Development of 3D Printed Miniature GPS Drifters for River Flood Studies

Thomas Paul William Quinn; B224307
Electronic and Electrical Engineering

Abstract: Real time hydrological models are used to better understand and predict flooding, without measured data from the field validation of these models is not possible. With the use of additive manufacturing a small, robust, easily rechargeable drifter was designed and manufactured that acquired GPS positional data while floating downstream. A LabVIEW software platform was also developed to allow easy wireless retrieval of the locally saved data. Due to the size of the drifter and its low position when floating, experiments were conducted to investigate how a GPS modules height above water effected the received signal quality. Results revealed a small improvement in the horizontal dilution of precision and number of satellites used but no significant degradation could be seen, however the project is not suitable for large scale deployment in its current state due to a lack of live tracking capabilities and long period of enclosure manufacture time. With a software architecture able to scale and uniquely identify multiple devices, with the resolution of the limitations discussed, the project would be suitable for large scale, long term real-world deployment. The projects software and digital files can be found on github (see https://github.com/thomasquinn93/gps-drifters).

1 Introduction

Effective flood prediction depends on the availability and analysis of various different large data sets, such as precipitation history and storm surges, fed into prediction models to provide accurate results and flooding forecasts [1]. The limited availability of data is in part related to the high expense and cumbersome nature of Metrological parameter stations and environmental monitoring systems, leading to a lack of large scale monitoring due to cost constraints and inconvenience [1]. The acquisition of this data is still however the prerequisite for hydrological model validation and accurate flood predictions.

Surface current is another important metric in monitoring surface water speeds and sediment transportation aiding predictions before, during and after flooding has occurred [2]. Flow measured over time from a static point (Eulerian measurement) is typically logged to a static land-based Data Acquisition System (DAQ) whereas the position of surface flow as it travels downstream (Lagrangian measurement) is measured via floating sensors that save the data locally [3]. These floating sensors are commonly known as “drifters”.

Since the 1950s aquatic drifters have been developed for oceanographic research finding success in oceans, where fixed infrastructures are generally uncommon [4]. River environments present extensive obstacles both natural and man-made such as vegetation and bridges, and are some of the reasons drifters have yet to be widely adopted in these environments [5]. Usually accompanied by personnel in boats retrieving the drifters when they get caught in obstacles, propulsion and autonomous obstacle avoidance is typically omitted to reduce cost and size [5].

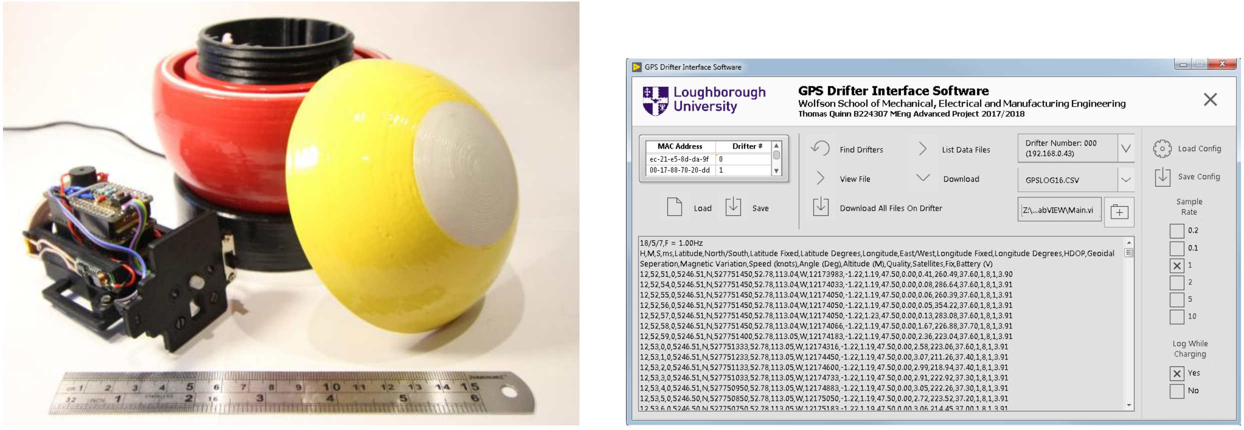
With the recent popularity of small fused deposition modelling (FDM) 3D printers, it is now possible to manufacture small, bespoke parts fast and cheaply to match a users design and engineering requirements. The ability to utilise agile product development reduces the production cost of small batch designs while allowing incremental changes, but has yet to gain popularity in the production of many real world products.

The idea of combining the development of a small river drifter with additive manufacturing techniques lead to the conceptualisation of the work conducted and discussed in this paper. The aim of the project was to develop a standalone, easily rechargeable, 3D printed system that acquired Global Positioning System (GPS) data to track surface flow, while remaining transportable and robust to casual handling. Once data collection had concluded a method of easily and uniquely identifying the drifter was developed, enabling the retrieval of recorded data wirelessly. An image of the system and accompanying software meeting these objectives can be seen in Figure 1.

The next Section 2 provides a description of both electrical/physical hardware and software used in the construction of the drifter and accompanying charging stand. Section 3 details experimental results from system testing and how findings fed into design choices. The paper concludes with a discussion on the feasibility of large scale deployment and directions for future research [6].

2 System Description

First the hardware electronics and enclosure design will be discussed followed by the custom embedded and computer software. Justification for design choices resulting from testing will be examined briefly, being further expanded on in Section 3.



(a) GPS drifter enclosure, electronics and charging stand. (b) Software interface for downloading recorded data.

Figure 1: Photo of drifter and screen shot of interface software.

2.1 Hardware (Electronics)

The drifter design consisted of a microcontroller, a GPS module, a wireless communications module and data logging module powered via a wirelessly charged small battery. A block diagram of the systems architecture can be seen in Figure 2. With an emphasis on price and ease of construction the project was designed and constructed from commercial off the shelf components (COTS) enabling the construction of multiple units if needed. A full system schematic and Bill of Materials (BOM) can be found in [Appendix 1, 2].

2.1.1 Microcontroller & Power Management

With a plethora of microcontrollers and microprocessors readily available a subset of brands were considered when deciding on an appropriate platform for the project. Considering both lesser known controllers such as the PICAXE 18M2 Peripheral Interface Controller (PIC) chip for size, low power consumption and low cost, in addition to well known single-board computers such as the Raspberry Pi for processing power and high-level language support, the Adafruit Feather M0 “Adalogger” microcontroller was ultimately chosen as the brain of the drifter.

The Adalogger contained a 32-bit ATSAMD21G18 ARM Cortex M0 processor, clocked at 48 MHz using 3.3 V logic and containing 256 K of FLASH and 32 K of RAM, and was part of a new Arduino compatible development platform from Adafruit known as the “Feather” range [7]. The microcontroller main-boards and daughter-boards (FeatherWings) are a cross-compatible platform allowing users to mix-and-match microcontrollers, wireless protocols and various components through vertically stacking circuit boards. With a Micro Secure Digital (MicroSD) card and micro Universal Serial Bus (USB) port available, the board eliminated the need for a Future Technology Device International (FTDI) RS232-USB converter chip/cable by having USB-to-Serial programming built in, allowing easy debugging and integration with standard Arduino sudo-C++ libraries.

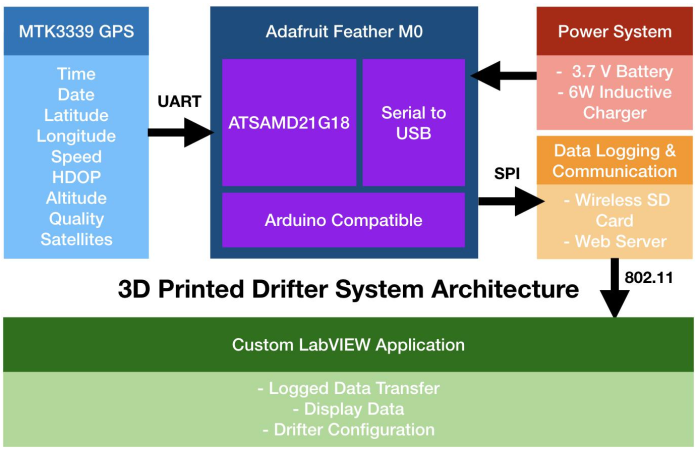


Figure 2: High-level overview of drifter system architecture. All components COTS.

(a) Toshiba FlashAir Wireless SD Card. (b) MicroSD to SD Card adapter [8].

Figure 3: Wireless SD card and adapter used.

An integrated 3.3 V, 500 mA peak regulator, 100 mA MCP73831/2 battery charger and JST connector for use with 3.7 V lithium polymer (lipo) batteries were also present on the Adalogger eliminating the need for a custom Battery Management System (BMS), although one using lithium-ion 18650s was considered. Further discussion on the systems power requirements can be found in Section 3.4.

2.1.2 Data Logging & Wireless Module

Combining both a wireless communication and data logging module into one, a 4th Series Toshiba FlashAir Wireless Secure Digital High Capacity (SDHC) card was used as the data storage medium for the drifter. The FlashAir contained an embedded linux based micro computer and wireless Local Area Network (LAN) functionality[1], allowing reading and writing of data remotely to a computer on the same wireless LAN via a built-in web server [9]. Communication with the embedded server used HyperText Transfer Protocol (HTTP) communication and is discussed in Section 2.4.

Although not necessary due to the traditionally low sample rates of aquatic drifters, the FlashAir was rated by the manufacturer to have Ultra High Speed (UHC) Class3 sequential read/write speeds of 90 MB/s and 70 MB/s respectively and a wireless transfer rate of approximately 31.4 Mbps [9].

Due to the Adaloggers built in MicroSD port, an adapter was needed to use the standard size FlashAir SD card with the microcontroller and communicate with it over the boards Serial Peripheral Interface bus (SPI). Figure 3 shows both the FlashAir and the MicroSD-to-SD adapter.

2.1.3 GPS Module

The module chosen for the project was an Adafruit Ultimate GPS FeatherWing built around the MTK3339 chipset due to its ability to track 22 satellites on 66 channels, having a manufacturer stated -165dBm tracking sensitivity, 1.8 m accuracy with the built in ceramic patch antenna, up to 10Hz update rate, low voltage requirement of 3-5.5 V and a low current draw of 20 mA during navigation [10]. Additionally the module supported Differential Global Positioning System (DGPS) and Assisted Global Positioning System (AGPS) with a u.FL connector present on the board for use with an external active antenna if required. Being part of the Adafruit Feather range, the board was stacked directly above the Adalogger microcontroller eliminating the need for any additional interface circuitry.

By using a network of at least 24 active satellites[2] orbiting over 20,000 km above Earth, GPS modules calculate their position from radio frequencies (between 1.1 and 1.5 GHz) transmitted from the satellites [11]. Known as trilateration, timestamps and orbital position data received from a minimum of 4 satellites[3] can be used by a module to accurately determine the distance between itself and each satellite in view, combining this information to calculate its position on the Earths surface [12].

Disregarding proprietary formats, most GPS modules output National Marine Electronics Association (NMEA) formatted data over a Universal Asynchronous Receiver-Transmitter (UART) serial interface. The lines of data are comma delimited and formatted into a possible 58 interpreted standard strings known as sentences [13].

The COTS modules considered for this project, including the Adafruit GPS FeatherWing, were limited to 2 NMEA sentences, Global Positioning Recommended Specific data (GPRMC) and Global Positioning System Fit data (GPGGA) and both contained Coordinated Universal Time (UTC), Latitude and Longitude in

[1]IEEE 802.11b/g/n (2.4GHz SISO, HT20/HT40) compliant.
[2]Maximum of 12 visible at any time.
[3]Position can be obtained from less than 4 satellites but the error in positional accuracy is larger than desired.

degree.decimal minutes. GPRMC contained information regarding signal validity, speed (knots), true course/heading (degrees), date stamp and magnetic variation and the GPGGA sentence contained a GPS quality metric, number of satellites in use, Horizontal Dilution of Precision (HDOP), altitude above sea level (meters) and geoidal separation (meters). An example of both sentences can be found in [Appendix 3].

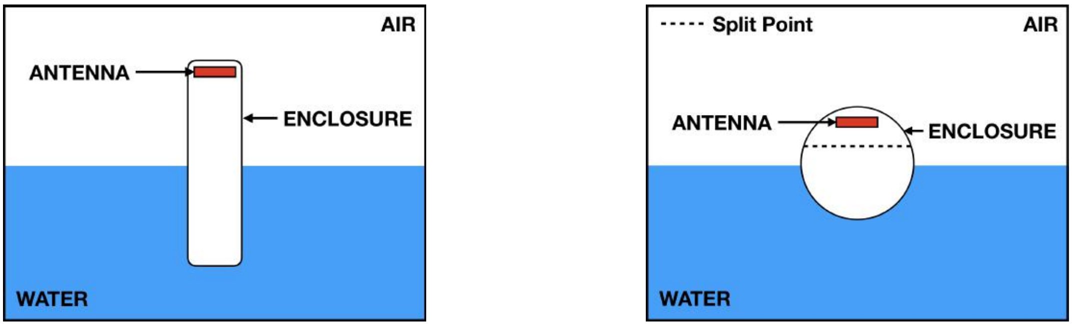
2.1.4 Additional Components

To enable the switching between different states in the embedded software (discussed in Section 2.3) a button was added to the system. A unipolar chopper stabilised hall effect sensor switch was also added to duplicate the functionality of the button while the enclosure was sealed. The sensor was triggered typically at 10 Gs of magnetic flux density via the south pole of a small magnet pressed against the enclosure. A 3 V piezoelectric buzzer and Red Green Blue (RGB) Light Emitting Diode (LED) were also added to provide audio/visual feedback while operating, the reasoning for which is discussed in Section 3.2.

2.2 Hardware (Electrical & Enclosure)

2.2.1 Enclosure Design

Typically drifters are cylindrically shaped, floating with 50% of the enclosures length above the surface of the water as illustrated in Figure 4a. These types of designs allow the GPS antenna to be located at a distance approximately half the length of the enclosure above the waters surface, providing an elevated location for signal reception. Through experimental testing (Section 3.3), the difference in general signal quality received by the MTK3339 GPS module at varying heights above the surface of water was negligible. Despite the GPS antenna being closer to the waterline (illustrated in Figure 4b) a spherical geometry design was chosen due to high water resistant properties discovered through testing detailed in Section 3.2.



(a) Example of cylindrical enclosure.

(b) Example of spherical enclosure.

Figure 4: Side elevation of different enclosure types in water.

An enclosure diameter of 124 mm was chosen to ensure an impact resistant 8 mm outer wall could be incorporated into the design, in addition to remaining compact enough to stay within the constraints of a typical FDM 3D printer build platform. Acrylonitrile butadiene styrene (ABS), a common 3D printing material, was chosen as the material for the enclosure due to its low cost, wide availability, insolubility in water and low print temperature compared to more exotic 3D print materials [14]. The design was modelled using 3D Computer Aided Design (CAD) software incorporating enclosed areas to aid buoyancy, a separate inner area for electronics, a “drip tray” to channel any water ingress and a channel for an o-ring seal, all of which are seen in Figure 5.

Given a spheres uniform shape the choice was made to split the design horizontally, allowing access to the interior, generating a top and bottom hemisphere. An appropriate height to split the enclosure was chosen by calculating the depth H at which it would float below the waters surface using Equations 1 and 2 [15]

$$V = \int_0^H \pi r^2(h)\, dh = \int_0^H \pi \left(1 - (1 - h)^2\right) dh = \left(H^2 - \frac{H^3}{3}\right) \tag{1}$$

$$\left(H^2 - \frac{H^3}{3}\right) = \frac{\left(\rho_e + \frac{4\pi r^2}{3}\right)}{\rho_w} \tag{2}$$

where V is the volume of the enclosure, r is the radius, ρ_e is the density of the enclosure and ρ_w is the density of water. By calculating a submersion depth of $\approx$ 43.2mm the split was placed 30 mm above this height to ensure the opening would not be level with the surrounding water.

The lower hemispheres base was flattened allowing the drifter to remain at rest on a flat surface. Additionally the shell thickness was reduced by $\approx 80\%$ to 1.5 mm to allow the wireless charging coils adequate separation for power transmission. The reduction in material enabled the placement of 2 magnets in holding cavities that when paired with a matching pair in the charging base aided alignment of the charging coils. These features are illustrated in Figure 5 and Table 1.

Eliminating external metal fixtures/fasteners and therefore potential sources of corrosion lead to the development of an alternative method of securing the hemispheres together and sealing the enclosure. An inner tube with a standard M76 thread, 4 mm pitch (M76x4) was modelled and printed to act as an inner core allowing the two hemispheres to screw together, additionally acting as a fixed point to mount the electronics and can be seen in Figure 6 and Table 2.

The upper part of the top hemisphere was printed with a reduced thickness in translucent white filament allowing light from the RGB LED to pass through. The colour change was not modelled but can be seen in Figure 1a. The LED was mounted to the "electronics holder", shown in Figure 6 in purple and to the shelf in Figure 7. Further images of the project can be seen in [Appendix 4] in more detail.

Designed using multi-part construction to accommodate limitations with additive manufacturing, the holder assembly slid into the core and housed all drifter electronics. M3 threaded inserts were heated and pressed into holes allowing pan head machine screws to join parts. Metal fixings were used internally as they would not be in contact with liquid.

2.2.2 3D Print Settings

FDM 3D printing is the process of making a physical object from a three-dimensional digital model by laying down thin layers of molten thermoplastic successively. All designs for this project were printed on a Flashforge Dreamer FDM 3D printer at settings listed in Table 4.

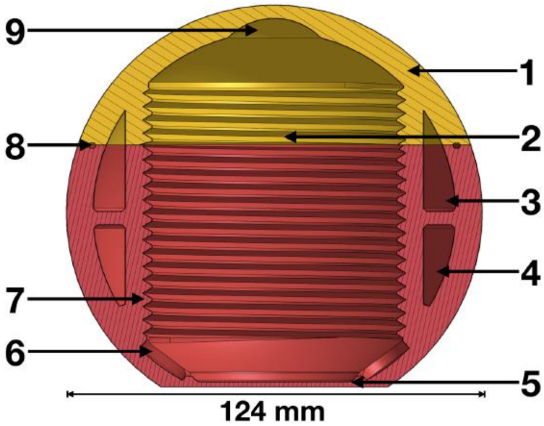


Figure 5: Spherical enclosure cross sectional model. Top in yellow, bottom in red.

Feature	#
Impact resistant outer wall	1
Inner void for electronics	2
Drip tray	3
Enclosed area for buoyancy	4
Thin base for charging	5
Space for magnets	6
M76x4 thread	7
O-ring channel	8
Thin top for LED	9

Table 1: Enclosure features.

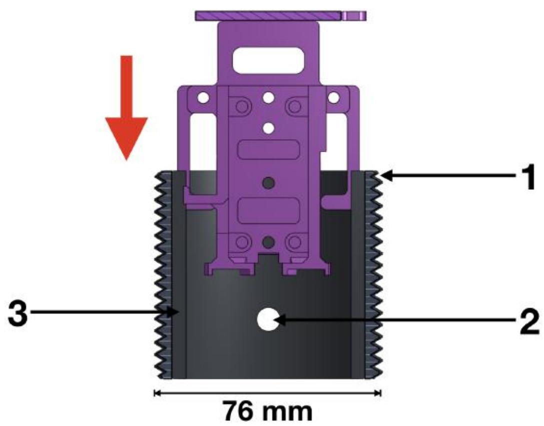


Figure 6: Inner core and electronics cradle cross sectional model. Core in black, cradle in purple.

Feature	#
Matching M76x4 thread	1
Holes to aid insertion into main body	2
Rails to guide electronics cradle	3

Table 2: Core features.

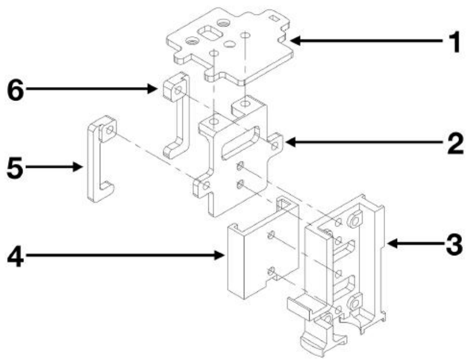


Figure 7: Electronics cradle exploded view.

Part	#
LED/Component Shelf	1
Wing Connector	2
Cradle	3
SD Card Holder	4
Left Wing	5
Right Wing	6

Table 3: Enclosure features.

Part	First layer speed (mm/s)	Retraction (mm)	Time		Estimated weight (g)	Material used (m)
			Estimated	Real		
Enclosure Top	5	1.3	6h 57m	7h 6m	69.69	27.86
Enclosure Bottom	5	1.3	16h 43m	17h 6m	172.14	68.81
Core	5	1.3	3h 20m	3h 40m	50.15	20.05
Component Shelf	15	2	22m	28m	3.83	1.53
Wing Connector	15	2	24m	30m	3.61	1.44
Cradle	15	2	54m	1h 10m	8.70	3.48
SD Card Holder	15	2	24m	29m	4.09	1.64
Left Wing	15	2	6m	6m	0.94	0.38
Right Wing	15	2	6m	6m	0.94	0.38
Charging Base	5	1.3	7h 16m	7h 35m	71.10	28.42
Total				38h 16m	385.19	153.99

Table 4: Component print settings and times. All printed with settings discussed in Section 3.1 and a base print speed of 30 mm/s.

Preliminary experiments were conducted to determine the optimum settings to produce an IP67 rated enclosure and were used for all prints [14]. The only significant parameters differing from the ones experimentally determined in Section 3.1 were the “base print speed”, “first layer print speed” and “retraction amount”. The base and first layer print speeds are speeds at which the extruder moves during extrusion and during the first layer respectively. A lower speed increased print time but yielded higher quality parts and reduced the risk of print failure. Retraction is the amount the extruder retracted the plastic at the end of each successive layer, optimal retraction resulted in high dimensional accuracy and surface finish. A reasonably high surface finish was needed on all parts as irregularities/deformities lead to water ingress.

The printer incorporated a heated chamber, allowing a constant temperature gradient across the object printing reducing premature/uneven cooling and warping, and was allowed to heat to an even temperature of 50°C 30 min prior to each print. A honeycomb infill lattice was needed instead of a traditional square pattern due to experiments demonstrating hot trapped air causing a “pillowing effect” on the top layer. Pillowing was reduced significantly when using hexagonal infill combined with more than 2 top solid layers, an example of which can be seen in Figure 8. The 1.5 mm base of the enclosure was printed without infill to reduce the amount of material separating the induction charging coils.

2.2.3 Charging Dock

To eliminate the need to open the drifter during charging, a 6 W maximum (5 V 1.2 A) inductive charging system was purchased and installed at the base of the electronics cradle. When pressed against the thin base of the lower hemisphere the spacial difference between the coils was small enough to allow the transfer of ≈ 645 mW, charging a 3.7 V 500 mAh lipo battery through the Adaloggers built in lipo charger. As the receiving coil was located on the inside of the enclosure the transmission coil was designed to be flush with the exterior of the enclosure base to minimise separation, sitting in a cut-out seen in Figure 9.

Commonly known as wireless charging, the module purchased conformed to the Qi open interface standard defining wireless power transfer using inductive charging [17]. The transmitter contained a coil of 20 turns as part of a series resonant circuit while the receivers coil was 16 turns and integrated into a parallel resonant circuit. A USB input was chosen as the voltage supply interface as most USB DC chargers are limited to 5 W. Figure 10 shows the transmitter coil embedded in the 3D printed charging base.

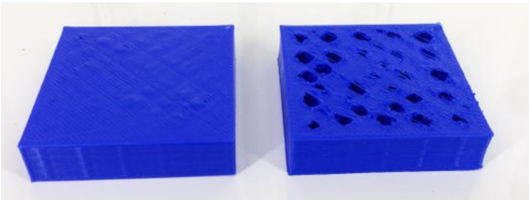


Figure 8: An example of 3D print pillowing. Left - Reduced pillowing, Right - Severe pillowing [16].

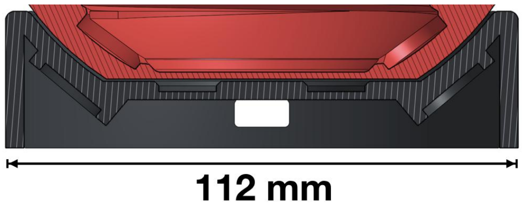


Figure 9: Inductive charging base model cross section. Base in black, enclosure base in red.

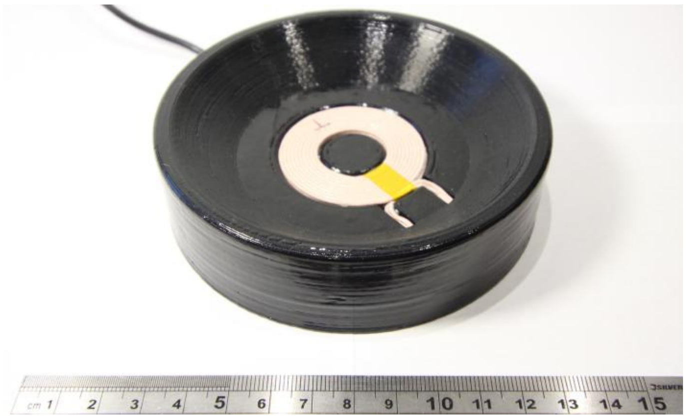


Figure 10: Transmitter coil embedded in 3D printed charging base.

2.3 Software (Embedded)

The embedded software for the drifter was written in the sudo-C++ Arduino programming language enabling the use of extensive support libraries for the hardware listed in Section 2.1. With three main modes: idle, live (sending GPS data to the serial port) and record (saving GPS data to the FlashAir SD card) a high-level overview of the software written is illustrated in Figure 11. While in record mode GPS data was saved as a .csv file to the FlashAir in the order: hour, minute, second, millisecond, latitude degree.decimal minutes, north/south indicator, latitude degree.decimals, longitude degree.decimal minutes, east/west indicator, longitude degree.decimals, HDOP, geoidal separation, magnetic variation, speed (knots), angle/heading (degrees), altitude (meters), signal quality, number of satellites being used, fix indicator and battery voltage. An example of the GPS data files can be found in [Appendix 5]. Git version control was used throughout the project and advanced functionality was developed to increase robustness, reliability and porting of code to

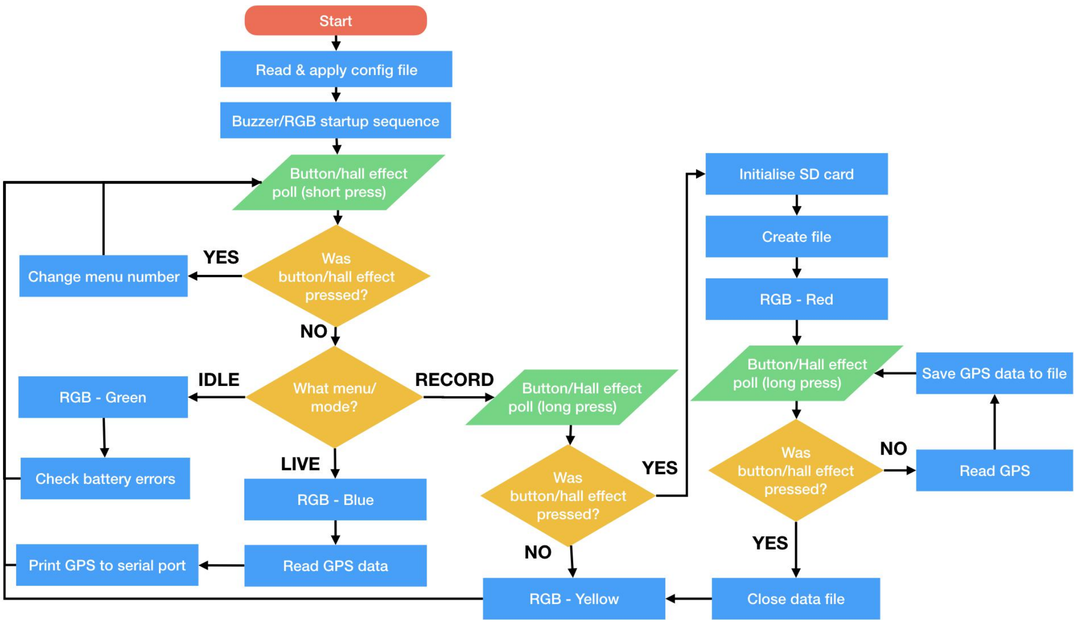


Figure 11: High-level process diagram of drifter embedded software.

future projects. These included:

- All code was written using Object Oriented (OO) design methods enabling multiple objects of each class to be instantiated if required.
- All classes were custom and written with an effort to limit dependancies between classes to reduce compile times and reduce software bugs.
- A change log was kept at the top of the “main” file to keep track of software revisions.
- If an SD card is present on startup a config file is read from the card, applying sample rate and enabling/disabling the ability to record data while charging.
- A simple startup routine that started the serial output port at 115200 baud, cycled through all RGB colours, played a C major arpeggio on the buzzer and started the GPS module allowed the user to observe the system was operating correctly.
- The RGB LED class contained functionality for both analogue and digital outputs enabling the fading between colours or simple on-off functionality for each internal diode.
- The RGB LED and buzzer class did not use any delays for fading or flashing methods to ensure time critical functions were executed when expected. Variables and the internal millisecond timer were used to calculate timings.
- The RGB LED, button and hall effect sensor classes were written to operate with both pin HIGH and pin LOW inputs/outputs, selectable through one variable.
- The button and hall effect classes were identical and capable of software debouncing the input signals, detecting a short press of less than 2 seconds and a long press of over 2 seconds without the need for delays. Variables and the internal millisecond timer were used to calculate timings ensuring time critical functions were executed when expected.
- When sat on the charging base the drifter used a battery class to detect the voltage on an analogue input pin. If charging was detected the RGB would continuously fade from on to off in the colour of the current menu selected. When removed from the base the RGB would stop fading and start continuously flashing with a duty cycle of 25% over 2 s to conserve battery life.
- Instead of writing the NMEA GPS sentences directly to the SD card the GPS class parsed sections to variables allowing the software to perform actions based on the outputs.
- Custom SD and GPS classes were written encapsulating existing SD and GPS libraries. This allowed the changing of just the custom class if an alternative component replaced an existing one.
- The SD class initialised the FlashAir SD card at the start of every recording instead of just during the startup routine allowing the SD card to be removed and reinserted without a system reboot.
- A dynamic header was generated automatically for each file containing the date, sample frequency and column titles for all data.
- An error function was designed to alert the user of easy to fix errors through the RGB LED such as: battery low voltage/over voltage (magenta), SD card initialisation failure (cyan) and no valid GPS lock (white).
- Through the use of the config file the recording sample rate of the drifter could be altered to 100 mHz, 200 mHz, 1, 2, 5 or 10 Hz, the speeds supported by the GPS FeatherWing.

2.4 Software (LabVIEW)

Figure 1b shows the Graphical User Interface (GUI) of the custom LabVIEW application written to retrieve recorded data on the drifter. LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) is a graphical programming language and design environment developed by National Instruments and is most commonly used for data acquisition, machine vision and instrumentation interfacing/control [18]. Figure 13 shows a simplified view of the queued state machine producer consumer architecture used in the program.

Utilising the FlashAirs embedded web server to gain access to the log files the FlashAir was first changed from default Access Point (AP) mode into Station Mode (STA), allowing an automatic connection to preconfigured 802.11 WiFi networks.

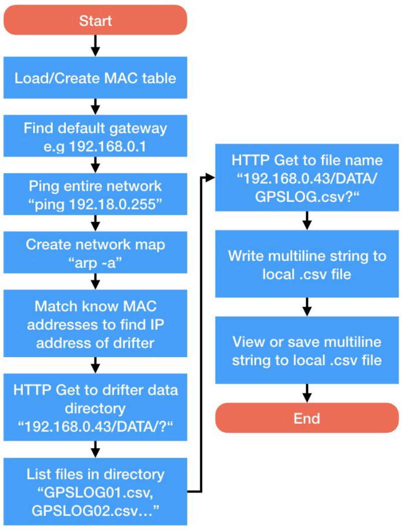


Figure 12: High-level process diagram of LabVIEW software.

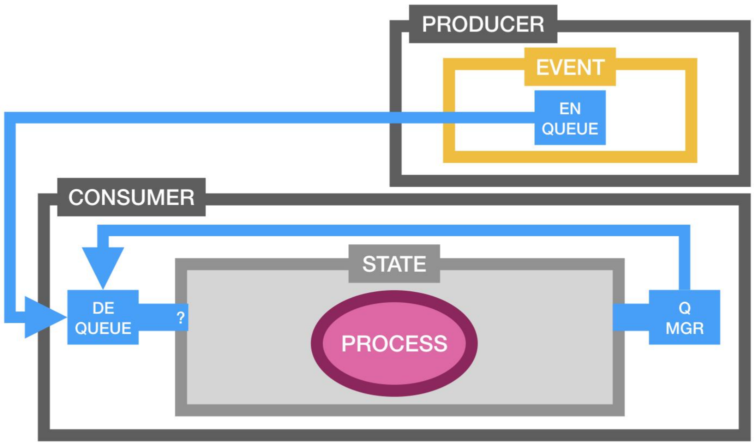


Figure 13: Illustration of queued state machine producer consumer architecture used in LabVIEW software.

The LabVIEW application interfaced with the command line terminal and first found the default gateway address of the network and pinged all connected devices. A network map was then built containing all connected hardware, listing the MAC addresses and associated Internet Protocol (IP) addresses. If any MAC addresses were present in both the map and lookup table (loaded/created by the application containing drifter MAC addresses), a HTTP Get command was issued to the data file directory of the associated IP address. A list of data files was generated by scanning the output string of the Get command and counting the matching instances of the data file prefix "GPSLOG". A Get command could then be issued to the valid filename within the data directory, listing the contents of the .csv file. The multiline output string containing the data could then be viewed or saved to a file on the host computer. A "download all data" feature was implemented allowing all data on a drifter to be saved in separate files by clicking one button. Figure 12 visually demonstrates this process.

The config file mentioned in Section 2.3 could also be configured and uploaded to the root directory of the SD card at any time while a connection was established using a HTTP Post command.

When sending a HTTP Get command the application would ping the IP address of the drifter to validate its connection. If the returned string contained the word "unreachable" the drifter was deemed to be disconnected from the network and further commands were not issued. An additional usability feature was the triggering of a "mouse busy" event when issuing terminal commands, preventing the user from selecting other options until the current task had completed.

3 System Trials & Analysis

3.1 3D Printed Container Porosity

As stated in Section 2.2.2 multiple small scale experiments were conducted to determine optimum print settings for an IP67 rated enclosure. For ABS the optimum parameters were found to be a nozzle temperature of 235°C, platform temperature of 105°C, 10%[4] infill, 3 top layers, 3 bottom layers, 3 outer walls and a base print speed of 80 mm/s. Cylindrically shaped enclosures outperformed rectangular counterparts in both porosity and submersion experiments due to the lack of 90° corners where print deformities were more likely to occur. Allocating a random start point for each successive layer further increased performance limiting localised deformities, with the thermoplastic able to fill/correct voids left when deformities were distributed randomly across the model. Exposing the prints to heated acetone vapours as a final stage of preparation yielded a smooth surface finish, melting and combining outer layers to appear homogeneous and provide extra water ingress protection [14]. The acetone smoothing process was used in the finishing of the final enclosure.

3.2 Miniature Enclosure River Trail

Results from the porosity experiments were used to design a half scale spherical drifter enclosure that could be tested in a river. The aim of the experiment was to gain an appreciation of the real-world challenges and

[4] Infill refers to the structure printed inside an object. Extruded in a predefined design percentage and pattern set via software, infill percentage and pattern influence print weight, material usage, strength, print time and decorative properties [19].

enviroments observed by a river drifter. 10 miniature enclosures were manufactured, 6 in red and white ABS, 2 in yellow and red ABS and 2 in water soluble Polyvinyl Alcohol (PVA). The ABS spheres were placed in the main flow of a river to observe their movements through obstacles, height of floatation and visibility. When caught in debris the white spheres were difficult to identify due to white foam generated by the flow, making it problematic to decide whether the enclosure had been dragged underwater. The PVA balls were placed at the top of a 3 tier weir to observe how they would perform in the hydraulic features, confirming an IP67 rating necessary. The PVA balls were lost in the river but dissolved due to the non-toxic solubility of the material. From the observations the decision to use yellow and red ABS was made to provide greater contrast to the surroundings. The buzzer mentioned throughout the paper was included to inform the user of submersion as the buzzer was not audible underwater.

3.3 GPS Experiments

One of the concerns with a spherical drifter was the position of the GPS antenna with respect to the water and how this would effect signal quality. Initial testing was conducted to ensure the project was viable by designing and manufacturing a test rig that allowed the placement of a simplified data logger, using the same MTK3339 GPS module, at varying heights above the surface of a 32 l water container. The levels were denoted 1-24 and increased in 30 mm increments. Experiments were conducted with a sample rate of 1 Hz at 3 different levels, L7, L8 and L11 that were 24, 54, and 144 mm away from the waters surface respectively. Each level was tested without water to act as a control, with and without an external antenna. Each test lasted between 4-5 minutes, with a 1 minute interval between, and was repeated 4 times. The 1 minute period between tests allowed the GPS module a "cold start" when acquiring GPS almanac data, the initial data needed about the GPS network before a connection is established. The first experiment for each test was disregarded as the startup transient phase of the module getting an initial fix would cause irregularities in the data, the remaining 3 were averaged together. The experiment was conducted in an environment where $\approx 130°$ of the sky was unobstructed while also having no view of the horizon, mimicking a 3.5 m wide river. The experimental setup can be seen in Figure 25 in [Appendix 4].

3 parameters were of interest in the experiment, the HDOP, signal quality and number of satellites used. The HDOP, a measure of the accuracy of the signal received based on transmitter range errors, was measured in Figure 14 at a maximum dilution of 3, representing a good level of accuracy able to make reliable in-route navigational suggestions if needed, and could be seen to improve over the period of the experiment.

Signal quality is a measure of how many valid fixes are present, 1 being the minimum required. Test data in Figure 15 showed all experiments experienced a minimum average of 1 fix, increasing to multiple fixes as the antenna height above water increased.

The number of satellites was an indicator of active satellites being used to acquire a fix. All experiments displayed the minimum of 4, with the "With Water/With Antenna" increasing to the maximum possible of 12 at periods during the L8 experiments in Figure 16. Tests at L7 and L8 showed an increase in the number of satellites used with water present, compared to without.

All results appeared positive and demonstrated the use of GPS in close proximity to water was viable, allowing work on the project to commence. [Appendix 7] displays the additional data captured during the experiment.

3.4 Integrated System Tests

With an appropriate o-ring installed the drifter was placed under 304 mm of water for 1 hour and left floating for 24 hours to give validity to the IP67 rating. While floating the depth of submersion was measured and was within $\pm$ 5 mm of the calculated value in Section 2.2.1.

The drifter had an operating battery voltage of 4.2 V to 3.6 V, the minimum required by the integrated voltage regulator to power the FlashAir. The system lasted 2 hours 59 minutes at a constant current of 150 - 160 mA and generated a file size of 819 bytes. A graph showing the battery voltage drop over the period can be seen in [Appendix 6].

The accuracy of the drifter was measured by walking the full system in a 0.7 km loop and comparing the recorded data against perfect latitude and longitude coordinates. The drifter was found to have a standard deviation of 1.25 m and a mean accuracy of 1.19 m, 0.61 m less than the manufacturers specified 1.8 m. A map of the planned route and actual route taken, along with data collected can be seen in [Appendix 8].

4 Conclusions

This paper has presented a 3D printed miniature river drifter and accompanying software architecture capable of scaling to large scale deployment at a total cost of £169.56, substantially cheaper than existing systems [20, 21]. The drifter and software developed were created as a proof of concept and requires the

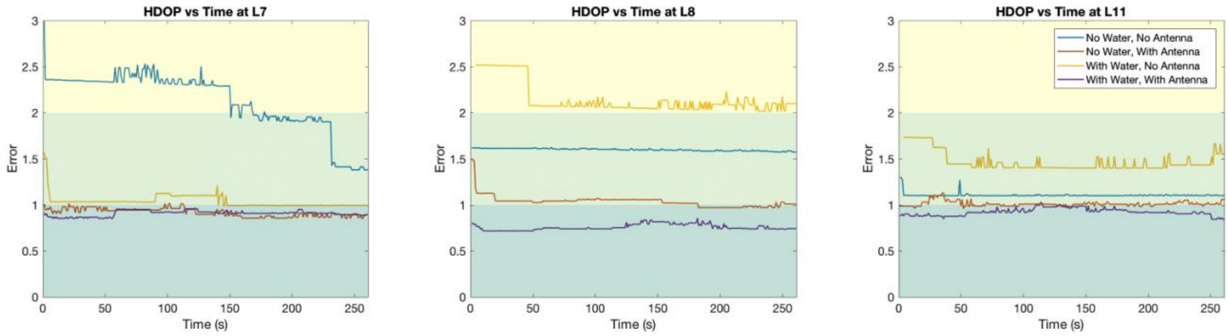


Figure 14: Mean HDOP at different heights above water. < 1 = Ideal, 1-2 = Excellent, 2-5 = Good.

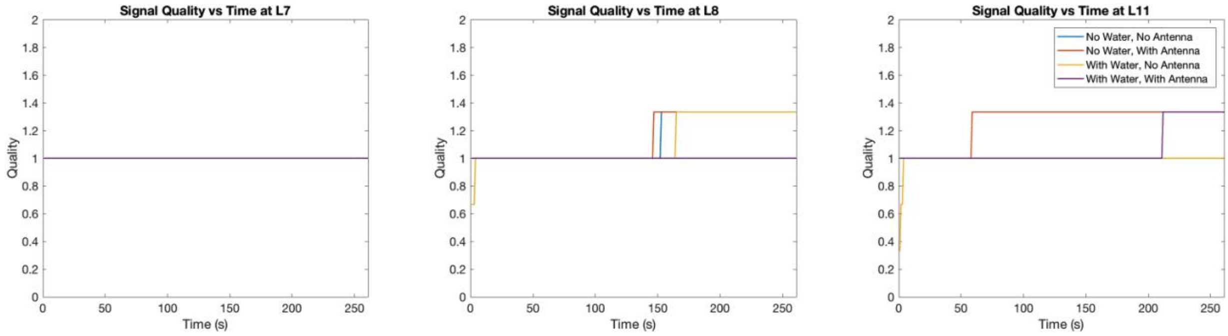


Figure 15: Mean quality of fix at different heights above water. 0 = Invalid, 1 = GPS fix, 2 = Diff. GPS fix.

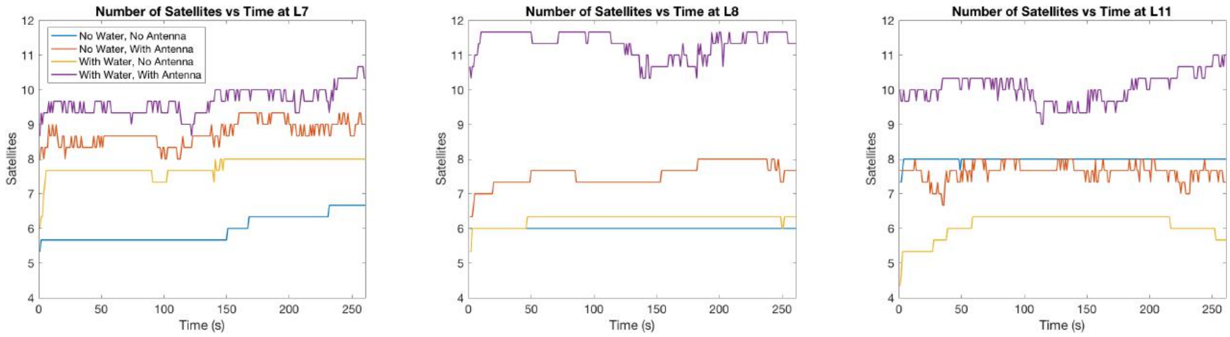


Figure 16: Mean number of satellites used at different heights above water.

resolution of several limitations before being practically applied, most notably the length of print time for each enclosure far exceeds that of other manufacturing methods such as injection moulding, a process that would allow the mass production of the design [22, 23]. The need for human presence throughout deployment, due to a lack of live tracking, combined with short battery life meant long term deployment was not possible but could be easily remedied using a larger battery and a Global System for Mobile communications (GSM) module. Furthermore the longterm effects of water on ABS require investigation to determine what, if any, degradation occurs. Future work could focus on features such as an adaptive sample rate that changes based on the measured speed of the drifter, increasing the sample rate in fast flowing currents. Mechanical design modifications could be made such as annealing the enclosure material to increase strength in the direction of printing. Solar or kinetic energy harvesting could also be investigated along with incorporating additional sensors to measure parameters such as acceleration, river depth, salinity and turbidity.

5 Acknowledgements

I would like to thank my project supervisor Dr Paul Lepper of the Wolfson School of Mechanical, Electrical and Manufacturing Engineering at Loughborough University for his continued enthusiasm, support and guidance throughout the project. He consistently allowed the work conducted to be my own, steering me in the right direction when needed. I would also like to thank Dr Robert Edwards as the projects examiner, providing valuable guidance on the projects direction. Finally I would like to express my gratitude to Jonathan Witherstone for aiding me with the miniature river trials, providing the necessary equipment and expertise to perform a safe experiment.

References

[1] P. Thangarasu, R. Srinivasan, S. Muthukumaravel, and T. Sudhakar, "Multidisciplinary real time data management of gprs based coastal drifter and insat based automatic weather station," in *2016 International Conference on Communication and Signal Processing (ICCSP)*, April 2016, pp. 0950–0955.

[2] H. Chien, H. Y. Cheng, and Y. Zhong, "Validating of coherent-on-receive marine radar in nearshore using drifter cluster," in *OCEANS 2017 - Aberdeen*, June 2017, pp. 1–4.

[3] [Accessed: 1 May. 2018]. [Online]. Available: http://secoora.org/classroom/flowing_ocean/tracers

[4] J. Swallow, "A neutral-buoyancy float for measuring deep currents," *Deep Sea Research (1953)*, vol. 3, no. 1, pp. 74 – 81, 1955.

[5] K. Weekly, L. Anderson, A. Tinka, and A. M. Bayen, "Autonomous river navigation using the hamilton-jacobi framework for underactuated vehicles," in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 828–833.

[6] D. Boydstun, M. Farich, J. M. III, S. Rubinson, Z. Smith, and I. Rekleitis, "Drifter sensor network for environmental monitoring," in *2015 12th Conference on Computer and Robot Vision*, June 2015, pp. 16–22.

[7] [Accessed: 1 May. 2018]. [Online]. Available: https://www.adafruit.com/product/2796

[8] [Accessed: 1 May. 2018]. [Online]. Available: https://www.amazon.co.uk/SD-card-MicroSD-Transflash-adapter/dp/B00IA1LGG8

[9] [Accessed: 1 May. 2018]. [Online]. Available: https://www.flashair-developers.com/en/discover/usage/

[10] [Accessed: 1 May. 2018]. [Online]. Available: https://learn.adafruit.com/adafruit-ultimate-gps-featherwing

[11] [Accessed: 1 May. 2018]. [Online]. Available: https://spaceplace.nasa.gov/gps/en/

[12] [Accessed: 1 May. 2018]. [Online]. Available: https://learn.sparkfun.com/tutorials/gps-basics

[13] [Accessed: 1 May. 2018]. [Online]. Available: http://aprs.gids.nl/nmea/

[14] T. Quinn, "Eld030 project; research report," Wolfson School of Mechanical, Electrical and Manufacturing Engineering, Loughborough University, Project Report, 2017.

[15] [Accessed: 1 May. 2018]. [Online]. Available: https://www.math.uh.edu/ jingqiu/math4364/root_nonlinear.pdf

[16] [Accessed: 1 May. 2018]. [Online]. Available: https://www.hestay.nl/en/3d-print-support/3d-printing-problems-solutions/pillowing/

[17] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless charging technologies: Fundamentals, standards, and network applications," *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1413–1452, Secondquarter 2016.

[18] Y. Xu and F. Yu, "Research on the image acquisition and camera control of machine vision camera based on labview," in *2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, vol. 2, Aug 2013, pp. 499–502.

[19] [Accessed: 1 May. 2018]. [Online]. Available: https://3dplatform.com/3d-printing-tech-tips-infill-percentage-and-pattern-explained/

[20] B. Chadwick, C. Katz, J. Ayers, J. Oiler, M. Grover, A. Sybrandy, J. Radford, T. Wilson, and P. Salamon, "Gps drifter technologies for tracking and sampling stormwater plumes," in *OCEANS 2016 MTS/IEEE Monterey*, Sept 2016, pp. 1–10.

[21] Z. Yang, M. Li, and Y. Liu, "Sea depth measurement with restricted floating sensors," in *28th IEEE International Real-Time Systems Symposium (RTSS 2007)*, Dec 2007, pp. 469–478.

[22] A. Tinka, M. Rafiee, and A. M. Bayen, "Floating sensor networks for river studies," *IEEE Systems Journal*, vol. 7, no. 1, pp. 36–49, March 2013.

[23] W. K. Lee, S. J. Q. Ho, and M. Schubert, "Multi-source energy harvesting and storage for floating wireless sensor network nodes," in *2016 IEEE Industrial Electronics and Applications Conference (IEACon)*, Nov 2016, pp. 67–72.

Appendices

1 System Schematics

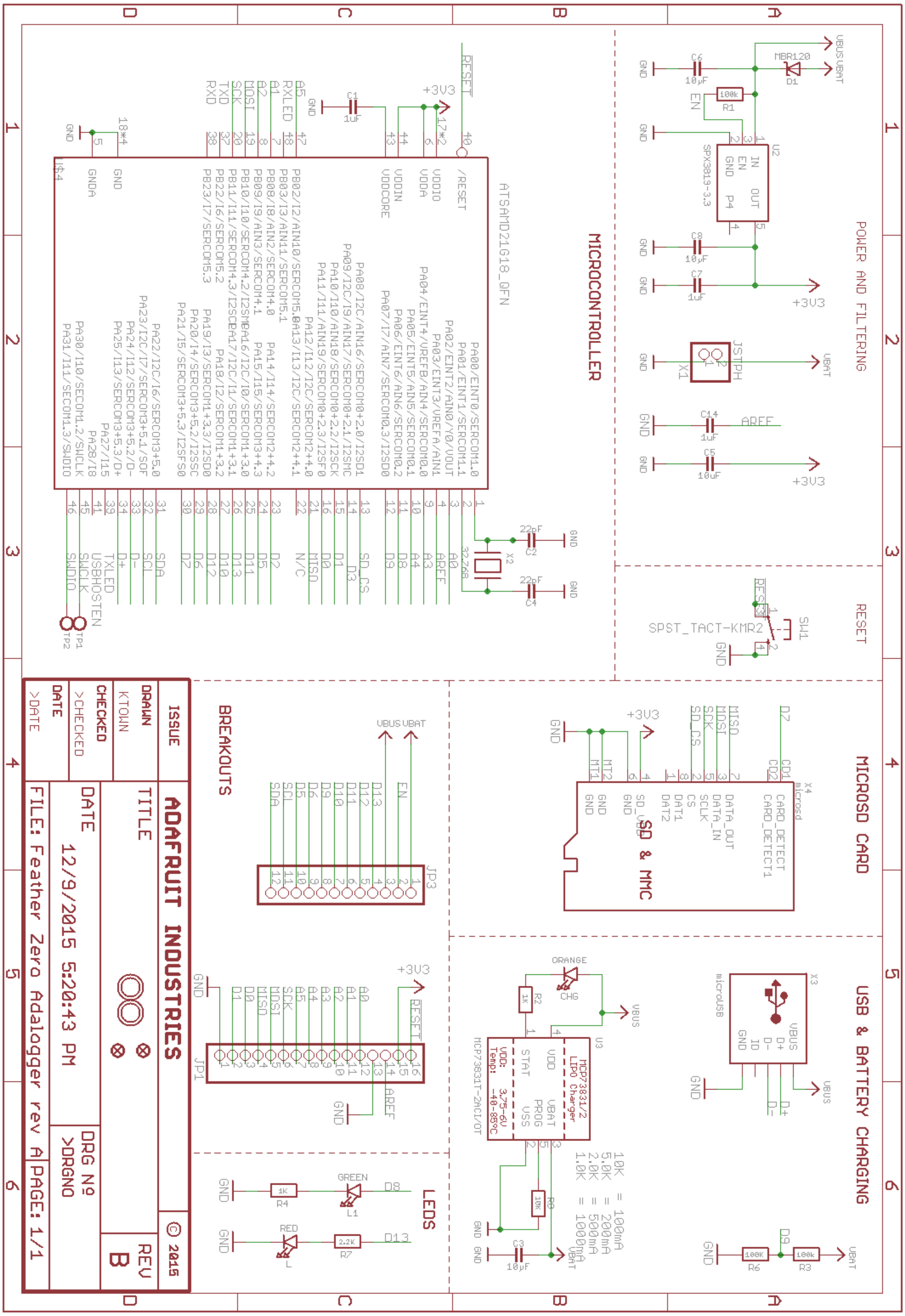


Figure 17: Adafruit Feather M0 Adalogger schematic [7].

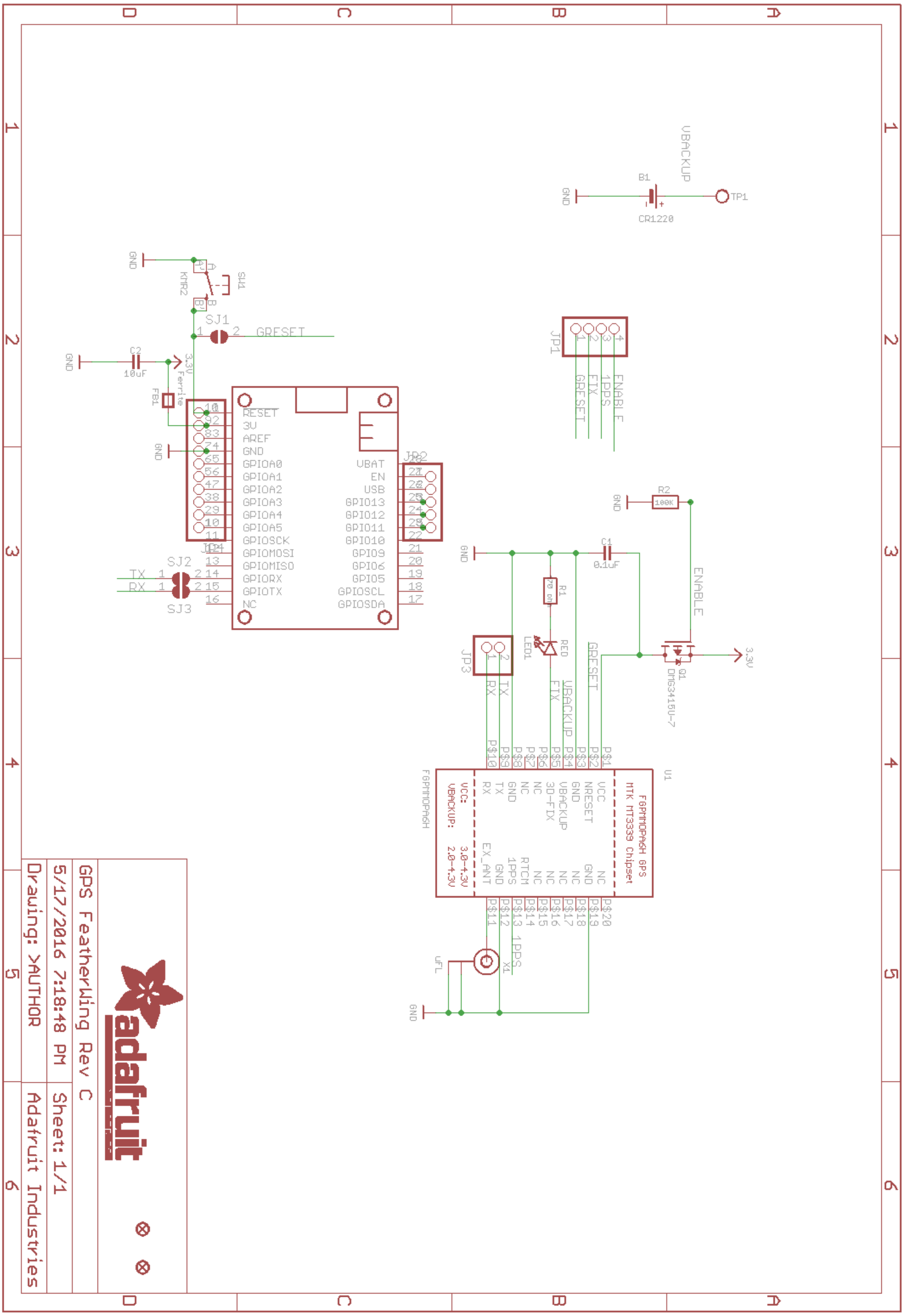


Figure 18: Adafruit GPS FeatherWing schematic [10].

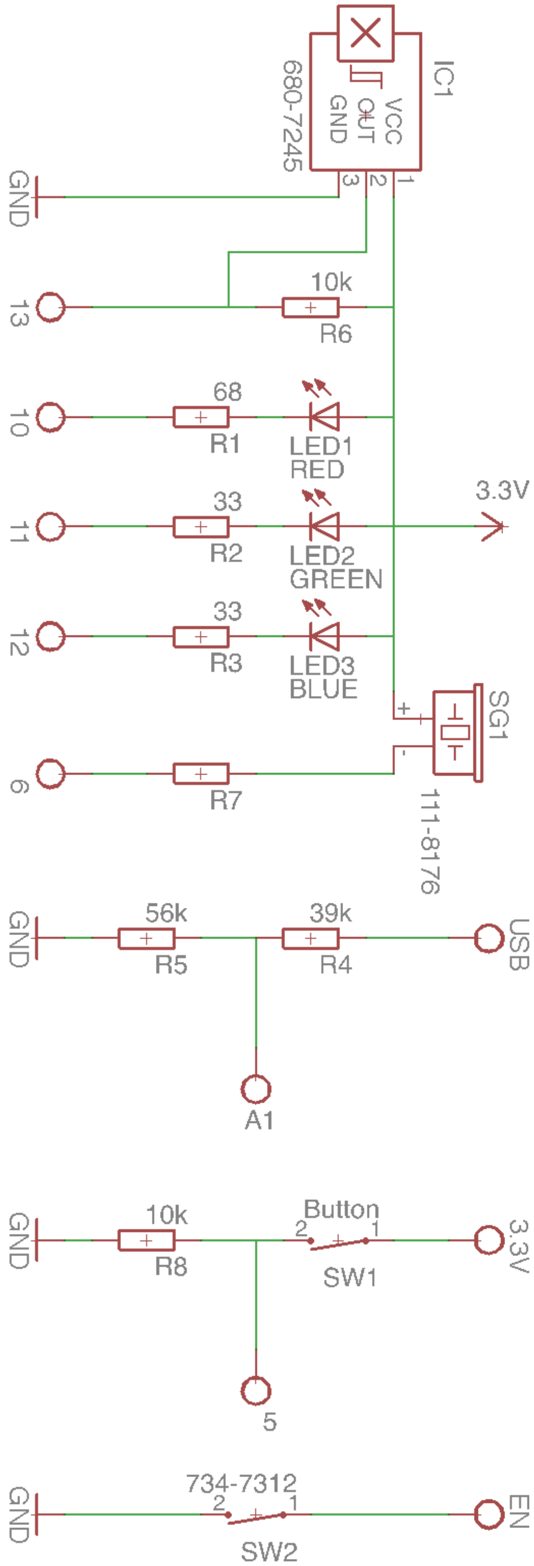


Figure 19: Additional components schematic. Pins relate to Adalogger I/O.

2 BOM

Item	Supplier	Part number	Quanitiy	Price per unit (£)	Price (£)
Adalogger M0	RS Components	124-5517	1	16.38	16.38
GPS Feather Wing	Pimoroni Ltd	ADA3133	1	40.00	40.00
Proto FeatherWing	Pimoroni Ltd	ADA2884	1	5.50	5.50
Inductive Charging Modules	Cool Components	1595	1	8.51	8.51
Toshiba FlashAir	Amazon	N/A	1	29.97	29.97
SD Card Adapter	Amazon	N/A	1	19.38	19.38
500 mAh Battery	Amazon	N/A	1	2.75	2.75
Unipolar Hall effect Sensor	RS Components	680-7245	1	0.84	0.84
SPDT Slide Switch	RS Components	734-7312	1	2.55	2.55
RGB Common Anode LED	Ebay	N/A	1	0.20	0.20
DC Buzzer	RS Components	111-8176	1	0.28	0.28
Tactile Momentary NO Switch	Ebay	N/A	1	0.85	0.85
0.25 W Misc Resistors	Ebay	N/A	8	0.05	0.40
Brass Inserts and Screws	Ebay	N/A	1	5.00	5.00
Ridgid Ink 1.75mm ABS Filaments	Amazon	N/A	1	36.95	36.95
Total					169.56

Table 5: Bill of Materials

3 NMEA Sentence Structure

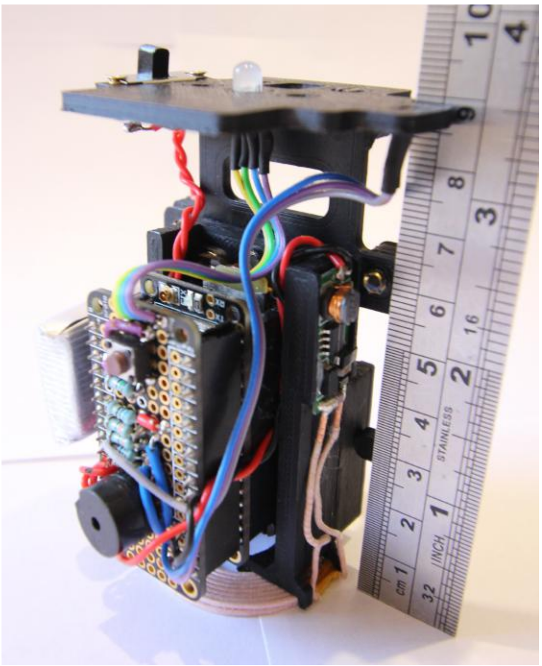
```
eg4. $GPRMC,hhmmss.ss,A,llll.ll,a,yyyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh
1    = UTC of position fix
2    = Data status (V=navigation receiver warning)
3    = Latitude of fix
4    = N or S
5    = Longitude of fix
6    = E or W
7    = Speed over ground in knots
8    = Track made good in degrees True
9    = UT date
10   = Magnetic variation degrees (Easterly var. subtracts from true course)
11   = E or W
12   = Checksum
```

Figure 20: Global Positioning Recommended Specific data sentence structure [13].

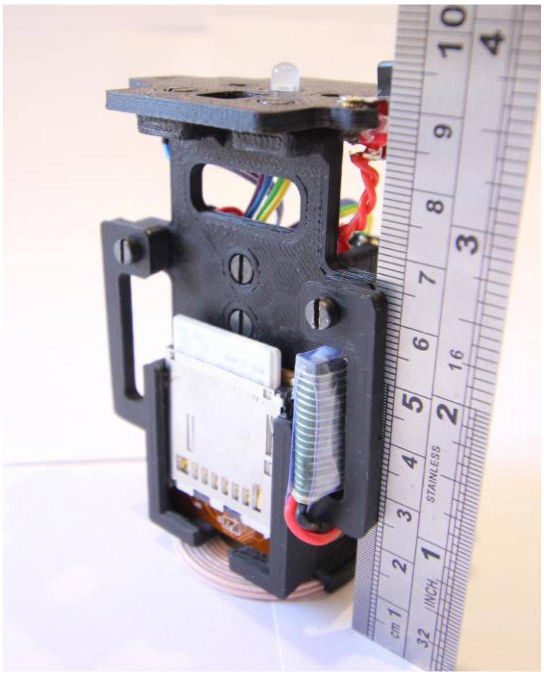
```
eg3. $GPGGA,hhmmss.ss,llll.ll,a,yyyyy.yy,a,x,xx,x.x,x.x,M,x.x,M,x.x,xxxx*hh
1    = UTC of Position
2    = Latitude
3    = N or S
4    = Longitude
5    = E or W
6    = GPS quality indicator (0=invalid; 1=GPS fix; 2=Diff. GPS fix)
7    = Number of satellites in use [not those in view]
8    = Horizontal dilution of position
9    = Antenna altitude above/below mean sea level (geoid)
10   = Meters  (Antenna height unit)
11   = Geoidal separation (Diff. between WGS-84 earth ellipsoid and
       mean sea level.  -=geoid is below WGS-84 ellipsoid)
12   = Meters  (Units of geoidal separation)
13   = Age in seconds since last update from diff. reference station
14   = Diff. reference station ID#
15   = Checksum
```

Figure 21: Global Positioning System Fit data sentence structure [13].

4 Project Images



(a) Front.



(b) Rear.

Figure 22: Electronics cradle full assembly.

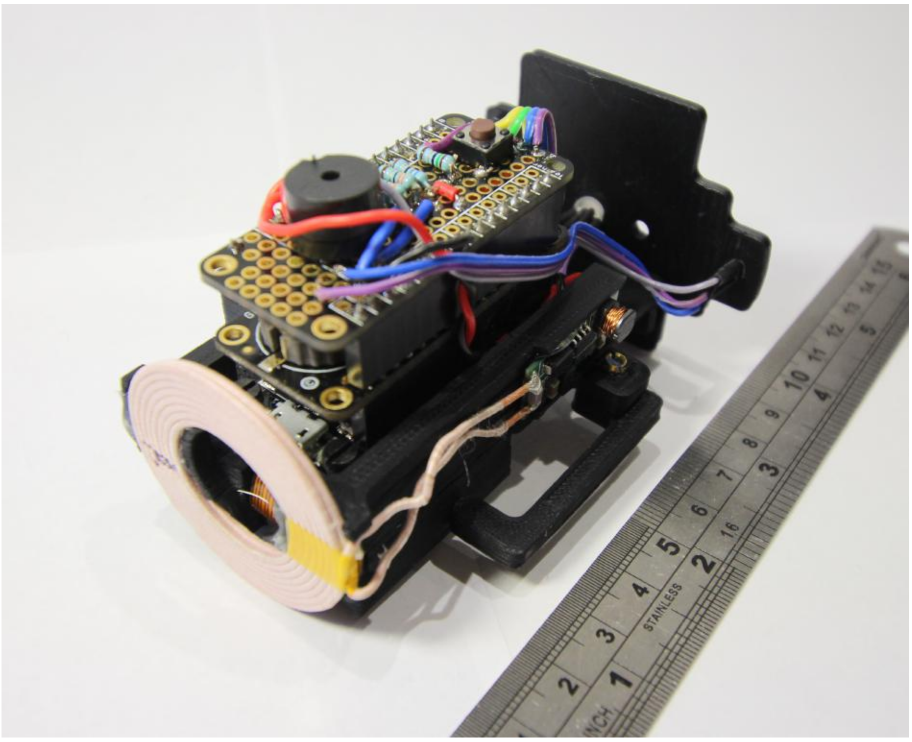


Figure 23: Electronics cradle assembly base.

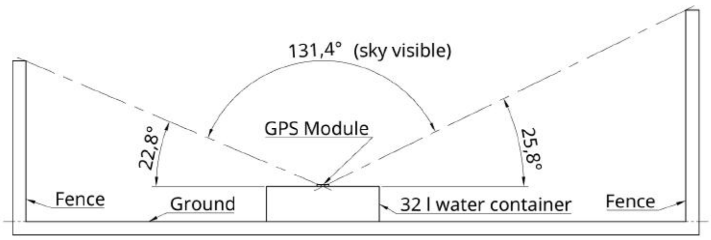
(a)

(b)

(c)

Figure 24: Stages of assembly.

(a) Rig for testing GPS signal quality over water.



(b) Diagram of test rig and experimental setup.

Figure 25: GPS signal experimental apparatus.

5 Data File Example

18/5/7	F = 1.00Hz																				
H	M	S	ms	Latitude	North/South	Latitude Fixed	Latitude Degrees	Longitude	East/West	Longitude Fixed	Longitude Degrees	HDOP	Geoidal Seperation	Magnetic Variation	Speed (knots)	Angle (Deg)	Altitude (M)	Quality	Satellites	Fix	Battery (V)
12	52	51	0	5246.51	N	527751450	52.78	113.04	W	12173983	-1.22	1.19	47.50	0.00	0.41	260.49	37.60	1	8	1	3.90
12	52	54	0	5246.51	N	527751450	52.78	113.04	W	12174033	-1.22	1.19	47.50	0.00	0.08	286.64	37.60	1	8	1	3.91
12	52	55	0	5246.51	N	527751450	52.78	113.04	W	12174050	-1.22	1.19	47.50	0.00	0.06	260.39	37.60	1	8	1	3.91
12	52	56	0	5246.51	N	527751450	52.78	113.04	W	12174050	-1.22	1.19	47.50	0.00	0.05	354.22	37.60	1	8	1	3.91
12	52	57	0	5246.51	N	527751450	52.78	113.04	W	12174050	-1.22	1.23	47.50	0.00	0.13	283.08	37.60	1	8	1	3.91
12	52	58	0	5246.51	N	527751450	52.78	113.04	W	12174066	-1.22	1.19	47.50	0.00	1.67	226.88	37.70	1	8	1	3.91
12	52	59	0	5246.51	N	527751400	52.78	113.05	W	12174183	-1.22	1.19	47.50	0.00	2.36	223.04	37.60	1	8	1	3.91
12	53	0	0	5246.51	N	527751333	52.78	113.05	W	12174316	-1.22	1.19	47.50	0.00	2.58	223.06	37.60	1	8	1	3.91
12	53	1	0	5246.51	N	527751233	52.78	113.05	W	12174450	-1.22	1.19	47.50	0.00	3.07	211.26	37.40	1	8	1	3.91
12	53	2	0	5246.51	N	527751133	52.78	113.05	W	12174600	-1.22	1.19	47.50	0.00	2.99	218.94	37.40	1	8	1	3.91

Figure 26: Example of data .csv file created by the drifter.

6 Battery Discharge Graph

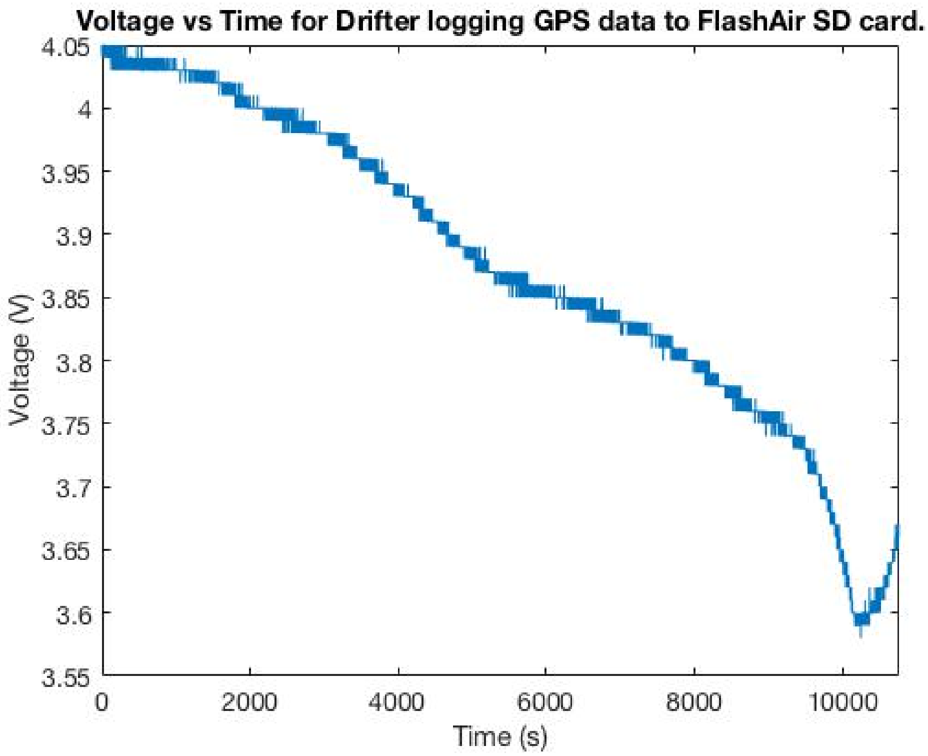


Figure 27: Battery voltage drop over time.

7 Additional GPS Experimental Data

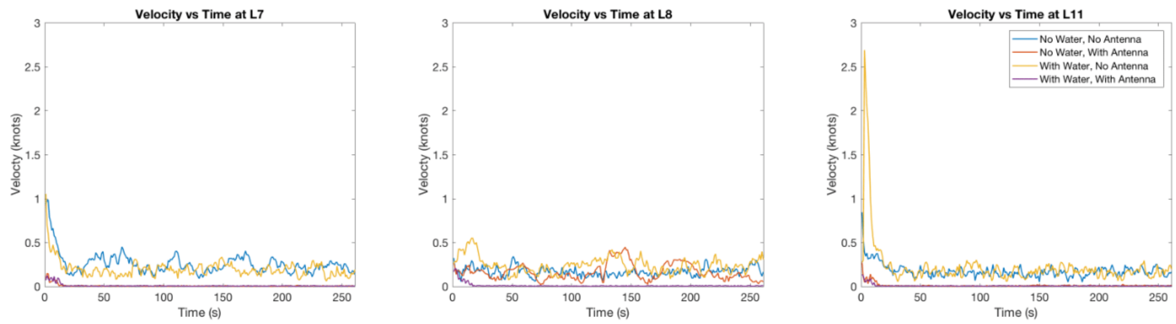


Figure 28: Mean velocity in knots at different heights above water. Note: experiment was stationary.

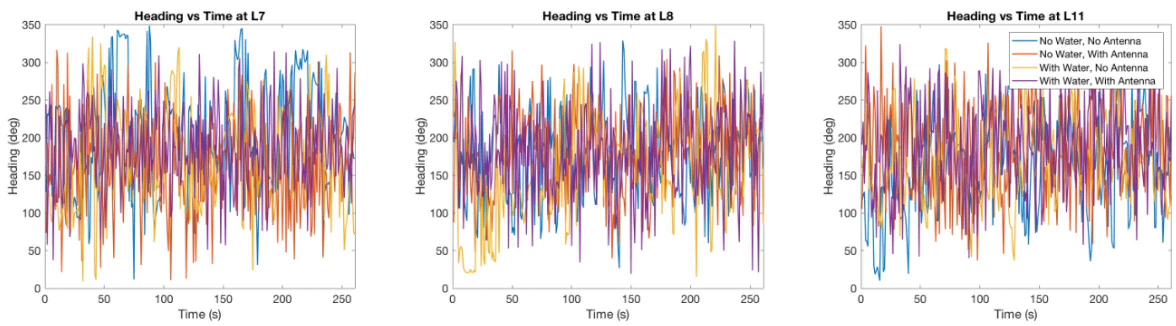


Figure 29: Mean heading at different heights above water. Note: experiment was stationary.

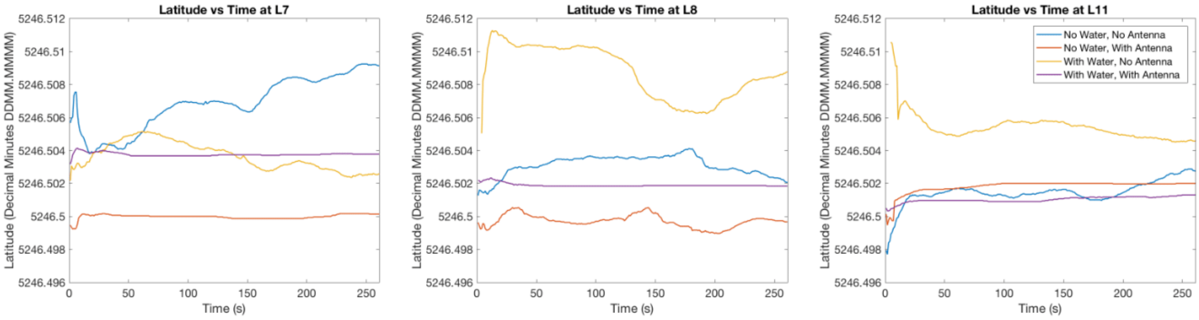


Figure 30: Mean latitude at different heights above water. Note: experiment was stationary.

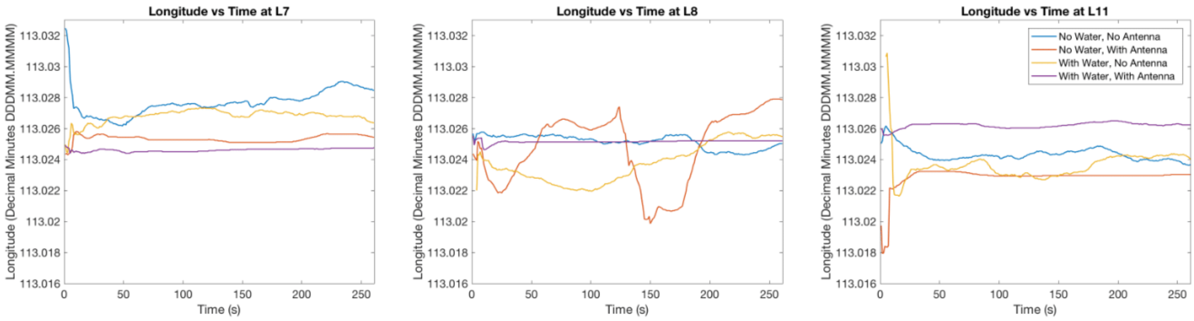


Figure 31: Mean longitude at different heights above water. Note: experiment was stationary.

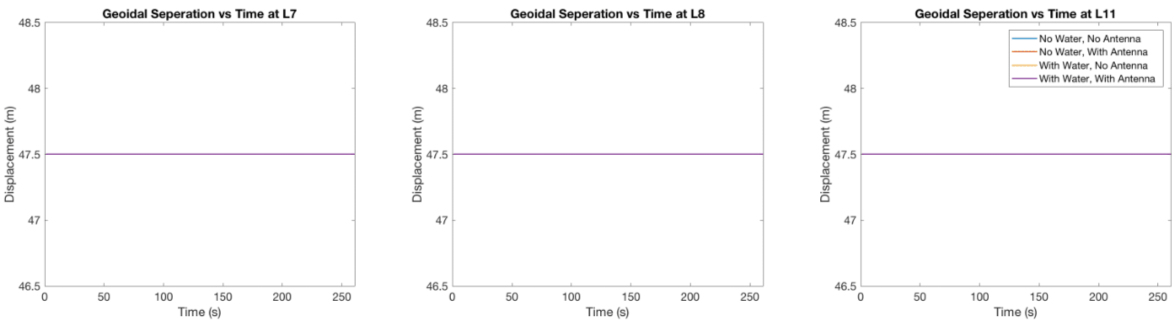


Figure 32: Mean geoidal seperation at different heights above water. Note: experiment was stationary.

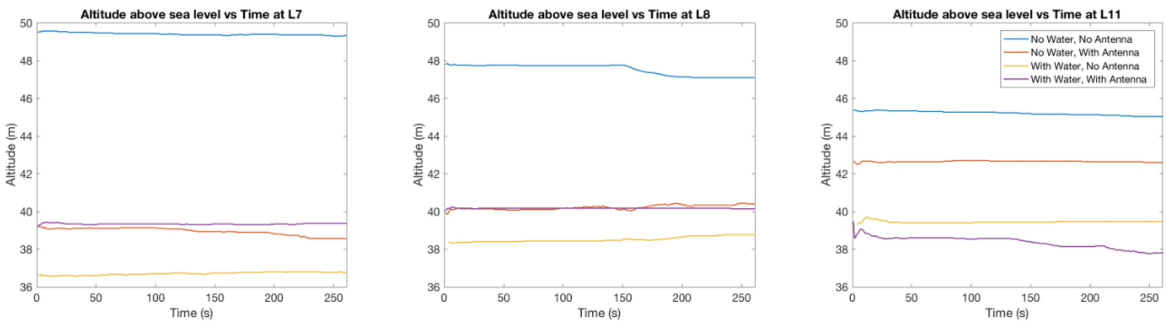


Figure 33: Mean altitude at different heights above water.

8 System Accuracy Experiment

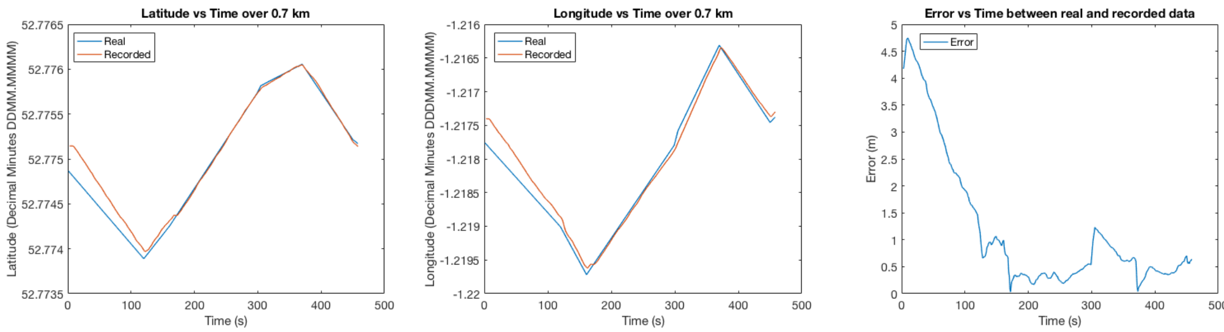


Figure 34: Data recorded during full system testing.

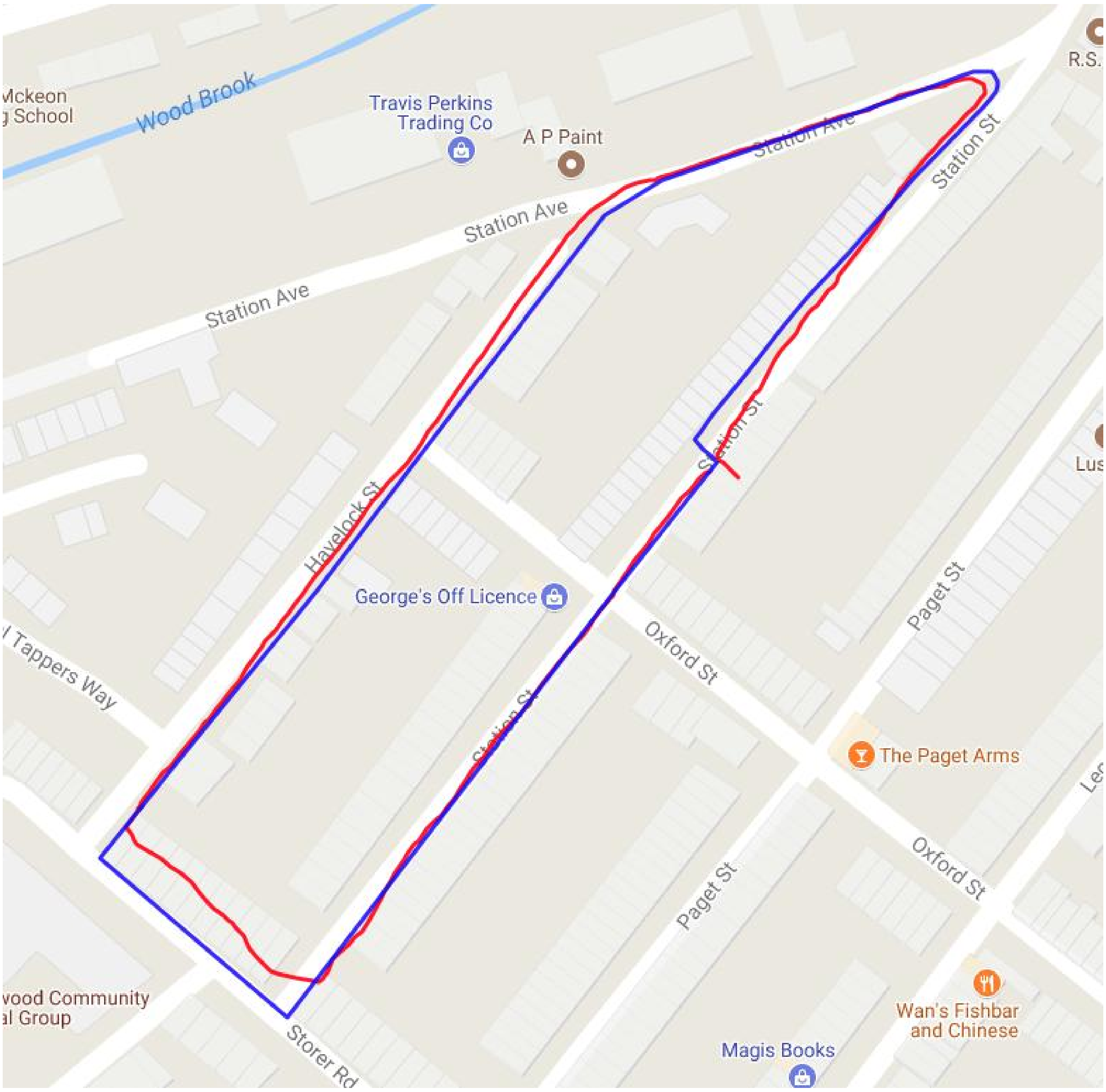


Figure 35: Path taken during full system testing. Blue - real path, Red - recorded.
Centre: 52.77546, -1.21613.