Buoy Memory Management

# Introduction

This document details the flow of data between 4 SPI Flash Chips and the stm32l476RG Microcontroller. Flash storage is a fundamental component for high volume data. These chips will provide permanent storage to IMU Data (+- 2 – 10KB per set), GPS data (+- 25 bytes), Environmental Data (+- 8 bytes) and Power information. However, in order to optimize for performance and robustness, some form of data organization must be implemented for intelligent retrieval/ storage of data in a meaningful way. In addition, the system requires some form of back up should the device be unable to connect to the flash chips. The following document outlines the planning and implementation for such a system

# Flash Chips Physical Description

The flash Chips being used are AT45DB641E SPI Serial Flash Chips. Each chip can hold up to 64Mbit of data. Data can be read/ written at speeds of up to 85MHz of 15MHz in low power mode. The device is low power with high data retention requiring a supply voltage of 1.7V – 3.6V and draws a maximum of 11mA in Active Read mode thereby making it one of the lowest power consumption components in the system. In addition, the device comes with 2 x 256byte buffers that can store data while a read/ write operation is taking place. Memory is Organized into sectors (2 – 256 Kbs long), blocks (2kB long) and pages (256 bytes) with write, read and erase options at each level. The following diagrams are taken from the data sheet and show the architecture and memory organization of the chips

A screenshot of a social media post

Description automatically generated

Figure : Block diagram showing the architecture of the chip

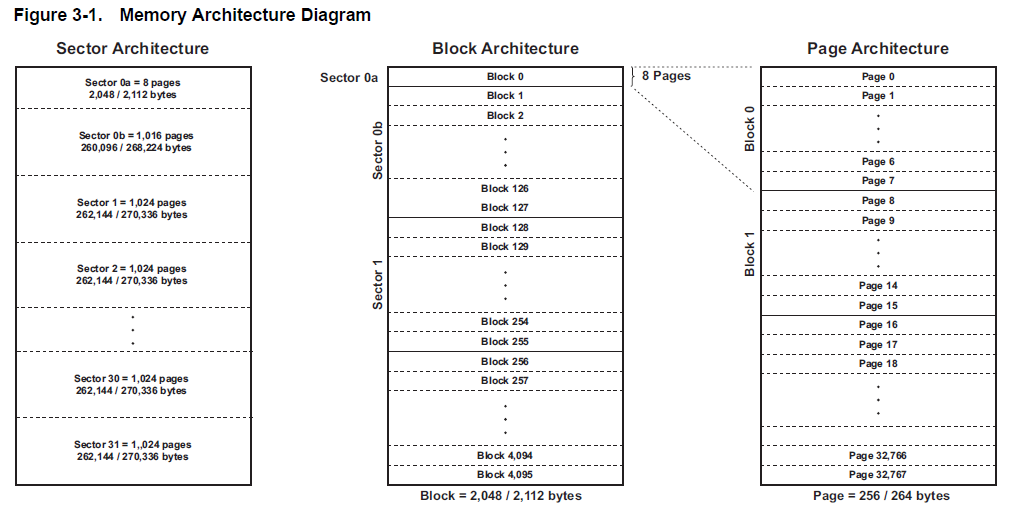


Figure : Memory Layout of Flash Chip Showing pages, sectors and blocks as well as sizes in bytes

# Research Question

In order to implement a robust solution, the following research questions need to be answered.

1. How can I organize data so that it is convenient to store and retrieve?
2. How can I organize the chips, so data is stored in an optimized sequential location?
3. What can I do if I lose contact with one or more chips?
4. How can I recover data/ reinitialize the system?
5. Do I need a backup storage location?

# Determining data sizes

## GPS

GPS data is sampled over a 2-hour interval. During this interval, the sample rate can vary between 1 sample every 15 mins and 1 sample every half an hour. During each sample period, the device takes 5 readings spaced 1 second apart of the following data:

1. Time and Date Information
2. Global Latitude
3. Global Longitude
4. Dilation of Precision (Vertical, Horizontal, Positional)
5. Number of Satellites
6. Fix Type

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Variable Type** | **Size (bytes)** |
| **Epoch Time** | uint32\_t | 4 |
| **latitude** | float32\_t | 4 |
| **longitude** | float32\_t | 4 |
| **HDOP** | uint8\_t [2] | 2 |
| **PDOP** | uint8\_t [2] | 2 |
| **VDOP** | uint8\_t [2] | 2 |
| **Number of Satellites** | uint8\_t | 1 |
| **Fix Type** | uint8\_t | 1 |
|  | **Total** | 20 |

Therefore, in a 2-hour sample window the total number of data collected is shown in the table below for a single sample session.

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 8 | 160 |
| 1/30 mins | 4 | 80 |

Should we wish to store 5 samples per sample point the total data per sample would be

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 40 | 800 |
| 1/30 mins | 20 | 400 |

## Environmental Data

|  |  |  |
| --- | --- | --- |
| **Variable** | **Size** | **Number of bytes** |
| **Temp** | int32\_t | 4 |
| **Pressure** | uint32\_t | 4 |

Data is collected from the BMP280 environmental sensor from 2 variables: Temperature and Pressure. Both compensated and uncompensated Temperature values are stored as long its (int32\_t) with the final temperature value being a signed integer value. Pressure is stored as a long unsigned int (uint32\_t)

If environmental Samples are synchronized with GPS samples, then the amount of data we can expect from a single session is as follows:

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 8 | 64 |
| 1/30 mins | 4 | 32 |

If we were to increase the number of samples per sample point to 5 then the amount of data, we can expect becomes

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 40 | 320 |
| 1/30 mins | 20 | 160 |

## Current and Voltage Sensor

The INA219 Power sensor outputs the following information

1. Shunt Voltage
2. Bus Voltage
3. Current
4. Power

The variable breakdown is shown in the table below

|  |  |  |
| --- | --- | --- |
| **Variable** | **Size** | **Number of bytes** |
| **Bus Voltage** | int16\_t | 2 |
| **Shunt Voltage** | int16\_t | 2 |
| **Current** | int16\_t | 2 |
| **Power** | int16\_t | 2 |

Therefore, the following memory allocation is required for INA219 Data

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 8 | 64 |
| 1/30 mins | 4 | 32 |

Once again, should we require 5 samples per sample point, our data requirement is as follows

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 40 | 320 |
| 1/30 mins | 20 | 160 |

## Total Drift Data Breakdown

|  |  |
| --- | --- |
| **GPS** | 20 |
| **Environmental** | 8 |
| **Power** | 8 |
| **Total:** | 36 |

Hence for a single sample point, we can accumulate 36 bytes of data. For a 5 – point sample this total increase to 180 bytes per sample

Over a 2-hour sample period, we can accumulate the following amount data for a single sample point taken at a fixed rate.

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 8 | 288 |
| 1/30 mins | 4 | 144 |

By accumulating 5 samples every sample point, this total increases to

|  |  |  |
| --- | --- | --- |
| **Sample Rate** | **Number of Samples** | **Total Bytes** |
| 1/15 mins | 40 | 1440 |
| 1/30 mins | 20 | 720 |

Therefore, in our data range, we can expect 288 – 1440 Data samples for a sample rate of one sample every 15 mins, and 144 – 720 for a sample rate of one sample every half an hour. Data in memory will be grouped by pages. Pages can be configured as 256 or 264 bytes long. This will influence how data is stored and how many pages need to be allocated. The following table presents a breakdown of the number of pages required for each sample interval given the different configurations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Number of Samples** | **Samples per point** | **Sample Rate** | **Total Bytes** | **Number of pages (256 byte)** | **Number of pages (264)** |
| 4 | 1 | 1/30 mins | 144 | 1 | 1 |
| 8 | 1 | 1/15 mins | 288 | 2 | 2 |
| 20 | 5 | 1/30 mins | 720 | 3 | 3 |
| 40 | 5 | 1/15 mins | 1440 | 6 | 6 |

As shown in the table above, the number of pages required to fit all the data is the same for both 264- and 256-page sizes. When the page size is 256, at the highest possible case, data will take up 6 pages i.e. 1.536 kB this results in 96 bytes of data left over. When the page size is 264 bytes long, data will be allocated 1.584 kB of space resulting in 144 bytes of unallocated data. This is wasted space, even though there are additional variables that will require storage, not nearly as much memory is required to store these variables.

Therefore, it is advisable to configure the Flash chips for a page size of 256 bytes and allocated the first 6 pages for GPS, ENV and POWER sensor data storage and the first 96 bytes for system/ flash variables. This will account for 0.0192%. At this rate, a single flash chip can store 12 years’ worth of data at the lowest sample rate and 1 years’ worth of data at the highest.

# IMU Data

IMU data is the most critical data set for this project, the number of samples expected is dependent on the following variables:

1. Sample Window Period
2. Sample Rate
3. Number of axes to be sampled

The MPU6050 is a 6 doF IMU (3 axes Accelerometer and a 3 axes Gyroscope) with additional temperature measurement outputs and expansion ports for a magnetic compass. In its current state, we disregard the additional expansion ports and temperature reading and focus only on accelerometer and gyroscope. Each axis is stored as a signed 16-bit integer with the MSB and LSB stored in separate 8-bit registers. Assuming calibrated and compensated data, each integer value is mapped to a rotation/ acceleration based on the Full-Scale Range, Filter Settings and ADC resolution.

From our user requirements the sample period for collecting wave data is a minimum of 15 mins. Average ocean wave sample periods are recorded at 20 mins sometimes even as high as 30 mins for significant wave height. Using Nyquist sample theory, the dominant wave frequency occurs at about 1 Hz. Sampling at 2 Hz is a bare minimum however, 5Hz is recommended. 10 Hz is ideal. It may also be that we require the imu to detect and sample collisions. The requirements for this data are

## Wave Data

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Number of Axes:** | | | **3** |  | |  | |  | | |
| **Sample Frequency (Hz)** | **Sample Period (mins)** | | | | | | | | | |
|  | **15** | **20** | | | **25** | | **30** | |
| **2** | 10800 | 14400 | | | 18000 | | 21600 | |
| **5** | 27000 | 36000 | | | 45000 | | 54000 | |
| **10** | 54000 | 72000 | | | 90000 | | 108000 | |
| **20** | 108000 | 144000 | | | 180000 | | 216000 | |
| **50** | 270000 | 360000 | | | 450000 | | 540000 | |

Table : Number of bytes generated from a range of sample periods and frequencies for sampling from 3 axes

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Number of Axes:** | | | **6** |  |  |  |
| **Sample Frequency (Hz)** | **Sample Period (mins)** | | | | | | |
|  | **15** | **20** | | | **25** | **30** |
| **2** | 21600 | 28800 | | | 36000 | 43200 |
| **5** | 54000 | 72000 | | | 90000 | 108000 |
| **10** | 108000 | 144000 | | | 180000 | 216000 |
| **20** | 216000 | 288000 | | | 360000 | 432000 |
| **50** | 540000 | 720000 | | | 900000 | 1080000 |

Table : Number of bytes generated from a range of sample periods and frequencies for sampling from 6 axes of the IMU

From Tables 1 and 2 above, we can see that IMU sampling will account for the bulk of data storage. It should be noted that for 6 axes sampling, any frequency above 50 Hz and a sample period greater than 30 mins is bound to overrun the data storage of a single chip and will require additional, more complex memory management schemes. In addition, another consideration is whether the data is to be logged or transmission. Transmission via Iridium is unadvisable for a multitude of reasons:

1. The extremely narrow Transmission bandwidth means that multiple transmissions will be required
2. Lack of reliability means that we cannot be certain all data packets will be transmitted
3. Long transmission times means that the buoy could take up to 20 mins to transmit all data. Spacing the packets out implies that the data could take up to a month to be fully transmitted
4. High Power Consumption means that multiple transmission attempts will deplete the power source much faster

In order to fully characterize the data set, it is important that we sample both Accelerometer and Gyroscope Data therefore, the full 6 axes need to be sampled. By choosing the sample frequency to be 5Hz and the sample period to be 20 mins, we select a range that allows for enough samples to be collected to determine the dominant wave height as well as allowing us to sample well above the Nyquist frequency of the waves.

Therefore, our data requirement for the IMU is 72000 bytes per sample. IF we sample the IMU twice a day and assume no Drift data has been collected, we can expect the Flash chip to hold 55 days’ worth of data. If we allocated 3.5 Flash chips to the IMU set, then we can record 194.5 days’ worth of data

# Translating Data requirements into Flash Chip Requirements

As shown in the above sections, the storage from a single flash chip should be sufficient to record data for transmission over the expected lifespan of the buoy, having 2 or more chips will not only increase the storage capacity of the chip but allow for advanced memory management techniques such as circular buffers, redundant data storage and backups. The flash chips are a critical component and as such, we cannot deploy/run the system without them. However, there exists a possibility that one or more flash chip will fail to initialize or become corrupted. This section also outlines the procedure for dealing with such failures.

## Chip Priority System

In the current Version of the Buoy, there is an array of 4 flash chips resulting in a total memory storage of 32Kb. The chips will be labelled in order of appearance from left to right on the board.

When the Chip initialization procedure begins, the First chip from the left will be designated as the start chip. This chip will be partitioned with the first 6 pages reserved for Drift data and the remaining data for Environmental data. Each chip will have a byte at the beginning of memory allocated towards the status of the byte. If the first chip is unavailable, the next available sequential chip is allocated as the start and the chip numbering system is adjusted accordingly.

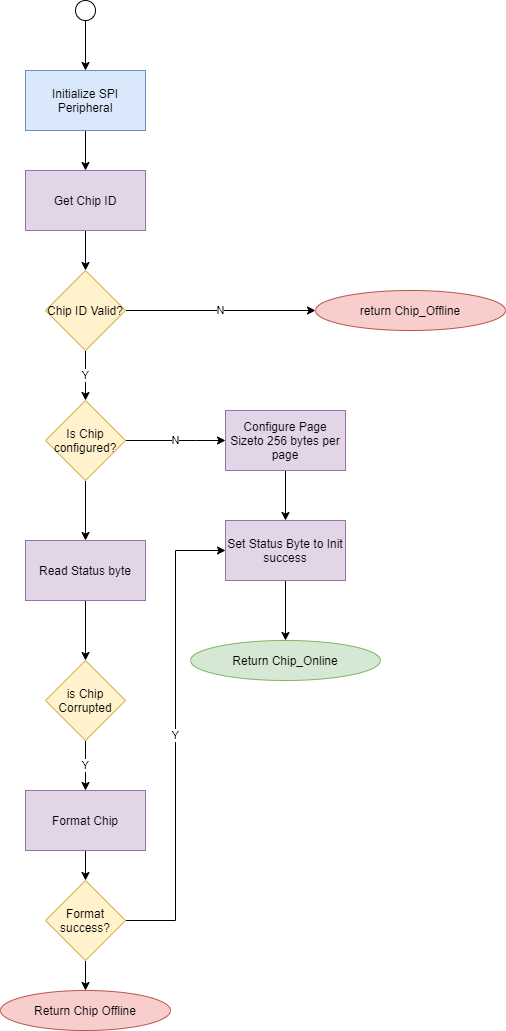
If the data of a chip is overrun, the data will be stored in the immediate next chip. If this chip is unavailable, the next sequential blank chip is prioritized. If none of the chips are blank, the system will prioritize the closest, non-back up chip. If this is unavailable, the system will use the embedded RAM for storage. The onboard storage is extremely limited and storing data in flash invites the possibility of collisions with code storage. Therefore, this is only to be used in extreme circumstances.

Therefore, the chip selection priority will be as follows

The following identifiers are programmed to the chip during Initialisation:

1. Active: Chip is online and designated to store data
2. Next Active: Chip is online, Available and will be used when the Active Chip is full
3. Inactive: Chip is online, available but not used
4. Full: Chip has reached capacity and can no longer be used
5. BackUp: Chip is used as a back up storage location

# Chip Peripheral Initialization Routine



Once each chip has been initialized, the device will check to see what the current status of each chip is. If this is a first time initialization process/recovery from corruption and will assign a chip a priority. Each priority will dictate the order in which