# Semiautonomes Fernerkundungssystem eine Jugend forscht-Arbeit von

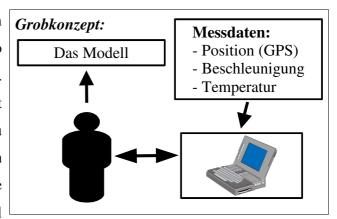
## Kai Rosebrock und Marco Möller

## Inhaltsverzeichnis

1	Die Idee	2
2	Die Hardware	2
2.1	Das Flugzeugmodell	3
2.2	Die Fernsteuerung	4
2.3	Pic-Progammer	4
2.4	Telemetriesender im Flugzeug	4
2.5	Telemetrieempfänger am Boden	6
2.6	ATV-Übertragung	7
3	Software	7
3.1	Die PIC Software im Flugzeug.	7
3.2	Die Schnittstelle	8
3.3	Das Kartenmodul	10
3.4	Die Messdatenauswertung.	11
3.5	Die Steuerung	12
4	Die Einsatzgebiete	13
4.1	Fernerkundung	13
4.2	Unterstützung von Piloten	13
4.3	Experimental plattform	14
5	Das Ergebnis	14
6	Die Hilfen	15
7	Verwendete Literatur	15

#### 1 Die Idee

Die Idee, ein autonomes System zu entwickeln, schwebte Marco Möller schon des längeren vor. Zuerst war angedacht, dies mit Hilfe eines U-Boot Modelles zu realisieren. Aufgrund von verschiedenen Problemen wurde diese Idee allerdings schnell



wieder verworfen. Da es schon fast fertige Baukästen für Flugzeugmodelle gibt und Kai Rosebrock Flugzeugmodellbau als Hobby betreibt, schien dies als Basis die bessere Wahl.

Wir stellten uns vor, eine Art GPS unterstützten Autopiloten zu entwickeln. Somit könnte das Flugzeug selbstständig eine vorher festgelegte Route abfliegen. Desweiteren sollten dabei Messwerte der Umgebung, wie z.B. Höhe über Grund, Temperatur, Luftfeuchtigkeit, etc. aufgenommen werden, um sie später am PC auszuwerten.

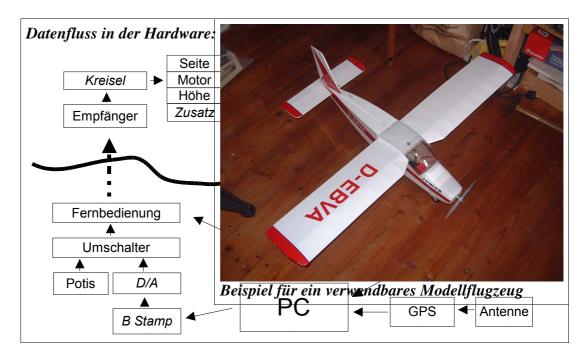
Dieses hatten wir das letzte Jahr bereits größtenteils realisiert. In diesem Jahr wurde die Software noch einmal erweitert und verfeinert, und auch die Hardwareplattform komplett umgestellt (von Basic Stamp auf PIC). In unseren Vorstellungen haben wir zugegebenermaßen von dem Plan des komplett autonomen Fliegen Abstand genommen, und unterstützen stattdessen den Piloten lieber beim manuellen Fliegen. Hierfür haben wir viele neue Möglichkeiten entdeckt und realisiert.

#### 2 Die Hardware

Die Hardware des Semiautonomen Fernerkundungssystem besteht aus verschiedenen Einzelkomponenten, die später noch im Einzelnen erläutert werden. Bei diesen kam es uns vor allem auf Funktionalität und im Flugzeug auch auf Gewichts- und Stromverbrauch an.

## 2.1 Das Flugzeugmodell

Diesmal haben wir auf ein eigenes Modellflugzeug verzichtet, und dafür auf die Modelle eines erfahrenen Modellfliegers, Frank Budnik, zurückgegriffen. Hierdurch



hatten wir auch gleich die Möglichkeit einen potentiellen Kunden zum Erfahrungen sammeln mit unserem System auszustatten. Er konnte uns so wertvolle Tipps für weitere Verbesserungen geben. Weiterhin ergab sich der Vorteil, das wir uns nichtmehr um das Modell kümmern mussten, und somit viel mehr Zeit für unserere eigentliche Zielsetzung übrig blieb.

## 2.2 Die Fernsteuerung

Da wir dieses Jahr das System wieder manuell steuern lassen, erübrigt sich der Einsatz unseres PC -> Fernbedienungsadapters. Diese Arbeit ist allerdings nicht komplett vergebens, da diese, wenn wir in Zukunft die Funktionalität vom autonomen Fliegen

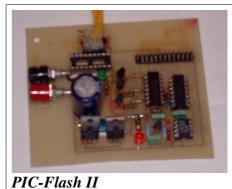


wiederbeleben, dann bereits vorhanden ist. Weiter wäre denkbar das Flugzeug hiermit über einen PC-Joystick fliegen zu können. Dadurch wären weitere "Sensorkanäle", nämlich die drei Steuerleitungen, für die Auswertung und Analyse verfügbar.

## 2.3 Pic-Progammer

Es sollte allerdings nicht unerwähnt bleiben, das wir dieses Jahr durch die Umstellung von Basic Stamp auf PIC, wie im nächsten Kapitel beschreiben, ein Gerät zum beschreiben dieser IC's brauchten. Da unser Budget beschränkt war, haben

wir hier nicht auf eine kommerzielle Lösung gesetzt, sondern auf den Nachbau eines im Internet (www.uc-project.de) vorgestellten Programmers, dem PIC-Flash II. Zwar ist der Nachbau ansich nichts besonderes, hat aber eine Menge an Arbeit gekostet, vorallem was die Fehlersuche anging. Hierbei hatte sich durch einen 74er IC aus der falschen Reihe (HS anstatt



LS) ein eigentlich offensichtlicher aber trotzdem sehr schwer zu findender Fehler eingeschlichen.

#### 2.4 Telemetriesender im Flugzeug

Der Kern unseres Projekts ist das Sammeln und Auswerten von Sensordaten. Um die Daten zu sammeln haben wir eine Sensoreinheit konzipiert. Diese sammelt momentan GPS – Position, Beschleunigungswerte, Helligkeitsdaten und Temperatur. Zusammen werden sie 3 mal pro Sekunde zur Auswertung zu Boden gefunkt. Das ganze ist aber so konzipiert, das es auch Befehle vom Boden in Empfang nehmen kann, und weitere Sensoren jederzeit nachrüstbar sind.

Zum Navigieren benötigen wir GPS, weshalb es eine sehr wichtige Stellung in unserem Projekt einnimmt. Die Integration in ein Modellflugzeug war hauptsächlich ein Gewichtsproblem. Alleine die GPS Antenne wiegt schon 120 g. Nach dem Zerlegen einer Standardfahrzeugantenne (Bosch) fiel uns auf, dass sie aus einem massivem Metallblock

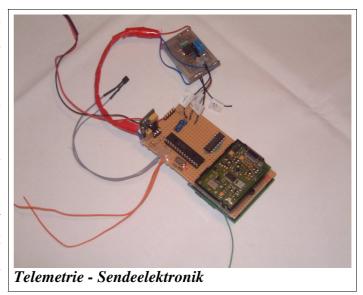


und einer kleinen Platine mit Keramikempfänger besteht. Zudem ist die ca. 1,5 m lange Anschlussleitung ebenso schwer. Daraufhin ersetzten wir einfach den Metallblock durch Alufolie und kürzten die Antennenleitung auf 30 cm. Die Stabilität ist trotzdem sichergestellt, da alles mit Heißkleber vergossen wurde. Zudem verzichteten wir auf den HF-Stecker und löteten die Antennenleitung direkt an den Empfänger.

Die Sensoren für Helligkeit und Temperatur bestehen aus einem LDR (Light Detecting Resistor) bzw. KTY (Silizium Themperaturfühler). Dessen aktuelle Werte ermitteln wir, in dem wir die Teilspannung abgreifen, die über ihnen in einer Reihenschaltung anfällt. Diese wird nun von unserem Prozessor mit 10 Bit Seite 4 von 15

Genauigkeit digitalisiert (interne A/D Wandler – successive Approximation). Mit Hilfe dieser Methode ließen sich noch weitere Sensoren realisieren (vorerst bis zu 5 Kanäle), was den Charakter einer Plattform zur Messwerterfassung stärkt. Der komplette Aufbau wurde auf ein Gewicht von ca. 115 g reduziert, was zwar mehr als letztes Jahr, aber trotzdem noch ausreichend leicht ist.

All diese Daten werden nun von den **PIC** 16F876 gesammelt und komprimiert. Anschließend werden sie an das Funkmodem übermittelt, welches das Modulieren und Senden der Daten übernimmt. Es ist ein FTR-700 der Firma PICIORGROS. Mit nur 25 g Gewicht ist dieses Modem optimal für ein



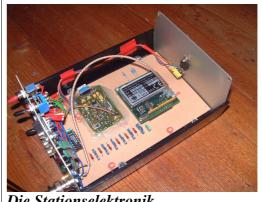
Modellflugzeug geeignet. Die 64 Kanäle im Bereich von 433,100 MHz bis 433,750 MHz gewährleisten uns, einen ungestörten Funkbetrieb in diesem stark genutzten Frequenzbereich (ISM – 70 cm). Mit der Sendeleistung von 10 mW erreichen wir eine Reichweite von ca. 500 m was vollkommen genügt, da die Fernbedienung einen ähnlichen Bereich abdeckt. Die Datenrate ist mit 1200 bzw. 2400 bps für unsere Zwecke ebenfalls vollkommen ausreichend, da wir nur 32 Byte pro Paket senden wollen. Diese Pakete werden ca. 3 mal pro Sekunde gesendet, was uns eine Auflösung im Beschleunigungsverlauf bis auf 1,5 Hz erlaubt. Bei der Spannungsversorgung an Bord des Flugzeugmodells mussten wir aus Gewichts- und Akkuhaltbarkeitsgründen ein Schaltnetzteil verwenden. Das von uns integrierte liefert 1 A bei 5 V und hat einem Wirkungsgrad von ~ 95 %, aber dennoch nur einem Flächenbedarf von 2 cm². Somit ist es ebenfalls optimal für den Einsatz im Modellflugzeug geeignet, bei dem es auf Gewicht, Größe und geringen Stromverbrauch ankommt.

Die gesamte Flugzeugelektronik funktionierte – wider unseren Befürchtungen – beim ersten Versuch. Nur für das "Umbauen" der GPS – Antenne waren mehrere versuche notwendig. Das einzige Problem bei der Flugzeugelektronik war das Warten auf die

geätzte Platine für unser Schaltnetzteil, da beim Layouten einige Fehler unsererseits mit der Eagle Bedienung (Layoutsoftware für Platinen) entstanden.

#### 2.5 Telemetrieempfänger am Boden

In der Station werden alle Daten des Modellflugzeugs empfangen und an den PC weitergeleitet. Hier wird erneut das FTR-700 Funkmodem genutzt, dessen Daten mit Hilfe eines Max-232 vom TTL ins RS-232 Format konvertiert werden. Die weitere Auswertung übernimmt dann der PC.



Die Stationselektronik

Um die GPS Ungenauigkeit von ca. 100 m in der Breite, und 150 m in der Höhe zu verkleinern, nutzen wir unser eigenes D-GPS (Differential - Global Positioning System). Dazu haben wir einen zweiten GPS-Empfänger am Boden aufgebaut, von welchen wir wissen, dass er sich nicht bewegt. Sobald dieses jedoch geschieht, handelt es sich um künstliche Verfälschungen durch GPS. Diese sind im Flugzeug sehr ähnlich. Dadurch können wir dies bei der Position des Flugzeugs heraus rechnen, was uns zu einer Genauigkeit von +/- 20 m verhilft. Um die Höhe über Grund grob zu erhalten, können wir bei ebenen Gelände den Stationsstandort als Nullpunkt definieren. Die Differenz zwischen unserer Höhe und der Höhe des Flugzeuges ist also die Höhe über Grund.

Die Station wird über 2 RS-232 Anschlüsse mit dem PC verbunden. Ein Kanal für GPS Daten mit 4800 Bps (wahlweise JRC / NMEA Protokoll), und einen zweiten mit 9600 Bps, der die empfangenen Funkdaten ausgibt und wahlweise Daten aussendet bzw. zur Fernbedienung durchschweift. Durch alle diese Möglichkeiten kommen die vielen Anschlüsse und Schalter auf der Front zustande.

## 2.6 ATV-Übertragung

Als Ideale Ergänzung zu der Navigationshilfe für den Modellpiloten bietet sich eine Videoübertragung an. Mit der Hilfe eines Videobildes ist der Pilot in der Lage auch außerhalb seiner Sichtreichweite zu fliegen. Durch das Bild erhält er Informationen über die Umgebung, sowie zu der aktuellen Fluglage des Flugzeuges.

Zur Zeit sind wir dabei, diese Videoübertragung mit Hilfe der ATV - Technik (Amateur Television) zu realisieren. Die Übertragungstechnik (ATV-Sendemodul) Seite 6 von 15 haben wir aus einem Artikel der Zeitschrift TV-Amateur entnommen. Empfangen wird das Videobild mit Hilfe eines alten SAT-Recievers (ohne Umbau!).

#### 3 Software

Die Software hat hinterher die Aufgabe, die Daten der verschiedenen Quellen zu entschlüsseln und für den Benutzer leicht verständlich zu präsentieren. Zudem müssen Steueranweisungen für das Flugzeug aus der aktuellen Situation sowie dem Benutzerauftrag generiert werden.

Es wurde durchgängig versucht die Oberfläche vom Code abzugrenzen. Dies ist unter Visual Basic, in dem dieses Projekt erstellt wurde, gar nicht so einfach. Um die Modularität nicht nur optisch vorzuhalten, haben wir die einzelnen Formulare nur durch ein oder zwei Funktionen mit dem restlichen Programm kommunizieren lassen. Zudem wird diese Kommunikation durch das Hauptmodul geschleust. Hierdurch entsteht der Vorteil, das die Formulare bei Bedarf getauscht werden könnten, ohne das der Code an allen Stellen angepasst werden müsste. Dies wäre dann nur im Hauptmodul erforderlich.

Der Code auf dem PC ist aber nicht der Einzige bei uns. Um die Messungen und Verarbeitungsaufgaben im Flugzeug durchzuführen, wird auch dort Quellcode benötigt, der von einem PIC ausgeführt wird.

## 3.1 Die PIC Software im Flugzeug

In dem Flugzeug arbeitet eine PIC 16F876, um die Werte der Sensoren zu digitalisieren, die GPS – Daten zu filtern und um anschließend alles zu übertragen.

Da der Pic nur über eine serielle Schnittstelle verfügt (1 x TX / 1 x RX), wir aber 2 (2 x TX / 1 x RX) benötigen, mussten wir uns dafür einen Weg suchen. Dies erledigt ein MAX 3100 der von SPI auf USART (RS 232) wandeln kann. So sind wir in der Lage ein dutzend Schnittstellen am PIC zu erweitern, da die SPI Schnittstelle ein Bus ist.

Dies ist unser erster Gehversuch in der Programmierung in C und auch gleichzeitig im Einsatz von PIC's, was es nicht so sehr trivial machte. Teils scheiterte es an einigen Macken die der sehr alte Compiler (1995) mit sich brachte, da er durch unseren aktuellen PIC - Typ mit seinem großem RAM- und ROM-Ausbau sehr durcheinander gebracht wurde. Zum Beispiel ließen sich die Applikationen aus dem

Handbuch in Bezug auf Array's vom Type Long nicht nachvollziehen. So mussten wir hier zugegebenermaßen einige Umwege in Kauf nehmen.

Der Funkstring der nun Ausgesendet wird enthält verschiedene Daten. Die GPS Position wird direkt aus dem JRC Protokoll entnommen. Der komplette Datensatz wird so ca. 3 mal pro Sekunde gesendet.

Die Beschleunigungswerte für die X, Y und Z Achse erhalten wir in der Form eines PW Modulierten Rechteckes mit einer Frequenz von ca. 80 Hz. Dabei entspricht ein Tastgrad von 0,5 einem beschleunigungslosem Zustand. Diese Rechtecksignale geben wir auf Interrupt on Change Pins von Port B des PIC's. Der PIC hat einen Internen 16 Bit Counter, dessen Wert er bei jeder Änderung der Pins abspeichert wird. Durch diese Technik können wir das PWM Signal ungefähr mit einer Genauigkeit von 16 Bit abtasten (1 Periode entspricht ca. 56 000 Zähleinheiten). Durch diese Methode erhalten wir pro Kanal 4 Byte an Informationen (2 Byte Periodendauer und 2 Byte Pulsdauer). Diese 12 Byte an Informationen werden ebenfalls per Funk versandt.

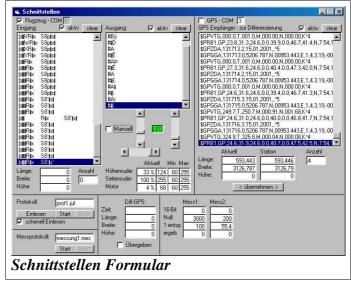
Das Digitalisieren der anderen analogen Sensoren erfolgt per interner A/D Wandler mit 10 Bit Genauigkeit. Hierunter fällt z.B. auch die Betriebsspannung die wir vorher auf ein Drittel herunter geteilt haben, um sie unter 5 V zu drücken, damit sie messbar wird.

#### 3.2 Die Schnittstelle

Die Schnittstelle ist als eigenes Formular ausgelegt und somit vom weiteren Code weitestgehend abgekapselt. Sie beinhaltet 3 Kommunikationskanäle. Zum Einen wäre da die Schnittstelle zum GPS Modul. Dieser Programmteil hat die Aufgabe, die Daten nach Strings zu zerlegen. Zudem werden die wesentlichen Informationen wie Länge, Breite, Höhe und Zeit herausgezogen. Es muss allerdings darauf geachtet werden, dass die Strings komplett und unverstümmelt sind.

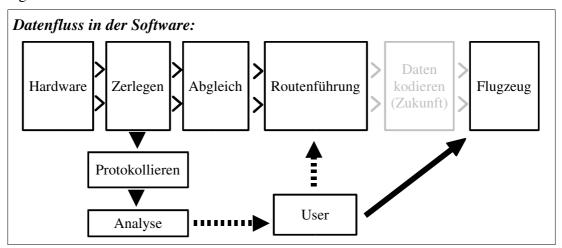
Der zweite Kanal wäre der Eingang der Telemetriedaten vom Flugzeug. Dieser muss ebenfalls als Erstes die Strings zerlegen. Nun werden diese dekomprimiert und den einzelnen Variablen zugeordnet. Die Variabelen werden anschließend in die wirklichen Physikalischen Größen zurückgerechnet. Ein Beispiel hierfür ist, dass zwei 16 Bit Zahlen in Beschleunigung in g zurückgerechnet werden.

Der dritte Kanal ist ein Sender, welcher die Steueranweisungen für das Flugzeug generiert. Hierbei müssen die Werte skaliert werden. In diesen Formular ist eine Möglichkeit geschaffen, den Nullpunkt zentral abzugleichen. Die eigentliche Steuerroutine muss dann nur einen Prozentwert noch



übergeben, der hier in Befehle für den D/A Wandler (noch über Basic Stamp) umgesetzt wird.

Außer dem reinen Empfangen von Daten ist auch noch eine Aufzeichnungs- und Wiedergabefunktion für die Schnittstellen integriert. Diese ist in der Lage, die Werte mit einer zeitlichen Genauigkeit von ca. 10 ms wiederzugeben. Die Daten werden dann direkt hinter der Stringzuordnung eingespeist, wo sie auch vorher entnommen wurden. Somit ist es für das restliche Programm nicht ersichtlich, dass die Daten eigentlich nur simuliert sind.

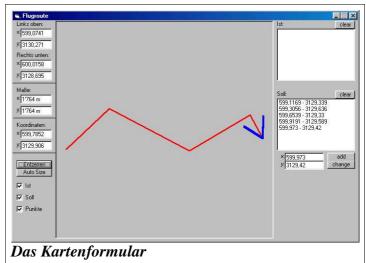


Um die GPS-Genauigkeit zu erhöhen, verwenden wir, wie unter 2.4 beschrieben, eine eigene Version des D-GPS. Damit die restliche Software nur noch die bereinigten Werte sieht, werden die Berechnungen hierzu ebenfalls in diesem Formular durchgeführt.

#### 3.3 Das Kartenmodul

In diesem Teil des Programms wird dem Benutzer als zentrales Steuerelement eine Picture Box in Form einer Landkarte zur Verfügung gestellt. In dieser Picture Box wird die zurückgelegte Flugstrecke angezeigt. Die Anzeige kann hierbei beliebig

verzerrt und gezoomt werden. Es gibt hierbei 2 automatische Zoomstufen. Bei "Auto-Size" wird der Bildausschnitt so gewählt, dass alle benutzten Positionen sichtbar sind. Zudem gibt es die Stufe "Entzerren", hierbei wird der X- und Y- Maßstab gleichgesetzt. Dadurch



erhält man, wie der Name schon sagt, einen entzerrten Kartenausschnitt.

Damit der Benutzer seine gewünschte Flugroute an das Programm übergeben kann, ist hier die Möglichkeit zur grafischen Eingabe der hierfür nötigen Koordinaten geschaffen worden. Durch Anklicken der Karte werden die GPS Koordinaten des jeweiligen Punktes angezeigt. Diese können nun in die zu fliegende Route aufgenommen werden.

Um die Anzeigen auf der Karte auseinander zu halten, ist es möglich, für jede dieser Strecken eigene Eigenschaften wie z.B. Farbe und Strichstärke festzulegen. Damit immer klar erkennbar ist, in welche Richtung eine Linie zeigt, sind diese mit Pfeilen abgeschlossen.

Hierbei versuchten wir den Code so unabhängig von der Oberfläche wie möglich zu gestalten. Abgesehen von dem Anzeigeelement der Picture Box ist dies auch sehr gut gelungen. Die eingegebenen Flug-Daten werden in Arrays bereitgehalten und können somit bei Bedarf weiterverarbeitet werden.

Eine weitere Möglichkeit ist die Darstellung der Messdaten in Form von farbigen Punkten (s. unter 3.4). Da diese auch von GPS – Positionen abhängig sind, war es naheliegend, ihre Darstellung auch von diesem Modul übernehmen zu lassen.

#### 3.4 Die Messdatenauswertung

Um die Messdaten dem Benutzer verständlich zu präsentieren müssen diese als Grafik aufgearbeitet werden. Grundlage beim Erstellen dieser Software war, dass dem Benutzer alle Möglichkeiten in der Zuordnung der Werte offen bleiben. Da die Diagramme in eigenen Fenstern dargestellt werden, lassen sich mehrere verschiedene des gleichen Typs erzeugen um alle Werte im Auge zu behalten. Diese gewollten Features haben das Erstellen des Programmes nicht gerade leicht gemacht. Prinzipiell sind uns zwei verschiedene Diagrammtypen eingefallen.

Als Erstes wäre da das herkömmliche XY – Säulendiagramm. Beim XY – Diagramm wird der X – Achse eine feste Datenquelle (z.B. Zeit) zugeordnet, der Y – Achse hingegen können mehrere verschiedene Datenquellen mit jeweils einer eigenen Skala zugeordnet werden. Die Säulen können "nebeneinander", "übereinander" und "übereinander auf eine bestimmte Länge gestreckt" dargestellt werden. Jede Säule (Datenquelle) bekommt hierbei ihre eigene Farbe zugeordnet.

Die zweite Möglichkeit besteht darin, eine Landkarte als Grundlage für die Darstellung zu nutzen. Da hierbei X und Y bereits für die Position vergeben sind, werden die Informationen über den eigentlichen Messwert mit unterschiedlichen Farben dargestellt. Hierbei kann man bis zu 3 Messwerte pro Punkt darstellen, da sich jeder Rot-, Gelb- oder Blauwert im Bereich von 0 bis 255 einzeln ändern lässt. Falls man nur eine Datenquelle anzeigen will, ist es möglich auf Farbe zu verzichten, und die Darstellung in Grautönen (ebenfalls von 0 bis 255) vorzunehmen. Das Resultat ist nun eine Fläche, auf welcher mehrere verschiedenfarbige Punkte eingezeichnet sind, die nun die Messwerte symbolisieren.

Als neue Darstellungsmöglichkeit lässt sich jeder Sensorkanal auf Wunsch über kleine Anzeigefenster beobachten, die auch in der Lage sind beim über- bzw. unterschreiten bestimmter Werte eine Warnung auszugeben. Auch hier wurden zwei unterschiedliche Typen kreiert. Einmal ein Messbalken, vergleichbar einer LED - Bar auf einer Stereoanlage, und zum anderen eine Windrose, mit der sich Richtungen gut darstellen lassen.

In diesem Jahr neu hinzugekommen ist eine Exportfunktion in Exceldokumente. Dies ist gerade für professionelle Auswertungen eine nahezu unentbehrliche Funktion. Es wäre nahezu unmöglich und vor allem unsinnig alle Funktionen, die Excel für die Analyse bereithält, nachzubilden.

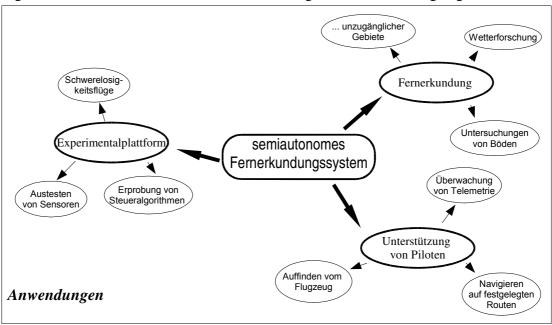
#### 3.5 Die Steuerung

Um das Flugzeugmodell im Rahmen der Benutzervorgaben eine gewisse Strecke fliegen zu lassen, hatten wir uns dafür eine Navigationshilfe überlegt. Diese Funktion sollte einmal in der Lage sein das Modell selbstständig zu steuern, da wir dies aber aus zeitlichen Gründen nicht geschafft haben, ist aus dieser Funktion etwas neues entstanden. Sie weißt nun dem Benutzer über eine der eben erwähnten Windrosen die Richtung, in die er fliegen muss, um seine vorgegebene Flugroute beizubehalten. Sie wäre zwar immer noch in der Lage diese Signale auch ans Flugzeug weiterzugeben, aber dies allein reicht nicht um das Flugzeug stabil in der Luft zu halten.

Zudem wird die aktuelle Flugrichtung, sowie die Richtung in der sich das Flugzeug befindet ausgegeben. Hierdurch wird es dem Piloten erleichtert das Flugzeug am Himmel wiederzufinden, falls er es verloren haben sollte. All diese Daten werden wie Sensoren (Virtuell) behandelt, und können somit auch aufgezeichnet und ins Verhältnis zu anderen Sensoren gesetzt werden. Die Flugrichtung lässt sich z.B. mit der Helligkeit in Verbindung setzen, um festzustellen, wie richtungsabhängig unser Lichtsensor ist.

#### 4 Die Einsatzgebiete

Uns sind direkt drei Bereiche eingefallen, in denen sich unser System anwenden lässt. Die folgenden Anwendungsmöglichkeiten sind nur ein Auszug aus den potentiellen Einsatzgebieten des semiautonomen Fernerkundungssystems. Es sind allerdings nicht alle dafür notwendigen Teile implementiert, sie sind jedoch leicht zu ergänzen. Darauf haben wir bei der Entwicklung besonderen Wert gelegt.



#### 4.1 Fernerkundung

Der erste Bereich, die Fernerkundung, beinhaltet alle Anwendungen, in denen es darum geht, aus großer Entfernung Informationen zu sammeln. Dabei dachten wir beispielsweise an die Erkundung unzugänglicher Gebiete (Tschernobyl, ...). Mithilfe des semiautonomen Fernerkundungssystem lässt sich eine direkte Gefahr für den Forscher vermeiden. Auch in der Wetterforschung hat ein Modellflugzeug gegenüber dem konventionell verwendeten Wetterballon einige Vorteile, da er sich direkt in die zu untersuchende Zone steuern lässt (Hurrican, ...). Mithilfe von Infrarotbildern, die mit einem Modellflugzeug als Kameraträger sehr leicht zu erstellen sind, lassen sich Aussagen über den Vegetationszustand eines bestimmten Gebietes (Agrarflächen, Biotope, ...) treffen. Damit lassen sich, am Beispiel der Agrarfläche betrachtet, die Pestizit- und Düngermenge auf ein Minimum reduzieren. Außerdem lassen sich stark durch Umweltbelastungen geschädigte Stellen in z.B. Biotopen ausfindig machen.

## 4.2 Unterstützung von Piloten

Ein weiterer Bereich der Anwendungsmöglichkeit ist uns im Gespräch mit einem Modellflugzeugpiloten eingefallen. Es handelt sich hierbei um mehrere Seite 13 von 15

Möglichkeiten, einen Piloten zu unterstützen: Der wichtigste Punkt ist vermutlich die Möglichkeit, eine Flugroute vorzugeben, die mithilfe der Softwarenavigation abgeflogen wird. Ein häufig auftauchendes Problem scheint das "Verlieren" des Flugzeugs zu sein. Dieses ist nun durch die Bildschirmanzeige sehr leicht wieder auffindbar. Außerdem erhält man kontinuierlich Informationen über den Zustand der einzelnen Flugzeugkomponenten (Motor, Akku, …).

#### 4.3 Experimentalplattform

Der dritte Bereich ist nicht ganz so praxisnah, sondern dient eher Forschungszwecken. Aufgrund der Beschleunigungssensoren lassen sich Schwerelosigkeitsflüge nachweisen und mit Experimenten (z.B. von anderen JuFo's) ausstatten. Da das semiautonome Fernerkundungssystem als Plattform ausgelegt ist, lassen sich jederzeit andere Sensoren anschließen und erproben. Für eine Entwicklung von Algorithmen, die zur Flugzeugsteuerung dienen, sind alle nötigen Soft- und Hardwaremodule vorhanden.

## 5 Das Ergebnis

Wir haben in diesem Jahr von der großen Idee, ein Flugzeug komplett autonom fliegen zu lassen, ein wenig Abstand genommen und im Gegenzug dafür ein bei weitem ausgereifteres und noch vielfältigeres System in der Hand. Desweiteren ist jetzt die Möglichkeit eines Rückkanals durch den erheblich leistungsfähigeren Prozessors im Flugzeug geschaffen worden.

Als Erweiterungsmöglichkeiten bieten sich eine Fülle von Sensoren an, die uns im späteren Verlauf des Projektes noch aufgefallen sind (Luftfeuchtigkeit, Luftdruck). Wir haben im Laufe des letzten Jahres einige Erkundigungen über die Schwierigkeit der automatischen Steuerung von Modellflugzeugen eingeholt. Uns gegenüber wurde mehrfach behauptet, dass diesdurch die richtige Technik (teuere Gyros anstatt billigem Kreisel von Conrad) sehr effizient möglich wäre. Da wir zum Zeitpunkt der Projektplanung noch nicht über ein ausreichend großes Modell verfügten, haben wir die autonome Steuerung für dieses Jahr erst einmal gelassen, zumal sich solche Lösungen für Modellflugzeuge fertig kaufen lassen. Wir sehen in der Messdatenerfassung und der Navigation ein im praktischem Einsatz viel mehr benötigtes System.

#### 6 Die Hilfen

Diesen Absatz haben wir den Leuten gewidmet, ohne die dieses Projekt niemals in dieser Form zustande gekommen wäre. Allen vorweg natürlich der TGA, der Ausbildungsabteilung von Bosch – Blaupunkt, und damit unserem Abteilungsleiter Herrn Königsdorff. Er hat uns vor allem beim Publizieren geholfen. Desweiteren geht großer Dank an unsere Ausbilder, die auch privat immer für uns da waren. Bei allen GPS bezüglichen Problemen standen uns Herr Bode, sowie Herr Mahli aus der Blaupunktentwicklung mit Tips und Material zur Seite. Ein Dank geht ebenso an die FV/SLH (Forschung, Herr Brenzel und Herr Semmler) die uns mit Tips und Material zur Seite stand.

Ferner möchten wir uns bei Frank Budnik bedanken, der uns in allen modelltechnischen Problemen zur Seite stand. Er war auch der Pilot während der Erprobung und stellte uns seine Modelle für die Testflüge zur Verfügung.

#### 7 Verwendete Literatur

Bücher: - Peter Monadjemi, Visual Basic 6 Kompendium, Markt &

Technik Buch- und Software-Verlag GmbH Haar bei

München, 1999

Bedienungshandbücher: - Artikel "Universeller ATV-FM Steuersender für 3,13 um

23 cm" aus der Zeitschrift TV-Amateur (Ausgabe

unbekannt da nur Kopie vorhanden)

- PIC 16F87x Serie Datenbuch von Microchip

- MPLABC (PIC C-Compiler) Handbuch

- PIC-Flasch II Dokumentation; www.uc-projekt.de

Datenblätter: - MAX 232 Datenblatt von Maxim (RS-232 Pegelwandler)

- MAX 512 Datenblatt von Maxim (D/A Wandler)

- MAX 3100 Datenblatt von Maxim (SPI zu USART

Wandler)

- FTR-700 der Firma PICIORGROS (Funkmodem)

- KTY - Silizium Temperatur Sensor Datenblatt von

Infineon

- ADXL 210 - 2 Achsen Beschleunigungsmesser von

**Analog Devices** 

- NMEA-String Dokumentation; National Marine Electronics Association; NMEA 0183 Version 2.01;

01.08.1994 (GPS String)

- JRC-String Dokumentation von Blaupunkt