RIT SPEX Hyperion Protocol

Dylan R. Wagner, Alec Herbert, Chris Johnson

February 1, 2018

1 Abstract

The Hyperion Protocol defines data packing techniques and configurations in order to achieve communication between endpoints. A universal header will be defined which allows senders the freedom to include a wide range of information from individual sensors or information groups. In addition, multiple other data configurations will be defined. These additional data configurations will store information regarding data groups from sensors without any notion of time captured.

2 Header

The header will be required for any outgoing and incoming data within the realm of the Hyperion protocol. This header will include information regarding the which data group (Sensor values) is contained in a message, flags and time sent. Any data wished to be sent with the Hyperion Protocol will be tacked on behind the header. The Hyperion Protocol header will be **40bits** long or **5 bytes**.

2.1 Header Data

Data contained in the header will have a maximum size, this size will be defined in bits and largest value stored.

Data Group (4 bits) The Data Group section will contain information to distinguish which data is contained after the header. A size of 4 bits will be allocated to this section, this allows for 16 separate values. This will be plenty for the number of data groups sent within the Hyperion system.

Flags (4 bits) The Flags section will be used to define how the data will be parsed or handled. A size of 4 bits will allow for 4 flags to be set at a time.

Sent Time (32 bits) The Sent Time section will be used to determine the time sent in seconds from the Hyperion system. This will be used for timestamps/data evolution analysis. This section will need to be 32 bits. This allows for a value of maximum value of 4294967295. A large maximum value is needed to store the time in seconds from unix epoch (Jan 01 1970. (UTC)). Currently at the time of writing this, this value is in the 1.5 billion range.

2.2 Visualization

Table 1: Hyperion Header

1	2	Sent Time (32 bits)
Data		

1 Data Group (4 bits)
2 Flags (4 bits)

2.3 Header Values

This section displays mappings between header values and their meanings. In the header only two sections will need further definitions. These sections are the Data Group and Flags sections.

2.3.1 Data Group Mapping

Table 2: Data Group Mapping

Value	Mapping
0	No Data
1	LSM9DS1
2	BME280
3	CCS811
4	PerfectFlite Strato Logger
5	...
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

2.3.2 Flag Mapping

Table 3: Flag Meanings

Bit Pos	Flag Meaning
0	...
1	
2	
3	

3 Data Groups

3.1 LSM9DS1

This section will define the data frame for the LSM9DS1 data group. The LSM9DS1 data group contains information regarding acceleration, gyroscope and magnetometer sensors in the X, Y and Z axis. In total, this data frame will be 36 bytes in size.

3.1.1 LSM9DS1 Data

Acceleration X, Y, Z (12 bytes - 96 bits) The acceleration section of the LSM9DS1 data frame will store the X, Y, Z values. Each axis is 4 bytes or 32 bits long. The X, Y, Z values will be placed in that order.

Gyroscope X, Y, Z (12 bytes - 96 bits) The gyroscope section of the LSM9DS1 data frame will store the X, Y, Z values. Each axis is 4 bytes or 32 bits long. The X, Y, Z values will be placed in that order.

Magnetometer X, Y, Z (12 bytes - 96 bits) The magnetometer section of the LSM9DS1 data frame will store the X, Y, Z values. Each axis is 4 bytes or 32 bits long. The X, Y, Z values will be placed in that order.

3.1.2 Visualization

Table 4: LSM9DS1 Data frame

ax (4 bytes)	ay (4 bytes)	az (4 bytes)
gx (4 bytes)	gy (4 bytes)	gz (4 bytes)
mx (4 bytes)	my (4 bytes)	mz (4 bytes)

3.2 BME280

This section will define the data frame for the BME280 data group. The BME280 data group contains information regarding atmospheric temperature, pressure and humidity. In total, this data frame will be 16 bytes in size.

3.2.1 BME280 Data

Temperature (4 bytes) The temperature section of the BME280 data frame will store the atmospheric temperature in degrees Celsius. In total, the temperature section will be 4 bytes.

Pressure (4 bytes) The pressure section of the BME280 data frame will store the atmospheric pressure in kPa. In total, the pressure section will be 4 bytes.

Humidity (4 bytes) The humidity section of the BME280 data frame will store the atmospheric humidity by percentage. In total, the humidity section will be 4 bytes.

BME_Altitude (4 bytes) The Altitude section of the BME280 data frame will store the altitude derived by pressure. In total, the BME_Altitude section will be 4 bytes.

3.2.2 Visualization

Table 5: BME280 Data Group

Temperature (4 bytes)	Pressure (4 bytes)
Humidity (4 bytes)	BME_Altitude (4 bytes)

3.3 CCS811

This section will define the data frame for the the CCS811 data group. The CCS811 data group contains information regarding atmospheric composition. In, total, this data frame will be 4 bytes in size.

3.3.1 CCS811 Data

Total Volatile Organic Compounds (2 Bytes) The Total Volatile Organic Compounds section of the CCS811 data frame will store the total volatile organic compounds in ppm. In total, this section will be 2 bytes.

Equivalent Carbon Dioxide (2 Bytes) The Equivalent Carbon Dioxide section of the CCS811 data frame will store the equivalent carbon dioxide in ppm. In total, this section will be 2 bytes in size.

3.3.2 Visualization

Table 6: CCS811 Data Frame

VOC (2 bytes)	ECD (2 bytes)

3.4 PerfectFlite Strato Logger

This section will define the data frame for the Strato Logger data group. The Strato Logger data group provides accurate information regarding altitude. In total, this data group will be 2 bytes in size.

3.4.1 Strato Logger Data

Altitude (2 bytes) The Altitude section of the Strato Logger data group will store the altitude in meters. In total, this data section will be 2 bytes in size.

3.4.2 Visualization

Table 7: PerfectFlite Strato Logger Data Group

Altitude (2 bytes)