See page 71-72 for old code that this replaces.

In the previous code, there is a delay of ~1s for reading each sensor. We would request the data, wait one second, and then read the data for each sensor. But the I2C Bus is free to use during the delay time...

Idea: ~~Read~~ Request data from all sensors back to back. Wait ~1 Second. Read all sensors response back to back.

```
18 void requestAtlas(int sensorAddress)
19 {
20   Wire.beginTransmission(sensorAddress);
21   Wire.write(82);                  // (R)ead
22   Wire.endTransmission();
23 } // end of request()
24
25 float readResponseAtlas(int sensorAddress)
26 {
27   Wire.requestFrom(sensorAddress, 20, 1);
28   code = Wire.read();
29
30   // 1 is success
31   // 2 syntax error
32   // 254 not ready
33   // 255 no data to send
34
35   // Read the data from the sensor
36   for (int ii = 0; ii < 20; ii++)
37   {
38     charIn = Wire.read();
39     sensorData[ii] = charIn;
40   }
41   Wire.endTransmission();
42   // Parse the data
43   data = atof(sensorData);
44
45   // Reset Variables
46   charIn = 0;
47   for (int jj = 0; jj < 20; jj++)
48   {
49     sensorData[jj] = 0;
50   }
51   return data;
52 } //end of readResponseAtlas()
```

```
73 void loop()
74 {
75   requestAtlas(DOaddress);
76   requestAtlas(ORPaddress);
77   requestAtlas(pHaddress);
78   requestAtlas(Caddress);
79
80   delay(timeDelay);
81
82   float DO  = readResponseAtlas(DOaddress);
83   float ORP = readResponseAtlas(ORPaddress);
84   float pH  = readResponseAtlas(pHaddress);
85   float C   = readResponseAtlas(Caddress);
86
87   Serial.print(DO);  Serial.print(" ");
88   Serial.print(ORP); Serial.print(" ");
89   Serial.print(pH);  Serial.print(" ");
90   Serial.print(C);   Serial.println();
91 }
```

only one delay!

~1s

```
10:39:39.687 -> 6.93 821.50 7.16 0.00
10:39:40.600 -> 6.93 821.60 7.16 0.00
10:39:41.492 -> 6.93 821.50 7.16 0.00
10:39:42.425 -> 6.93 821.60 7.16 0.00
10:39:43.327 -> 6.93 821.80 7.16 0.00
```

This is very useful for the autonomous version. We can gather data more often.