/Pressure(alther)	
		Sampling period	
		Sum of the counts	
		PM 1	
		PM 2.5	
		PM 10	
Firmware	0x29	Firmware version	
Configuration A	0x30	Bin Boundaries	
		Bin Particle Volumes A	
Configuration B	0x31	Bin Particle Volumes B	
		Bin Particle Densities A	
Configuration C	0x32	Bin Particle Densities B	
		Bin Sample Volume Weightings A	
Configuration D	0x33	Bin Sample Volume Weightings B	
		Gain Scaling Coefficient	
		Sample Flow Rate	
		Laser DAC and Fan DAC	
		Conversion factor	
		Space Bytes	

4.2 Data packets

The context of each parameter, its utility and the arrangement of its values is described below. In all the tables below, the validity bit is set to 1, which means the data is valid. The parameter descriptions below are aggregated based on the sensor-board they are situated on - Metsense, Lightsense and Chemsense.

4.2.1 Firmware Version

This is a 8 bytes version information that identifies hardware version, software version, and build information of the waggle node. The build time and the build git are included to varify the effectiveness of the software. Firmware version is bit masked and encoded through format 1, and build git is encoded through format 1.

0xFD	**0x88**	**Firmware version in Format 1**	**Build time**	**Build git in Format 1**
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 7]	Bytes[8 – 9]

Table 15: Sub-packet of Firmware version

3 bit major HW ver. | 3 bit minor HW ver. | 2 bit major SW ver.	Byte[2]
2 bit major SW ver. | minor SW ver. × 10 + sub SW ver.	Byte[3]

Table 16: Firmware version

4.2.2 Metsense

• **Metsense/Lightsense MAC address:** This is a six byte ID that uniquely identifies each Airsense board. This MAC address is also applied to each Lightsense board which has the same board number. The ID is provided by a DS2401 1-Wire DSN chip. The 1-byte family ID and CRC provided by the DSN chip are omitted, and the rest six bytes are used as the Unique ID. The Unique ID uses Format 3 for encoding and the arrangement is listed below.

0x00	**0x86**	**MAC address in Format 3**
Byte[0]	Byte[1]	Bytes[2 – 7]

Table 17: Sub-packet of Met/lightsense board MAC address

• **TMP112, TSYS01:** TMP112 and TSYS01 are digital temperature sensors, which provides the temperature values in centigrade.

Sensor ID (0x01 or 0x09)	**0x82**	**Temperature in Format 6**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 18: Sub-packet of a temperature sensor, TMP112

- **HTU21D:** HTU21D is a digital temperature and relative humidity sensor, which provides relative humidity value (%RH) and temperature value in centigrade.

0x02	**0x84**	**Temperature in Format 6**	**Relative humidity in Format 6**
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 5]

Table 19: Sub-packet of a temperature and relative humidity sensor, HTU21D

- **BMP180:** BMP180 is an digital temperature and barometric pressure sensor, which provides temperature in centigrade and pressure in hectopascals.

0x04	**0x84**	**Temperature in Format 6**	**Barometric pressure in Format 4**
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 5]

Table 20: Sub-packet of a temperature and barometric pressure sensor, BMP180

- **PR103J2:** PR103J2 is an analog temperature sensor whose resistance changes with change in temperature. The sensor is implemented in a voltage divider circuit, and the center-tap voltage is converted and packed into Format 1 using a 10-bit ADC.

0x05	**0x82**	**Voltage output in Format 1**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 21: Sub-packet of a temperature sensor, PR103J2

- **TSL250RD:** TSL250RD is an analog visible light sensor that produces an analog voltage that is representative of the irradiance measured in μW/cm^2. The output voltage of the sensor is converted and packed into Format 1 using a 10-bit ADC.

0x06	**0x82**	**Voltage output in Format 1**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 22: Sub-packet of a light intensity sensor, TSL250

- **MMA8452Q:** MMA8452Q is a digital three-axis accelerometer. The accelerations in three orthogonal directions, x, y and z, as a multiple of acceleration due to gravity (g) are obtained from the sensor, and a vibration value (represented as multiple of g) is calculated using high-frequency time series data from the three axis.

0x07	**0x88**	Acc(x) in Format 6	Acc(y) in Format 6	Acc(z) in Format 6	Vib. in Format 6
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 5]	Bytes[6 – 7]	Bytes[8 – 9]

Table 23: Sub-packet of a three-axis accelerometer, MMA8452Q

- **SPV1840LR5H-B:** SPV1840LR5H is a MEMS microphone that is sampled at high frequency to obtain the peaks and calculate the sound intensity for a time window. The raw calculated intensity is represented as a 16-bit integer value using Format 1.

0x08	**0x82**	**Voltage output in Format 1**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 24: Sub-packet of a sound level sensor, SPV1840LR5H-B

4.2.3 Lightsense

- **HMC5883L:** HMC5883L is a digital three-axis magnetometer. The magnetic field strengths in three orthogonal directions, x, y and z are obtained from the sensor.

0x0A	**0x86**	**Strength Hx in Format 8**	**Strength Hy in Format 8**	**Strength Hz in Format 8**
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 5]	Bytes[6 – 7]

Table 25: Sub-packet of a three-axis magnetometer, HMC5883L

- **HIH6130:** HIH6130 is a digital temperature and relative humidity sensor, which provides relative humidity value (%RH) and temperature value in centigrade.

0x0B	**0x84**	**Temperature in Format 6**	**Relative Humidity in Format 6**
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 5]

Table 26: Sub-packet of a temperature and relative humidity sensor, HIH6130

- **APDS-9006-020, TSL260, TSL250, MLX75305, and ML8511:** APDS-9006-020, TSL260, TSL250, MLX75305, and ML8511 are analog light intensity sensors that produce the analog voltage representing the general luminance or the irradiance measured in μW/cm^2. The output voltage of the sensor is converted and packed into Format 1 using a 16-bit ADC.

Sensor ID (0x0C ∼ 0x10)	**0x82**	**Voltage output in Format 1**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 27: Sub-packet of light intensity sensors, APDS-9006-020, TSL260, TSL250, MLX75305, and ML8511

- **TMP421:** TMP421 is a digital temperature sensor, which provides the temperature values in centigrade.

0x13	**0x82**	**Temperature in Format 6**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 28: Sub-packet of a temperature sensor, TMP421

- **SPV1840LR5H-B** SPV1840LR5H is a MEMS microphone that is sampled at high frequency to obtain the peaks and calculate the sound intensity for a time window. The raw calculated intensity is represented as a 16-bit integer value using Format 1.

0x14	**0x82**	**Voltage output in Format 1**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 29: Sub-packet of a sound level sensor, SPV1840LR5H-B

4.2.4 Chemsense:

- **Chemical sensors – Total reducing gases, Nitrogen dioxide, Ozone, Hydrogen sulphide, Total oxidizing gases, Carbon monoxide, and Sulfur dioxide:** These parameters provide the current output of the electrochemical ToR, NO_2, O_3, H_2S, ToX, CO, and SO_2 sensor. Each of the cell current is quantified using an AFE that uses a 24-bit ADC to convert it into a signed digital value. This value is represented in Format 5.

Sensor ID (0x15 ∼ 0x1C)	**0x83**	**Raw concentartion in Format 5**
Byte[0]	Byte[1]	Bytes[2 – 4]

Table 30: Sub-packet of a temperature sensor, TMP421

- **SHT25:** SHT25 is a temperature and relative humidity sensor. 100th of temperature and relative humidity are encoded in Format 2.

0x1D	0x84	100th of Temperature in Format 2	100th of Relative humidity in Format 2
Byte[0]	Byte[1]	Bytes[2 – 4]	Bytes[5 – 6]

Table 31: Sub-packet of a temperature and relative humidity sensor, SHT25

- **LPS25H:** LPS25H is a temperature and barometric pressure sensor. 100th of temperature and barometric pressureare encoded in Format 2 and Format 4 respectively.

0x1E	0x85	100th of Temperature in Format 2	barometric pressure in Format 4
Byte[0]	Byte[1]	Bytes[2 – 4]	Bytes[5 – 6]

Table 32: Sub-packet of a temperature and barometric pressure sensor, LPS25H

- **Si1145:** Si1145 is a light sensor for three factors; ultra-violet, visible, and infrared. The raw output from the sensor is encoded Format 1.

0x1F	0x86	Raw UV in Format 1	Raw visible light in Format 1	Raw IR in Format 1
Byte[0]	Byte[1]	Bytes[2 – 4]	Bytes[5 – 6]	Bytes[7 – 8]

Table 33: Sub-packet of an ultra-violet sensor, Si1145

- **Chemsense MAC address:** This is a six byte ID that uniquely identifies each Chemsense board. The Unique ID uses Format 3 for encoding and the arrangement is listed below.

0x20	0x86	Chemsense board MAC address in Format 3
Byte[0]	Byte[1]	Bytes[2 – 7]

Table 34: Sub-packet of Chemsense MAC address

- **ADC Temperatures – CO ADC Temp, IAQ/IRR ADC Temp, O3/NO2 ADC Temp, SO2/H2S ADC Temp, and CO LMT Temp:** Chemsense board measures temperature of sensor ADCs. This includes five parameters and all of them give ADC temperature in 100ths of degree celsious. Format 2 is used for encoding and the arrangement is listed below.

Sensor ID (0x21 ∼ 0x25)	0x82	Temperature in Format 2
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 35: Sub-packet of ADC Temperatures

- **Accelerometer:** The accelerations in three orthogonal directions, x, y and z, as a multiple of acceleration are obtained from the sensor, and a vibration index is calculated. Acceleration data are encoded in Format 2, and vibration index is encoded in Format 4.

0x26	0x89	A(x) in Format 2	A(y) in Format 2	A(z) in Format 2	Vib. in Format 4
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 5]	Bytes[6 – 7]	Bytes[8 – 10]

Table 36: Sub-packet of Accelerometer

- **Gyro:** The gyro in three orthogonal directions, x, y and z, as a multiple of acceleration are obtained from the sensor, and a orientation index is calculated. Gyro data are encoded in Format 2, and orientation index is encoded in Format 4.

0x27	0x89	O(x) in Format 2	O(y) in Format 2	O(z) in Format 2	Index in Format 4
Byte[0]	Byte[1]	Bytes[2 – 3]	Bytes[4 – 5]	Bytes[6 – 7]	Bytes[8 – 10]

Table 37: Sub-packet of Gyro

4.2.5 Alpha Sensor:

In this document, concise Alpha sensor information is provided. For more information, refer to 'Firmware commands version 18.xls' and 'OPC-N2 Manual Issue December 2015.pdf' at 'waggle/docs/alphasense-opc-n2'.

- **Histogram** This parameter provides various information as listed below. **All data from the alpha sensor is LSB first**.

0x28	0xBE	Various sensor information as listed above
Byte[0]	Byte[1]	Bytes[2 – 61]

Table 38: Sub-packet Alpha sensor histogram

  - Bin Counts (Bin0 - Bin15) are unsinged 16 bit integer variables (Bytes[2 – 33]).
  - An unsigned 8-bit integer represents the average amount of time that particles sized in the stated bin took cross the laser beam of the sensor. Value 10 represents 3.33 μs (Bytes[34 – 37]).
  - The sample flow rate in ml/s is provided as a float variable (Bytes[38 – 41]).
  - Temperature and Pressure alternating. Temperature is an unsigned 32-bit integer that represents temperature in Celsious multiplied by 10. Pressure is an unsigned 32-bit integer that represents pressure in pascals (Bytes[42 – 43]).
  - Actual sampling period of a measure of the historgam in seconds provided as 4 bytes of float variables (Bytes[44 – 47]).
  - The least significat 16-bit of the sum of the counts in all the histogram bins is provided by unsigned 16bit integers (Bytes[48 – 49]).
  - A float variable occupying 4 bytes for PM1. Unit is μg/m^3 (Bytes[50 – 53]).
  - A float variable occupying 4 bytes for PM2.5. Unit is μg/m^3 (Bytes[54 – 57]).

- A float variable occupying 4 bytes for PM10. Unit is μg/m^3 (Bytes[58 – 61]).

Parameter	**Data location**
Bin Count (32 Bytes)	Bytes[2 - 33]
Average Time (4 Bytes)	Bytes[34 – 37]
Sample flow rate (4 Bytes)	Bytes[38 – 41]
Temp/Pressure(alter) (2 Bytes)	Bytes[42 – 43]
Sampling period (4 Bytes)	Bytes[44 – 47]
Sum of the counts (2 Bytes)	Bytes[48 – 49]
PM1 (4 Bytes)	Bytes[50 – 53]
PM2.5 (4 Bytes)	Bytes[54 – 57]
PM10 (4 Bytes)	Bytes[58 – 61]

Table 39: Detail sub-packet of Alpha sensor histogram

- **Firmware** This parameter provides the Firmware version of the alpha sensor by unsigned 8-bit integer.

0x29	**0x82**	**Raw values**
Byte[0]	Byte[1]	Bytes[2 – 3]

Table 40: Sub-packet of Alpha sensor firmware

- **Configuration** This parameter provides various information as listed below. Since this configuration data is too long to fit into a sub-packet, the data seperated into four sections to be packed into four sub-packets. Thus each of sub-packet takes 64 Bytes in length.

Source ID (0x30 ∼ 0x33)	**0xC0**	**Various sensor raw values as listed above**
Byte[0]	Byte[1]	Bytes[2 – 65]

Table 41: Sub-packet of Alpha sensor configuration

  - Bin Boundaries (BB0 - BB14) are unsinged 16 bit integer variables, and two spare bytes (Bytes[0 – 31] from Alpha sensor, Bytes[2 – 33] on sub-packet A).
  - Bin Particle Volumes (BPV0 - BPV15) are float variables occupyting 4 bytes each (Bytes[32 – 95] from Alpha sensor, Bytes[34 – 65] on sub-packet A and Bytes[2 – 33] on sub-packet B).
  - Bin Particle Densities (BPD0 – BPD15) are float variables occupying 4 bytes each (Bytes[96 – 159] from Alpha sensor, Bytes[34 – 65] on sub-packet B and Bytes[2 – 33] on sub-packet C).

- Bin Sample Volume Weightings (BSVW0 – BSVW15) are float variables occupying 4 bytes each (Bytes[160 – 223] from Alpha sensor, Bytes[34 – 65] on sub-packet C and Bytes[2 – 33] on sub-packet D).
- Gain Scaling Coefficient (GSC) is float variable occupying 4 bytes (Bytes[224 – 227] from Alpha sensor, Bytes[34 – 37] on sub-packet D).
- Sample flow rate is a float variable occupying 4 bytes (Bytes[228 – 231] from Alpha sensor, Bytes[38 – 41] on sub-packet D).
- Laser DAC value is unsigned 8bit interger variable (Byte[232] from Alpha sensor, Byte[42] on sub-packet D).
- Fan DAC value is unsigned 8bit integer variable (Byte[233] from Alpha sensor, Byte[43] on sub-packet D).
- Time of Flight to Sample Flow Rate conversion factor is unsigned 8bit integer variable (Byte[234] from Alpha sensor, Byte[44] on sub-packet D).
- 21 spare bytes follow Configuration variables (Bytes[235 – 255] from Alpha sensor, Bytes[45 – 65] on sub-packet D).

Parameter	**Data location**
Configuration packet A (Sensor ID 0x30)	
Bin Boundaries (32 Bytes)	Bytes[2 – 33]
Bin Particle Volumes A (32 Bytes)	Bytes[34 – 65]
Configuration packet B (Sensor ID 0x31)	
Bin Particle Volumes B (32 Bytes)	Bytes[2 – 33]
Bin Particle Densities A (32 Bytes)	Bytes[34 – 65]
Configuration packet C (Sensor ID 0x32)	
Bin Particle Densities B (32 Bytes)	Bytes[2 – 33]
Bin Sample Volume Weightings A (32 Bytes)	Bytes[34 – 65]
Configuration packet D (Sensor ID 0x33)	
Bin Sample Volume Weightings B (32 Bytes)	Bytes[2 – 33]
Gain Scaling Coefficient (4 Bytes)	Bytes[34 – 37]
Sample Flow Rate (4 Bytes)	Bytes[38 – 41]
Laser DAC (1 Byte)	Byte[42]
Fan DAC (1 Byte)	Byte[43]
Conversion factor (1 Byte)	Byte[44]
Spare Bytes	Bytes[45 – 65]

Table 42: Detail sub-packet of Alpha sensor configuration

5 Sensor Data Units

5.1 Raw and Processed

The sensor boards output a set of values which have various units for the data. The table below lists the various units of sensor values. ‘Raw Units’ in the table means the unit of the packtized data, which is you can get directly from the packet, and ‘Processed Units’ means the unit which can be used after data conversion through designated equations. The equations will be provided comming subsections.

Table 43: Sensor units both in raw and processed format

Sensor/Parameter	**Raw Units**	**Processed Units**	**Comments**
Firmware version	No Units		*See appendix A
	Airsense board		
Air/Lightsense MAC	No Units	No Units	
TMP112	°C	°C	
HTU21D	°C, %RH	°C, %RH	
BMP180	°C, Pa	°C, Pa	
PR103J2	integer	°C	
TSL250RD	integer	μw/m^2	
MMA8452Q	g, g, g, g	g, g, g, g	
SPV1840LR5H-B	integer		
TSYS01	°C	°C	
	Lightsense board		
HMC5883L	G, G, G	G, G, G	
HIH6130	°C, %RH	°C, %RH	
APDS-9006-020	integer	lux	
TSL260RD	integer	μw/m^2	
TSL250RD	integer	μw/m^2	
MLX75305	integer	μw/m^2	
ML8511	integer	UV index	
TMP421	°C	°C	
	Chemsense board		
Total reducing gases	AFE ADC counts		Raw ADC reading
Nitrogen dioxide			
Ozone			

AFE ADC counts Raw ADC reading

Table 43 – continued from previous page

Sensor/Parameter	Raw Units	Processed Units	Comments
Hydrogen sulphide			
Total oxidizing gases			
Carbon monoxide			
Sulfur dioxide			
SHT25	100ths of °C / %RH	°C, %RH	
LPS25H	100ths of °C, Pa	°C, Pa	
Si1145	Three fixed dummy value		Uncompleted FW
Intel MAC address	No Units	No Units	
CO ADC temp	100ths of °C	°C	
IAQ IRR ADC temp			
O3 NO2 ADC temp			
SO2 H2S ADC temp			
CO LMP temp			
Accelerometer	raw register		Raw reading
Gyro			
Alpha sensor			
Histogram			
Bin count	raw integer		
Average time	raw integer		value 10 = 3.33 μs
Sample flow rate	ml/s		
Temp/Pressure(alter)	10ths of °C / Pa (alter)		
Sampling period	raw float		
Sum of the counts	raw integer		
PM1	μg/m^3		
PM2.5	μg/m^3		
PM10	μg/m^3		
Firmware			
Firmware	raw integer		
Configuration			
Configuration Packet A (Source ID 0x30)			
Bin boundaries	raw integer		
Bin particle Volumes A	raw float		
Configuration Packet B (Source ID 0x31)			

Table 43 – continued from previous page

Sensor/Parameter	Raw Units	Processed Units	Comments
Bin particle Volumes B	raw float		
Bin particle Densities A			
Configuration Packet C (Source ID 0x32)			
Bin particle Densities B	raw float		
Bin sample Volume Weightings A			
Configuration Packet D (Source ID 0x33)			
Bin sample Volume Weightings B	raw float		
Gain scaling Coefficient			
Sample flow Rate			
Laser DAC	raw integer		
Fan DAC			
Conversion factor			
Spare bytes			

5.2 conversion processure

5.2.1 Airsense:

- **TMP112, HTU21D, BMP180, MMA8452Q, TSYS01:** Raw outputs from the sensor boards for the sensors (TMP112, HTU21D, HIH4030, BMP180, MMA8452Q, and TSYS01) are the designated type of sensor value.

- **PR103J2:** Output of PR103J2 is an interger indicating output voltage from the sensor, which is mapped into integer values between 0 and 1023 with voltage range 0 to 3.3V. The raw integer value can be converted to resistance value through the equations below. The resistance value is needed to find corresponding temperature in a resistance-temperature look-up table (PR103J2 R-T table).

$$\text{resistance }(\Omega) = 47000 \times \left(\frac{1023}{\text{raw integer}} - 1\right)$$

- **TSL250RD:** Output of TSL250RD in airsense board is an interger indicating output voltage from the sensor, which is mapped into integer values between 0 and 1023 with voltage range 0 to 3.3V. The raw interger value can be converted to irradiance of visible light in micro-watt per square meter through equations below.

$$\text{irradiance } (\mu W/m^2) = \frac{\text{raw integer} \times 3.3}{1023} \times \frac{1}{0.064}$$

- **SPV1840LR5H-B:** Output value of SPV1840LR5H-B is an interger indicating amplified output voltage from the sensor, which is mapped into integer values between 0 and 1023. The raw output need to be converted to sound level in decibel (dB).

5.2.2 Lightsense

- **HMC5883L, HIH6130, and TMP421:** Raw outputs from the sensor boards for the sensors (HMC5883L, HIH6130, and TMP421) are the designated sensor value.

- **Light sensors using MCP3426 (Multiplexer) – APDS-9006-020, TSL260RD, TSL250RD, MLX75305, ML8511:** Packetized data of the ligth sensors (APDS-9006-020, TSL260RD, TSL250RD, MLX75305, and ML8511) are raw integer proportional to the output voltage from the sensor. The raw integers can be converted to irradiance through equations below.

All the sensor data coming through a common multiplexer and voltage divider, to the voltage output from the sensor is needed to calculate as shown below.

$$\text{output voltage } (V) = \text{output voltage} \times 0.0000625 \times \frac{5}{2}$$

- APDS-9006-020

  Raw output value of APDS-9006-020 is an analog voltage which is proportional to the irradiance. The output voltage can be converted irradiance in lux through the equation below.

$$\text{irradiance (lux)} = \frac{\text{output voltage}}{0.001944}$$

- TSL260RD

  Raw output value of TSL260RD is an analog voltage which is inverse proportional to the irradiance. The output voltage can be calculated though the equation below. Dark voltage is the output voltage at dark condition, and it is an unique parameter of each sensor, so that the dark voltage can be changed for individual sensor.

$$\text{irradiance } (\mu W/m^2) = \frac{\text{output voltage} - \text{dark voltage}}{0.058}$$

- TSL250RD

  Raw output value of TSL250RD is an analog voltage which is inverse proportional to the irradiance. The output voltage can be calculated though the equation below. Dark voltage is the output voltage at dark condition, and it is an unique parameter of each sensor, so that the dark voltage can be changed for individual sensor.

$$\text{irradiance } (\mu W/m^2) = \frac{\text{output voltage} - \text{dark voltage}}{0.064}$$

- MLX75305

  Raw output value of MLX75305 is an analog voltage which is inverse proportional to the irradiance. The output voltage can be calculated though the equation below. Dark voltage is the output voltage at dark condition, and it is an unique parameter of each sensor, so that the dark voltage can be changed for individual sensor.

$$\text{irradiance } (\mu W/m^2) = \frac{\text{output voltage} - \text{dark voltage}}{0.007}$$

- ML8511

  Raw output value of ML8511 is an analog voltage which is proportional to the irradiance. The output voltage can be calculated though the equation below. Dark voltage, offset voltage, and UV error are unique parameters of each sensor, so that these values can be changed for individual sensor.

  Dark voltage is the output voltage at dark condition, offset voltage is difference voltage between output voltage at 10 mW/cm^2 and dark voltage, and UV error is the error between real UV index and calculated UV index.

$$\text{UV index } = (\text{output voltage} - \text{dark voltage}) \times \frac{14.9916}{\text{offset voltage}} - \text{error term}$$
$$\text{error term} = \frac{14.9916}{\text{offset voltage}} - \text{UV error}$$

○ **SPV1840LR5H-B** Raw output value of SPV1840LR5H-B is an analog voltage which is proportional to the sound level. the Raw output need to be converted to sound level in decibel (dB).

5.2.3 Chemsense

- **Chemical sensors – Total reducing gases, Nitrogen dioxide, Ozone, Hydrogen sulphide, Total oxidizing gases, Carbon monoxide, and Sulfur dioxide:** AFE ADC values need to be conversed into ppm.

- **SHT25, LPS25H:** Given values of SHT25 and LPS25H are 100ths of temperature in Celsius and 100ths of relative humidity value. If barometric pressure need to be converted in hPa, refer that hPa is 100 times of Pa.

$$\text{temperature } (^\circ C) = \frac{\text{output value}}{100}$$

$$\text{relative humidity } (\%RH) = \frac{\text{output value}}{100}$$

$$\text{barometric pressure } (hPa) = \frac{\text{output value}}{100}$$

- **Si1145:** Si1145 is a light sensor. Raw values coming from the sensor are three fixed hex integers, however because Chemsense board driver is not completed the values are needed to be ignored.

- **ADC Temperatures – CO ADC Temp, IAQ/IRR ADC Temp, O3/NO2 ADC Temp, SO2/H2S ADC Temp, and CO CMT Temp:** Chemsense board measures temperature of sensor ADCs. All of them give ADC temperature in 100ths of degree Celsius.

$$\text{temperature } (^\circ C) = \frac{\text{output value}}{100}$$

- **Accelerometer, Gyro:** Raw reading of the sensor values need to be conversed into appropreate value.

5.2.4 Alpha Sensor:

- **Histogram, Firmware, and Configuration** Raw reading of the sensor values need to be conversed into appropreate value.

A Appendices

- **Firmware Version:** Hardware and software version are composed with 5 different numbers. Hardwares for v2 version are v3.1, and software version is differed by the physical components. We have two different software versions according to the existance of tha Chemsense board in the node. The version information needs to be listed as shown below.

Major HW version . Minor HW version   Major SW version . Minor and Sub SW version

  - 3.1 3.01: Chemsense board is not physically included in the node
  - 3.1 3.02: Chemsense board is physically included in the node