



Protokoll zum Projekt: Wetterballon

Patrick Vogt (924789)

Dietmar Sach (924738)

durchgeführt von Mai 2013 bis 19. Juli 2013

Protokoll eingereicht am 19. Juli 2013

Start des Wetterballons am 02. August 2013

Copyright © 2013 Patrick Vogt, Dietmar Sach

Grundlagen	3
1.1. Motivation.....	3
1.2. Herausforderungen.....	3
1.3. Grundsätzliches zur Funktion eines Wetterballons und zur Atmosphäre	4
1.4. Zusammenfassung der Reise des Wetterballons.....	7
1.5. Grundsätzliches zur Positionsbestimmung.....	7
2 Projekt.....	8
2.1. Transport-Box.....	8
2.2. Ballon und Fallschirm	10
2.3. Gewicht	10
2.4. Hardware.....	11
2.4.1 GnuBLIN	11
2.4.2 Webcam	13
2.4.3 PT1000-Messbrücke	13
2.4.4 UMTS-Stick	14
2.4.5 GPS-Empfänger	15
2.5. Stromversorgung.....	15
2.6. Software	17
2.6.1 Skript für den Autostart der Software	17
2.6.2 Watchdog-Timer	17
2.6.3 Trackingsystem	18
2.6.4 Temperaturanwendung	19
2.6.5 Webcam-Aufzeichnung.....	20
2.7. Testphase und Vorbereitung.....	20
2.7.1 Test der Wärmeisolation und Taschenwärmer.....	20
2.7.2 Test der Temperatur-Messeinheit	22
2.7.3 Test der Webcam.....	22
2.7.4 Test des Trackingsystems	22
2.8. Durchführung	23
2.9. Materialkosten.....	23
3 Fazit	24
4 Anhang.....	24
4.1. Literaturverzeichnis.....	24
4.2. Abbildungsverzeichnis	24
4.3. Abkürzungsverzeichnis	25
4.4. Quellcode.....	26
4.5. Erklärung	26

Grundlagen

1.1. Motivation

Von der Faszination des Alls getrieben und inspiriert durch andere Wetterballonprojekte von Entwicklern aus Nah und Fern, beschlossen wir am Anfang des Sommersemesters 2013 im Fach Embedded Linux ein Wetterballonprojekt zu starten. Unser primäres Ziel ist es, möglichst spektakuläre Bilder und Videos aus großer Höhe zu machen. Des Weiteren wollen wir während des gesamten Flugs Temperaturmessungen durchführen. Mit diesen Daten lässt sich der Verlauf der Temperatur mit steigender Höhe herausfinden.

1.2. Herausforderungen

Es war zunächst erforderlich, sich die möglichen Herausforderungen und Komplikationen vor Augen zu führen. Da ein Wetterballon nach dem Start unbeaufsichtigt fliegt und unter schwierigen Bedingungen funktionieren muss, sind hohe Anforderungen an die Organisation und an die Planung gestellt. Folgende Kernfragen haben wir uns gestellt und beantwortet:

Welche sensorischen Daten möchten wir erheben?

- Temperatur, Position, Kamerabilder

Wie schwer darf die Last sein und wie viel/welches Gas benötigen wir?

- 1 – 1,5 kg, dafür wird 1 – 1,5 m³ Helium benötigt

Wie finden wir den Wetterballon nach der Landung wieder?

- GPS-Koordinaten werden mittels eines GPS-Empfängers empfangen und mittels eines UMTS-Sticks über das Handynetz an uns weiter geleitet.

Wird die Hardware in dünner Luft und bei niedrigen Temperaturen funktionieren?

- Bei einer zu erwartenden Lufttemperatur von -50° Celsius funktioniert konventionelle Hardware nicht mehr. Daher wird eine isolierende Transport-Box benötigt. Zudem werden chemische Taschenwärmer beigelegt, die über mehrere Stunden hinweg Wärme produzieren.

Was geschieht, wenn die Hardware einfriert?

- In diesem Worst-Case gehen wir davon aus, dass nach der Landung bald die Betriebstemperatur wieder erreicht wird und der Akku noch nicht leer ist. Ein Hardware-Watchdog (WDT) soll das System dann zurücksetzen und so eine Rückkehr in den normalen Betrieb ermöglichen.

Die Stromversorgung

- Um die Elektronik zu betreiben, muss ein ausreichend starker Akku vorhanden sein. Die Effizienz des Akkus wird durch die niedrigen Temperaturen zudem extrem reduziert. Der Akku darf wiederum nicht zu viel wiegen.

Fluggenehmigung

- Um rechtlich auf der sicheren Seite zu sein, wird eine Starterlaubnis von der deutschen Flugsicherung benötigt.

1.3. Grundsätzliches zur Funktion eines Wetterballons und zur Atmosphäre

Zunächst sollte die prinzipielle Funktionsweise eines Wetterballons, der in diesem Experiment als Trägersystem fungiert, erläutern. Verfügt man über eine funktionstüchtige Nutzlast, so kann der Wetterballon mit Helium oder Wasserstoff gefüllt werden. Die Menge des Gases richtet sich nach der Größe des Ballons. Aus Sicherheitsgründen fiel die

Entscheidung auf Helium. Die Heliummenge muss vorher anhand des Gesamtgewichts bestimmt werden.

$$1,0 \text{ m}^3 \text{ He} \cong 1000 \text{ Liter He} \rightarrow \sim 1000 \text{ Gramm Nutzlast}$$

Danach verschließt man die Öffnung und vertäut den Ballon mit Fallschirm und Nutzlast. Nach dem Loslassen beginnt der Ballon aufgrund des Archimedischen Prinzips mit dem Aufstieg. Während des Aufstiegs nimmt die Ausdehnung des Ballons immer mehr zu, da der Außendruck mit steigender Höhe geringer wird. Diese Zunahme der Ausdehnung sorgt dann dafür, dass der Auftrieb konstant bleibt, obwohl die Dichte des Atmosphärgases während des Aufstiegs geringer wird. Wenn die Belastbarkeit des Latexballons schließlich ausgereizt ist, zerplatzt er. Beim Herunterfallen öffnet sich der kleine Fallschirm und sorgt für eine relativ sanfte Landung der Nutzlast.

Mit einer Füllmenge von ca. $1,5 \text{ m}^3$ Helium sollte ungefähr eine Flughöhe von 30.000 Meter erreicht werden können. Da sich das ganze in einiger Höhe abspielt, wird auch kurz auf den Aufbau der Atmosphäre eingegangen. Das besondere Augenmerk liegt dabei auf den beiden untersten Schichten, der Troposphäre und der Stratosphäre. Darüber liegen dann die Mesosphäre, die Thermosphäre und die Exosphäre, welche in den innerplanetaren Raum übergeht. Der hier kurz beschriebene Aufbau der Atmosphärenschichten wird samt ungefähren Grenzhöhen in Abbildung 1 dargestellt.

Im Wesentlichen durchquert der Wetterballon mit dem Gnublin-Board während seiner Reise zwei große Schichten der Atmosphäre. Da wäre zum einen die Troposphäre, die sich vom Erdboden bis zu einer Höhe von 8 km an den Polen und 17 km am Äquator ausdehnt. Die Troposphäre selbst wird in erster Linie nicht durch die Sonneneinstrahlung erwärmt, sondern durch den Erdboden, der die Energie aufnimmt. Daher ist der vertikale atmosphärische Temperaturgradient innerhalb der Troposphäre negativ. Die Troposphäre wird nach oben durch die Tropopause von der Stratosphäre abgegrenzt.

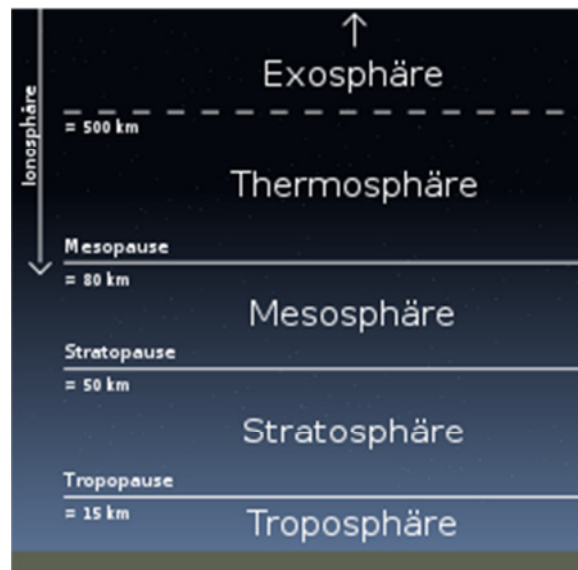


Abbildung 1 - Schematische Darstellung der Atmosphärenschichten mit Angaben der Grenzhöhen aus Wikipedia Atmosphärenschichtung [WIKISTD].

Die einfachste Definition der Tropopause als Ort der „Umkehr des Temperaturgradienten“ ist bei diesem Projekt von besonderer Bedeutung, da unsere Nutzlast über Temperatursensoren verfügt und somit versucht werden kann, die Höhe der Tropopause rechnerisch über Steigrate und Flugdauer zu bestimmen. Die Tropopause stellt also den Ort dar, an dem die Luft aufhört mit größerer Höhe kälter zu werden. In der Troposphäre geht die Temperatur mit steigender Höhe bis auf eine Temperatur von unter -50 Grad Celsius zurück, über diesem Punkt bleibt sie dann einige Zeit konstant.

Mit Erreichen der Grenzsicht beginnt dann die Temperatur zuzunehmen, wobei sie nach einiger Höhe wieder bis knapp an den Nullpunkt heranreicht. Das Wiederansteigen der Temperatur in der Stratosphäre lässt sich durch die darin befindliche Ozonschicht erklären, in der große Teile des UV-Lichts aus der Sonneneinstrahlung absorbiert werden. Auch in horizontaler Richtung ist innerhalb der Troposphäre mit der größten Bewegung des Ballons zu rechnen, da sich das „Wetter“ im Wesentlichen in dieser Schicht abspielt. Dies liegt unter anderem daran, dass die Stratosphäre den Lufttransport „wärmerer“ Troposphärenluft nach oben beendet, da sie ja selbst wieder ansteigende Temperatur besitzt. (vgl. [DEMT1] S. 220f)

1.4. Zusammenfassung der Reise des Wetterballons

Der Wetterballon beginnt nach dem Start mit dem Steigflug. Ein GPS-Empfänger zeichnet die Positionen periodisch auf und übermittelt diese automatisch. Nach etwa 1 – 2 Stunden erreicht der Ballon die erzielte Höhe von 30.000 m in der Stratosphäre. Durch den niedrigen Luftdruck in dieser Region dehnt sich der Ballon bis enorm aus und wird platzen. Danach geht die Nutzlast mit den Ballonresten in den freien Fall über. Ein Fallschirm soll sich nun entfalten, um den Flug abzubremsen. Über die vor und nach dem Aufprall übermittelten GPS-Koordinaten soll die Konstruktion wieder gefunden werden.

1.5. Grundsätzliches zur Positionsbestimmung

Zur Bestimmung der Position der Nutzlast wird das bekannte Global Positioning System (GPS), das ursprünglich vom amerikanischen Militär in der Umlaufbahn der Erde installiert wurde, verwendet. Auf dessen grundsätzliche Funktionsweise wird im Folgenden kurz eingegangen.

Das GPS-System wurde in den neunziger Jahren vom US-Militär in Form einer Vielzahl von Satelliten (24 – 31) im Orbit in einer Höhe von circa 20.000 km installiert. Mit einem geeigneten Empfänger ist es möglich, das Signal von mindestens vier Satelliten zu nutzen, um die eigene Position auf dem Globus bis auf einige Meter genau zu bestimmen. Dazu wird die Laufzeit der Signale gemessen, um den Abstand zwischen Satellit und Empfänger zu berechnen. Für die Ortsberechnung benötigt man also drei Entfernungen und damit drei Signale.

Ein viertes Signal gibt zudem die genaue Uhrzeit vor. Diese wird auf diese Weise synchronisiert, um eine korrekte Entfernungsmessung zu gewährleisten. Über dieses Signal wird auf dem GnuBlin-Board die Systemzeit mit Hilfe des GPS-Service-Daemons *gpsd* automatisch eingestellt, sodass die Aufzeichnung der GPS-Koordinaten über im zeitlichen Verlauf korrekt dargestellt wird.

Die im zivilen Bereich verwendete L1-Frequenz liegt bei 1575,42 MHz. Die Ortsbestimmung über GPS kann, je nach Sichtbarkeit der Satelliten, auf weniger als 10 Meter genau sein. Die Präzision der GPS-Zeit beträgt 14 ns. Eine Beschränkung der Funktion des GPS stellt die Höhenbegrenzung von ungefähr 18 km dar, die für zivile Empfänger gilt. Für Empfänger, die in größerer Höhe funktionieren, ist eine spezielle Lizenz erforderlich. Dies soll den Einsatz von Waffensystemen erschweren. Allerdings funktionieren viele zivile GPS-Sensoren dennoch höher als 18 km noch.

Über die genaue Funktionsweise von GPS könnten noch viele Seiten geschrieben werden, darum wird es bei diesem groben Überblick bleiben.

Für den Fall, dass der GPS-Empfänger aus irgendeinem Grund ausfallen sollte, besteht noch die Möglichkeit, die Konstruktion über das Mobilfunksignal zu orten, indem mittels des verwendeten Mobilfunk-Adapters die Netzwerk- und Zelleninformation ausgelesen und diese über einen Online-Service in Koordinaten übersetzt werden. Hierbei erreicht man allerdings in der Regel nicht annähernd eine mit GPS vergleichbare Genauigkeit und es ist nicht davon auszugehen, dass man durch eine solche Ortung eine erfolgreiche Bergung erreichen könnte. Daher ist die GPS-basierte Lösung zwingend erforderlich.

2 Projekt

2.1. Transport-Box

Um die Nutzlast am Ballon zu befestigen, wird ein Behälter benötigt, der sowohl stabil, als auch möglichst leicht sein und isolierend wirken soll. Wir entschieden uns für eine quaderförmige Kiste aus Platten aus besonders stabilem Polystyrol (Styropor), das auch zur Wärmeisolierung von Gebäuden verwendet wird.

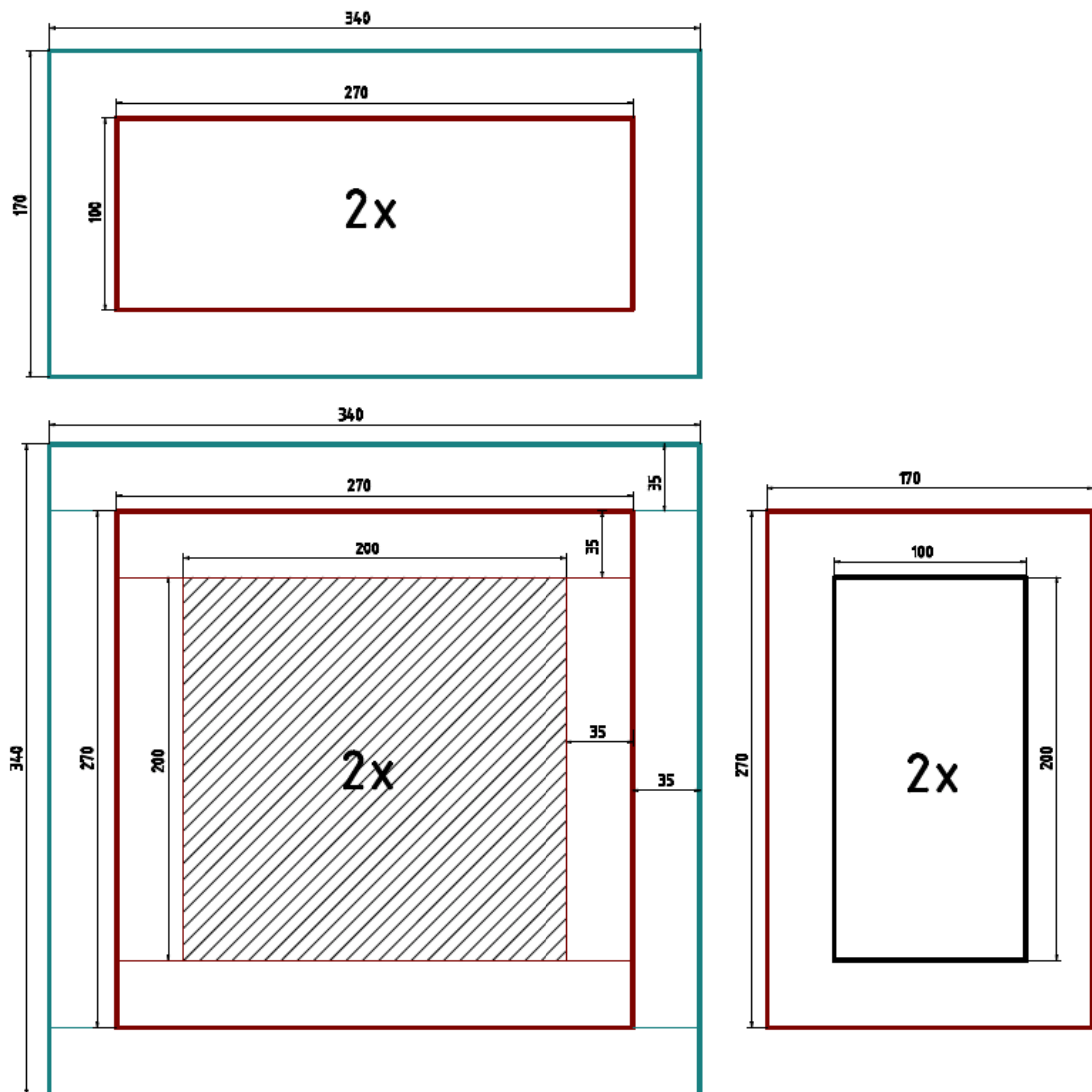


Abbildung 2 - 2D-CAD-Zeichnung der Styroporbox

Da uns eine Wandstärke von 35 mm nicht ausreichend erschien, verwendeten wir für jede Wand jeweils zwei Platten übereinander, was eine gesamte Wandstärke von 70 mm ergab.

Verbunden wurden die Platten mit einem kälteunempfindlichen, extrem stabilen PU-Schaum-Kleber. Eine zusätzliche Stabilisierung oder Verstärkung der strukturellen Integrität war durch das geringe Gewicht der Nutzlast nicht erforderlich. Nach unseren Schätzungen bleibt die Transport-Box demnach bei bis zu 5-6 kg Last stabil, die tatsächliche Last beträgt jedoch nur 1 – 1,5 kg.

2.2. Ballon und Fallschirm

Der Ballon, der im Versuch verwendet wird ist aus Latex, da Latex besonders gute Ausdehnungseigenschaften aufweist und somit eine hohe Steighöhe verspricht. In gefülltem Zustand ist der Ballon circa 1,9 Meter lang.

Nach dem Platzen des Ballonkörpers durch den geringen Außendruck befindet sich die Nutzlast im freien Fall. Bei einem ungebremsten Aufprall besteht die Gefahr, dass die Hardware beschädigt wird und dass das Ortungssystem ausfällt. Um den Fall zu bremsen und die Wucht des Aufpralles zu mindern, wird ein Fallschirm zwischen Ballon und Transport-Box befestigt.

Im Steigflug ist dieser lose gespannt, sodass er sich nicht entfalten kann. Nach dem Platzen des Ballons spannt sich der Fallschirm dann auf und lässt die Nutzlast sachte zur Erde zurück „fallen“. Für eine möglichst hohe Flugstabilität sorgen eine runde Form und ein Loch im Scheitel des Fallschirms, durch das Luft zur Stabilisierung strömt.

Der Fallschirm besteht aus einem reißfesten, leichten und weichem Gewebe, wodurch er sich beim Fall der Nutzlast leicht öffnen lässt. Der benötigte Mindestdurchmesser des Fallschirms beträgt ca. 125 cm für eine Sinkgeschwindigkeit von 3 – 4 m/s. Für die nötige Spannung zwischen Ballon und Transport-Box sorgen Befestigungsschnüre. Im freien Fall entspannen sich die zuvor tragenden Schnüre und durch den von unten kommenden Luftstrom entfaltet sich der Fallschirm.

2.3. Gewicht

Das Gesamtgewicht muss genau berücksichtigt werden, damit der Ballon die gewünschte Steiggeschwindigkeit und Höhe erreicht. Natürlich gilt, umso mehr Gewicht, desto mehr Helium wird benötigt. Umso mehr Helium aber eingefüllt wird, desto früher wird der Ballon aufgrund des größeren Volumens platzen. Folgendes Gesamtgewicht hat sich für den Wetterballon ergeben:

Transport-Box	800 g
Gnublin-Board	35 g
GPS-Modul	2 g
UMTS-Stick	20 g
Webcam	77 g
Messbrücke PT1000	16 g
Akku	265 g
Kabel / Stecker	30 g
Taschenwärmer	70 g
Fallschirm	150 g
Gesamtgewicht	1465 g

Der Grund für das hohe Gewicht dieses Wetterballons ist in der Vielzahl von unterschiedlicher Hardware sowie in dem hohen Eigengewicht der Transport-Box zu begründen. Daraus ergibt sich ein benötigtes Gasvolumen von ca. 1,5 m³.

2.4. Hardware

2.4.1 Gnublin

Im Rahmen der Vorlesung *Embedded Linux* wurde das Linux-Board *Gnublin* der Firma *Embedded Systems* präsentiert, das in diesem Projekt als Systemplatine verwendet wird.

Das Gnublin-Board ist eine ARM-basierte Ausbildungsplattform für Embedded Linux, die aufgrund seines geringen Stromverbrauchs und den vielen Erweiterungen inklusive einer Programmierumgebung gerne als Ersatz für typische Mikrocontroller-Anwendungen genutzt wird. Damit eignet es sich ideal für unser Projekt, in dem es eine Zahl von Sensoren steuern und als interaktive Schnittstelle für den Wetterballon dienen soll.



Abbildung 3 - Das GnuBLIN-Board

Das Board hat folgende Spezifikationen:

- ARM9 Prozessor mit 180 MHz (LPC3131)
- 32 MB SDRAM
- MicroSD Karten Steckplatz für Bootloader, Kernel, Dateisystem und Swap-Bereich
- USB-Device oder USB-Host Anschluss (per Jumper einstellbar)
- Verschiedene Bootoptionen (SD-Karte, USB oder RS232)
- Ext. Stromversorgung über Anschlussklemme möglich (7-12V)
- Verbrauch: ca. 60-70mA bei Boot + Running / ca. 30-50mA Idle
- 1 x Reset Taster
- 3 x GPIO
- 3 x ADC
- 1 x PWM
- 1 x SPI
- 1 x I2C
- 1 x USB-RS232 Wandler als Konsole für Linux
- 1 x LED steuerbar
- 1 x Power-LED
- 1 x Status-LED

Bei dem Einsatz im Wetterballon werden Akkus mit 7,2 V über die Anschlussklemmen angeschlossen. Am USB-Device-Port werden GPS-Sensor, UMTS-Surfstick und die Webcam angeschlossen.

Am ADC-Anschluss (Analog-Digital-Wandler) wird die Wheatstone'sche Messbrücke mit dem PT1000-Temperatursensor angeschlossen.

2.4.2 Webcam

Um während des Fluges Bilder aufzuzeichnen, wird eine handelsübliche Webcam verwendet. Die Entscheidung fiel auf die Logitech C170, weil von ihr akzeptable Bilder produziert werden und diese Webcam am Gnublin-Board bereits erfolgreich getestet wurde.



Abbildung 4 - Logitech C170

2.4.3 PT1000-Messbrücke

Um die Temperatur außerhalb der Styroporbox zu bestimmen kommt die Wheatstone'sche Messbrücke mit einem PT1000-Tempertursensor zum Einsatz, deren Funktionsweise hier kurz erläutert werden soll. Die Wheatstone'sche Messbrücke besteht im Wesentlichen aus einer Anordnung von vier Widerständen und einem Spannungsmessgerät.

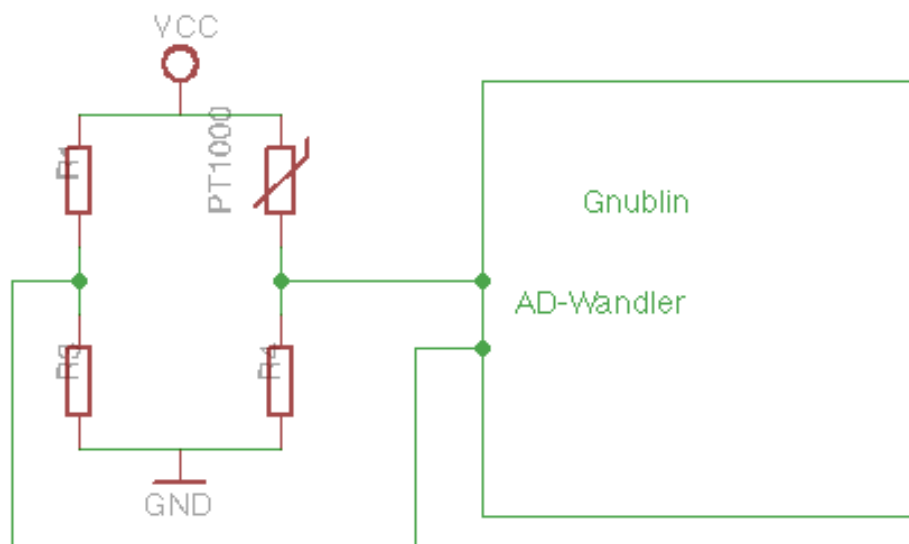


Abbildung 5 - Wheatstonesche Messbrücke mit Anschluss ans Gnublin-Board

Die Widerstände R1, R3, R4 sowie der PT1000-Temperatursensor lassen die angelegte Spannung abfallen. Die Widerstände haben 1 k Ω , was dem Widerstand des Temperatursensors bei 0° Celsius entspricht. Der Spannungsabfall beim PT1000 ist von der Temperatur abhängig. Je niedriger die Temperatur wird, desto geringer wird der Widerstand. Die Spannungen beider Seiten werden dann verglichen und im GnuBlin-Board über einen Analog-Digital-Wandler in einen Digitalenwert umgerechnet. Der digitale Wert kann dann durch Software in eine „Temperatur“ umgewandelt werden.

2.4.4 UMTS-Stick

Ein UMTS-Stick baut eine Verbindung zum Mobilfunknetz auf und bietet dadurch Zugriff auf Kommunikationsfunktionen wie UMTS-Datennetz, SMS, MMS oder Telefonie. Für die Kommunikation mit dem GnuBlin-Board wird der UMTS-SurfStick ZTE MF190 verwendet. Dieser kann nach einigen kleineren Anpassungen am Linux-Kernel am GnuBlin-Board verwendet werden. Der Betrieb des UMTS-Moduls benötigt besonders viel Energie, weshalb von dem Stick keine Datenverbindung aufgebaut werden soll. Stattdessen findet die Kommunikation über periodisch abzurufende und zu sendende SMS-Kurznachrichten statt.



Abbildung 6 - ZTE MF190 UMTS-Stick

Der Stick benötigt eine SIM-Karte für den Betrieb. Dafür wurde eine günstige Prepaid-Karte für 10 Euro mit 3000 Frei-SMS angeschafft.

2.4.5 GPS-Empfänger

Für die Ermittlung der Position wird ein GPS-Empfänger benötigt. Dieser sollte möglichst günstig, leicht und Stromsparend sein. Daher fiel die Wahl auf den Empfänger Gms-D1 von GlobalTop mit MediaTek MT3329 Chip.

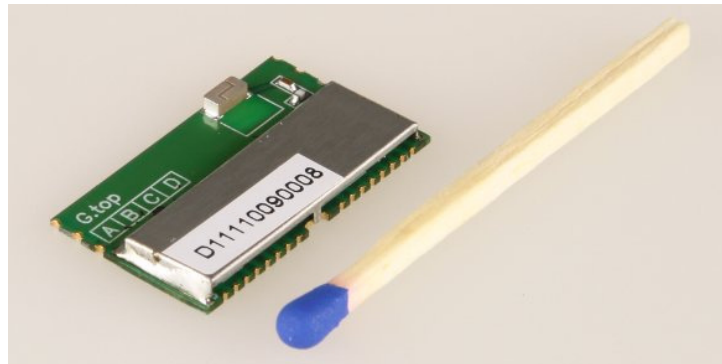


Abbildung 7 –GlobalTop Gms-D1 MT3329

Stromverbrauch: 37 – 48 mA

Gewicht: 2 g

2.5. Stromversorgung

Die Stromversorgung stellt eine besonders kritische Komponente dar, von welcher sämtliche Systeme abhängen. Zudem reduziert die extreme Kälte in der Stratosphäre von ca. -50° Celsius enorm die Akkulaufzeit. Es wird eine Reduzierung auf schlimmstenfalls 30 – 50 Prozent der Kapazität erwartet. Deshalb ist es ratsam, den Akku zu überdimensionieren.

Es kommen verschiedene Akku-Technologien in Frage: Nickel-Metallhydrid (NiMH), Lithium-Ionen oder Lithium-Polymer.

Generell lässt sich sagen, dass NiMH-Akkus kostengünstiger, dafür aber deutlich schwerer und kälteempfindlicher als Lithium-Ionen-Akkus sind. Zudem haben Lithium-Ionen-Akkus nur eine geringe Selbstentladung.

Da Lithium-Polymer-Akkus eine etwas höhere Energiedichte als herkömmliche Lithium-Ionen-Akkus aufweisen, fiel die Wahl auf diese.

Zur Bestimmung der Akku-Kapazität war es erforderlich, den Stromverbrauch der Geräte zu bestimmen:

Gerät	Stromverbrauch in mA
Gnublin	~ 50
GPS-Modul	~ 50
UMTS-Stick	~ 150
Kamera	~ 200
Temperatursensor und Messbrücke PT1000	~ 0,6
Gesamt	~ 400 mA

Es muss also mit einem maximalen Verbrauch von 400 mA gerechnet werden.

Der Akku muss des Weiteren die benötigte Leistung über die gesamte Flugdauer hinweg bereitstellen. Die Flugdauer ist von vielen Faktoren abhängig und daher unmöglich vorher zu sagen. Anhand der Erfahrungen anderer Wetterballon-Teams können wir die Flugdauer auf eine Zeit zwischen zwei und sechs Stunden eingrenzen.

Geht man von einem Verbrauch von konstant 400 mA und einer Flugdauer von 6 Stunden aus, ergibt sich folgende Rechnung:

$$\text{Kapazität} = 400 \text{ mA} * 6 \text{ h} = 2400 \text{ mAh}$$

Demnach müsste ein Lithium-Polymer-Akku mit mindestens 2400 mAh Kapazität eingesetzt werden. Da aufgrund der Kälte mit einem Verlust von 50 - 70 Prozent der Kapazität zu erwarten ist, sollte der Akku demnach eine Kapazität von 4800 mAh – 8000 mAh aufweisen.

Aus Gewichts- und Kostengründen haben wir uns für einen Lithium-Polymer-Akku mit einer Kapazität von 5000 mAh entschieden.

2.6. Software

2.6.1 Skript für den Autostart der Software

Im Verzeichnis `/dev/init.d/` wird beim Booten des Boards ein Skript ausgeführt, das alle wichtigen Programme für das Projekt startet. Im Start-Block des Skripts werden der Watchdog-Timer, die Anwendung zur Messung und Speicherung der Temperatur, die Anwendung zur Aufnahme von Fotos und Videos sowie die Anwendung für die Kommunikation und Ortung gestartet.

Da diese Funktionen essentiell für den Betrieb und die Wiederauffindung der Konstruktion sind, gelten hier ähnliche Anforderungen an die Betriebsstabilität der Software wie in der Luftfahrt. Die Software muss unter allen Umständen die erforderlichen Funktionen ausführen können und sich ohne Außeneinwirkung im Fehlerfall selbst zurücksetzen oder Fehlerbehandlungen durchführen können. Daher sorgt ein ebenfalls automatisch gestartetes Script dafür, dass ein unvorhergesehen beendetes Script sofort wieder gestartet wird. Sämtlicher Code aller Scripts ist mit Fehlerbehandlungsmaßnahmen versehen.

2.6.2 Watchdog-Timer

Der Watchdog-Timer (WDT) ist eine Systemfunktion des LPC3131-Prozessors. Diese kann softwareseitig gestartet werden und wartet dann in einem 10 Sekunden-Intervall auf ein Signal. Bleibt dieses aus, wird ein hard-reset des Gnublin-Boardes vorgenommen. Auf diese Weise soll ein permanentes Einfrieren beziehungsweise Abstürzen des Betriebssystems verhindert werden.

Um den internen Watchdog zu aktivieren wurde ein kleines C-Programm geschrieben, welches den WDT-Treiber anspricht. Der Watchdog präsentiert sich als Gerätedatei, die mit libc-Funktionen geöffnet und geschrieben werden kann.

```
int wdt_file = open("/dev/watchdog", O_WRONLY);
```

Falls der Watchdog nicht gestartet, die Datei also nicht geöffnet, werden kann, wird eine Fehlermeldung per SMS verschickt.

Wurde der Watchdog erfolgreich gestartet, wird er alle zehn Sekunden durch ein Script mit einem „Lebenszeichen“ versorgt.

```
ioctl(wdt_file, WDIOC_KEEPLIVE, 0);
```

Bleibt diese Benachrichtigung aus, wird das Gnublin-Board neugestartet. Beim Beenden des Watchdog wird die Gerätedatei wieder geschlossen und das Programm beendet.

2.6.3 Trackingsystem

Um die Bewegungen des Ballons verfolgen zu können, wird der UMTS-Stick zusammen mit dem GPS-Empfänger als Trackingsystem verwendet. Über die USB-Schnittstelle wird vom GPS-Empfänger periodisch die Position abgerufen. Diese Koordinaten werden über den UMTS-Stick per SMS an uns versendet. Natürlich setzt das den Empfang im Handynetzt voraus. Ein flächendeckender Empfang ist in dieser Region gegeben und bis zu einer Höhe von circa 1000 Meter möglich. Während des Großteils des Flugs werden wir also keine Positionsdaten erhalten, diese werden aber aufgezeichnet für die spätere Auswertung.

Besonders wichtig ist das Trackingsystem für die Zeit im Empfangsbereich vor der Landung und nach der Landung. Vor der Landung kann dadurch der ungefähre Aufschlagsort ermittelt werden. Wenn das System nach der Landung noch funktioniert, wird die Position gesendet, bis der Akku erschöpft ist.

Ersatzweise für die GPS-Koordinaten können vom UMTS-Stick per SMS auch Netzwerkinformationen, die die aktuelle Funkzellen-ID und Netzwerk-Code bei Empfang übermittelt werden. Diese Codes können

über die Internetplattform *OpenCellID*¹ in grobe Koordinaten umgewandelt werden.

Per SMS können auch Befehle an das Gnublin-Board geschickt werden. Ein Python-Script wertet diese Befehle aus und setzt sie in Kommandos um. Hier eine Funktionsübersicht:

- Setzen der Systemzeit (Zeit wird per SMS als String übergeben)
- Abrufen der Systemzeit
- Starten der Übermittlung von GPS-Koordinaten (mit Intervallangabe)
- Beenden der Übermittlung von GPS-Koordinaten
- Netzwerkinformationen übermitteln
- System neu starten
- Beliebigen Systembefehl ausführen (Befehl als String aus SMS)

Dieses Script `listener.py` wird in `/root` platziert und mittels `chmod +sx` ausführbar gemacht und mit Root-Rechten versehen. Um gewisse Systembefehle ausführen zu können, sind volle Privilegien erforderlich. Sicherheitsaspekte können hier vernachlässigt werden. Das Script wird beim Systemstart automatisch ausgeführt.

Es wird die Software *gammu* verwendet, welche eine Abstraktionsschicht für den Zugriff auf verschiedene mobile Geräte und eine Python-API zur Verfügung stellt. Diese vermeidet das Arbeiten mit AT-Kommandos und stellt komfortable Befehle z.B. zum Versenden von SMS zur Verfügung.

2.6.4 Temperaturanwendung

Um die Temperatur während des Flugs zu messen, wird eine Messbrücke mit einem PT1000-Temperatursensor an den Analog-Digital-Eingang GPA1 des Gnublin-Boards angeschlossen.

¹ <http://www.opencellid.org>

Der Analog-Digital-Wandler wird über eine Gerätedatei geöffnet und mit dem entsprechenden Kanal beschrieben. In diesem Fall wurde der Kanal 1 gewählt.

```
os.write(adc_file, "0x0001")
```

Es werden immer 256 Byte von GPA1 gelesen. Als Wert erhält man einen hexadezimalen String, der die Differenz der Spannung der Messbrücke angibt. Diese Differenz wird dann in eine dezimale Temperatur umgerechnet und in eine Temperatur-Messdatei geschrieben.

Die Temperaturanwendung kann per SMS gesteuert werden.

2.6.5 Webcam-Aufzeichnung

Die verwendete Webcam Logitech C170 wird vom System bereits unterstützt, daher entfallen aufwendige Konfigurationen. Es ist im System das Programm *uvccapture* enthalten, mit dem Webcams angesprochen werden können. Es soll in einem Intervall, etwa alle 2 Minuten, ein Bild aufgenommen werden. Diese Bilder werden im JPEG-Format auf der lokalen SD-Karte abgespeichert und werden nach der Bergung ausgewertet. Bei ausreichendem Bildmaterial könnte aus den Einzelbildern ein Videofilm geschnitten werden.

2.7. Testphase und Vorbereitung

Vor dem Start wurden alle Bauteile und die Software gründlich getestet um ein mögliches Fehlverhalten auszuschließen.

2.7.1 Test der Wärmeisolation und Taschenwärmer

Um die Güte der Isolation zu testen, wurde die Styroporbox mit dem Gnublin-Board und dem Temperaturmodule von *Embedded Projects* testweise in der Gefriertruhe eingefroren. Die tiefste Temperatur der Gefriertruhe im Normalbetrieb liegt bei -20° Celsius.

Beim ersten Versuch wurde der Nullpunkt in der Styroporbox nach circa einer Stunde erreicht. Dieser Wert musste sich deutlich verbessern, deshalb fügten wir der Box einen chemische Taschenwärmer und isolierendes Material aus Maisstärke bei.

Der chemische Taschenwärmer sollte 12 – 20 Stunden lang eine Temperatur von 60° Celsius erzeugen. Es handelte sich um einen auf Eisenoxid basierenden, exothermen Taschenwärmer der Firma *The Heat Company*.

Der zweite Gefriertruhentest verlief deutlich besser. Bei -20° Celsius wurde der Nullpunkt innerhalb der Schachtel erst nach über drei Stunden erreicht.

Tue Jul 18 19:57:34 UTC 2013
25.500 C

Tue Jul 18 20:00:57 UTC 2013
27.250 C

Tue Jul 18 20:04:17 UTC 2013
28.000 C

Tue Jul 18 20:07:42 UTC 2013
27.500 C

...

Tue Jul 18 23:23:59 UTC 2013
-2.750 C

...

Zunächst stieg die Temperatur durch die Taschenwärmer an, aber schon bald fiel die Temperatur konstant. Nach über drei Stunden wurde der Test beendet.

2.7.2 Test der Temperatur-Messeinheit

Um die Temperatur-Messeinheit zu testen, wurde die Spannung der Wheatstone'sche Messbrücke mit dem PT1000-Temperatursensor durchgemessen.

Damit die Messeinheit halbwegs unter realen Bedingungen getestet werden kann, wurde sie in der Gefriertruhe mit dem GnuBlin-Board eingefroren.

2.7.3 Test der Webcam

Da der Treiber für die Webcam bereits im System vorhanden war, musste lediglich das Modul geladen werden.

```
modprobe uvcvideo
```

Danach konnte mit einem Kommandozeilen-Befehl ein Bild von der Webcam gespeichert werden.

```
uvccapture -oBILD.jpg -m -x640 -y480
```

Dieser Befehl wird in einem Script in einer Schleife in Intervallen wiederholt aufgerufen. Dies erzeugt viele Einzelbilder, die nach der Landung und Bergung ausgelesen werden können.

2.7.4 Test des Trackingsystems

Das System wurde mit angeschlossenem UMTS-Stick gestartet. Da das Steuerungsscript automatisch gestartet wird, war das Gerät sofort empfangsbereit, was es durch eine SMS-Statusmeldung „*Device is up and listening*“ signalisierte. Es wurde jeder mögliche Befehl ausprobiert, woraufhin das Gerät wie erwartet antwortete.

Um den Betrieb bei niedrigen Temperaturen zu testen, wurde der Stick im Betrieb mit einer Plastikhülle geschützt und mit Eisspray auf bis zu -40° Celsius vereist. Das periodische Senden von SMS wurde dabei kurzzeitig unterbrochen, allerdings nach dem „auftauen“ sofort wieder ohne SMS-Verluste fortgesetzt.

2.8. Durchführung

Der Start des Ballons wird am 02. August 2013 um 08:00 CET in Druisheim (86690) stattfinden. Die genauen Koordinaten des Startorts lauten:

- 48,64 48° 38' 26,62'' Länge
- 10,81 10° 48' 45,96'' Breite

Die geografische Höhe beträgt 436 Meter über dem Meeresspiegel.

2.9. Materialkosten

Es fielen für das Projekt folgende Materialkosten an:

Gnublin	49,- Euro
Speicherkarte	10,- Euro
GPS-Modul	15,- Euro
UMTS-Stick	15,- Euro
SIM-Karte	10,- Euro
Webcam	20,- Euro
Messbrücke PT1000	15,- Euro
Akku	35,- Euro
Wetterballon	50,- Euro
Fallschirm	15,- Euro
Taschenwärmer	5,- Euro
Kabel / Adapter	10,- Euro
Helium	0,- Euro (gesponsert)
Gesamt	249,- Euro

3 Fazit

Die Planung und Organisation dieses Projektes war für uns eine besonders spannende Herausforderung. Es ist immer wieder eine Freude, interdisziplinär zu arbeiten. Wir erwarten den Starttermin mit Zuversicht.

4 Anhang

4.1. Literaturverzeichnis

Wikipedia. (kein Datum). *Wikipedia*. Abgerufen am 15. 07 2013 von Wikipedia: <http://de.wikipedia.org/wiki/Erdatmosphäre>

Wolfgang, D. *Experimentalphysik 1 - Mechanik und Wärme* (Bd. 1). Springer Spektrum.

4.2. Abbildungsverzeichnis

Abbildung 1 - Schematische Darstellung der Atmosphärenschichten mit Angaben der Grenzhöhen aus Wikipedia Atmosphärenschichtung [WIKISTD].....	6
Abbildung 2 - 2D-CAD-Zeichnung der Styroporbox	9
Abbildung 3 - Das Gnublin-Board	12
Abbildung 4 - Logitech C170	13
Abbildung 5 - Wheatstonesche Messbrücke mit Anschluss ans Gnublin-Board	13
Abbildung 6 - ZTE MF190 UMTS-Stick.....	14
Abbildung 7 –GlobalTop Gms-D1 MT3329	15

4.3. Abkürzungsverzeichnis

UV-Licht	Ultra-Violettes-Licht
GPS	Global Positioning System
MHz	Megahertz
CET	Central European Time
CAD	Computer Aided Design
UMTS	Universal Mobile Telecommunications System
PU	Polyurethan
ns	Nanosekunden
ARM	Advanced RISC Machines
mA	Milliampere
RISC	Reduced Instruction Set Computer
MB	MegaByte (1.048.576 Byte)
SDRAM	Synchronous Dynamic Random Access Memory
USB	Universal Serial Bus
V	Volt
GPIO	General Purpose Input/Output
ADC	Analog-to-Digital-Converter
PWM	Pulsweitenmodulation
SPI	Serial Peripheral Interface
I ² C	Inter-Integrated Circuit
LED	Light emitting diode
NiMH	Nickel-Metallhydrit
mAh	Milliampere-Stunden
SMS	Short Message Service

4.4. Quellcode

Der für das Projekt von uns erstellte Programmcode ist online auf Github zu finden:

<https://github.com/ramteid/gnublin-weatherballoon>

Erläuterung der Dateien:

<code>local.autostart</code>	<code>/etc/init.d/</code> Startet alle Prozesse
<code>wdt.c</code>	<code>/root/</code> Initialisiert den Hardware-Watchdog
<code>listener.py</code>	<code>/root/</code> Initialisiert den SMS-Listener
<code>listener_start.py</code>	<code>/root/</code> SMS-Listener, verarbeitet GPS-Daten und Kamerabilder
<code>script_watcher.sh</code>	<code>/root/</code> Hält die Prozesse am Laufen
<code>temperature.py</code>	<code>/root/</code> Liest den Temperatursensor aus
<code>_installation.sh</code>	<code>/root/</code> Kopiert die einzelnen Dateien und setzt Rechte

4.5. Erklärung

Hiermit erklären wir, dass wir die vorliegende Arbeit selbständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden.