

Inbetriebnahme einer freien Software zur Satellitenbahnvorhersage und Ansteuerung einer Hochleistungsantenne

STUDIENARBEIT
über das dritte Studienjahr

im Studiengang
Elektrotechnik, Nachrichten- und Kommunikationstechnik

an der DHBW Ravensburg
Campus Friedrichshafen

von

Sarah Brückner,
Maximilian Stiefel und
Hannes Bohnengel

15. Juli 2016

Bearbeitungszeitraum:	Oktober - Dezember 2015 April - Juni 2016
Betreuer:	Dipl.-Ing. (DH) Hardy Lau

Kurfassung

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean porttitor mi purus, sit amet efficitur velit semper sit amet. Mauris et pulvinar nunc, id maximus metus. Suspendisse convallis sapien nisi, sed maximus quam gravida eu. Duis faucibus elit non nunc posuere dignissim. Sed ullamcorper fringilla felis ac mollis. Nunc blandit tristique auctor. Praesent elementum dictum nulla et vulputate. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer sed rutrum lacus. Morbi tincidunt dui at augue molestie rhoncus. Curabitur sagittis sed sapien quis vestibulum. Cras dictum sem quam, nec tincidunt augue tempor et. Morbi consectetur, dui id scelerisque consequat, velit tortor gravida eros, ac hendrerit massa magna sit amet lacus. Quisque mattis nulla diam, sed efficitur mi fermentum vitae. Nam vestibulum iaculis rhoncus.

Abstract

Translation of „Kurzfassung“ comes here...

Erklärung

gemäß Ziffer 1.1.13 der Anlage 1 zu §§ 3, 4 und 5 der Studien- und Prüfungsordnung für die Bachelorstudiengänge im Studienbereich Technik der Dualen Hochschule Baden-Württemberg vom 29.09.2015.

Wir versichern hiermit, dass wir unsere Studienarbeit mit dem Thema:

**Inbetriebnahme einer freien Software zur
Satellitenbahnvorhersage und Ansteuerung einer
Hochleistungsantenne**

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben. Wir versichern zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Friedrichshafen, den 7. Juni 2016

Sarah Brückner

Maximilian Stiefel

Hannes Bohnengel

Inhaltsverzeichnis

Formelgrößen und Einheiten	I
Abkürzungen	II
1 Einleitung	1
2 Projektmanagement	2
2.1 Zeitplan	3
2.2 Anforderungsdefinition	3
2.3 Arbeitspakete	4
3 Amateurfunksatelliten	5
4 Hintergründe	7
4.1 Bahnmechanik	7
4.1.1 Die Keplerschen Gesetze	7
4.1.2 Die Bahnelemente	15
4.1.3 Bahnstörung	24
5 GPredict	25
5.1 Übersicht	25
5.2 Grafische Oberfläche	26
5.2.1 Grundansichten	27
5.2.2 Weitere Ansichten	29
5.2.3 Modul Pop-Up Menü	30
5.2.4 GPredict-Einstellungen	36
5.3 HamLib-Programmierschnittstelle	38
5.3.1 Übersicht	38
5.3.2 Parameter-Konfiguration	39
5.3.3 Verwendung	41
5.4 Inbetriebnahme unter Windows	42

5.4.1	Zusätzliche Software	42
5.4.2	Test der Bedienfelder	43
5.4.3	Test der Kommunikation zum Funkgerät	44
5.4.4	Test der Kommunikation zu den Rotoren	45
5.4.5	Konfiguration des Funkgeräts	46
5.4.6	Konfiguration der Rotoren	47
5.4.7	Probelauf und Testergebnisse	47
6	Zusammenfassung und Ausblick	48
	Abbildungsverzeichnis	III
	Tabellenverzeichnis	IV
	Literatur- und Quellenverzeichnis	V
	A Batch-Skript: <i>rigctl.d.bat</i>	VI
	B Batch-Skript: <i>rigctl.bat</i>	VII
	C Batch-Skript: <i>rotctl.d.bat</i>	VIII
	D Batch-Skript: <i>rotctl.bat</i>	IX
	E Batch-Skript: <i>rigctl.d-dummy.bat</i>	X
	F Batch-Skript: <i>rotctl.d-dummy.bat</i>	XI

Formelgrößen und Einheiten

Formelzeichen	Einheit	Abkürzung	Physikalische Größe
R	Ohm	Ω	Elektrischer Widerstand
U	Volt	V	Elektrische Spannung
P	Watt	W	Elektrische Leistung
f	Hertz	Hz	Frequenz
D	Bit	-	Daten
l	Meter	m	Länge
-	Dezibel	dB	Logarithmisches Maß (Pseudoeinheit)

Abkürzungen

API Application Programming Interface

AOS Acquisiton of Signal

BE Backend

CI-V Communication Interface-V

GPL General Public License

LOS Loss of Signal

PTT Push-to-Talk

CW Continious Wave

1 Einleitung

In der heutigen Gesellschaft ... Satelliten kaum mehr wegzudenken ... mobile Kommunikation, Navigation, Forschung, Satellitenfernsehen ...

Bei der Entwicklung und dem Test der Kommunikation zwischen Erde und Satellit haben Amateurfunksatelliten und Amateurfunkbodenstationen eine entscheidende Rolle gespielt (Quelle Amsat-Website)...

An der Dualen Hochschule Baden-Württemberg am Campus Friedrichshafen soll eine solche Bodenstation errichtet werden.

Die Software SatPC32 soll ersetzt werden und dient dabei als Referenz.

2 Projektmanagement

Schon Thomas Carlyle (1795–1881) erkannte die Wichtigkeit von strukturierten und organisiertem Vorgehen als er sagte:

“Unsere Hauptaufgabe ist nicht, zu erkennen, was unklar in weiter Entfernung liegt, sondern zu tun, was klar vor uns liegt”.

In einem Projekt ist das strukturierte und organisierte Vorgehen der klare Weg zu einem erfolgreichem Ziel. Daher wird sich in dieser Arbeit dem Projektmanagement bedient um die Antennennachführung für Satelliten in die richtige Richtung zu lotzen. Dabei lehnt sich das Management an das bekannte V-Modell, welche den Ablauf von Software-, als auch von Hardwareentwicklungsprozessen beschreibt. Dieses Modell soll einem Projekt die Richtung weisen, jedoch werden die einzelnen Schritte vom Projektmanager selbst definiert. Ein Vorgehensmodell wie dieses legt folgende Prozesse fest:

- die Aktivitäten die durchzuführen sind,
- die Reihenfolge des Arbeitsablaufes,
- die Definition von Ergebnissen,
- die Fertigstellungskriterien,
- die Ressourcen die vorhanden sind
- und die anzuwendenden Standards/Werkzeuge.

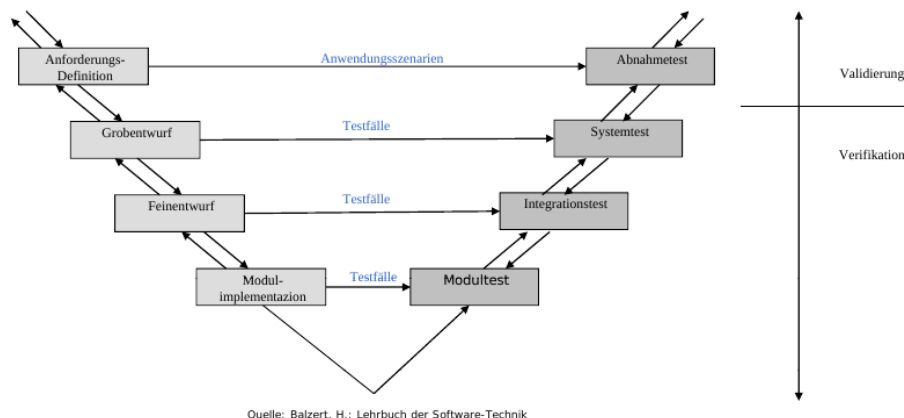


Abbildung 2.1: V-Modell, Quelle: [1]

Eine wichtige Rolle spielt die Qualitätssicherung, die das V-Modell sicher stellt. In diesem Modell sind die Verifikation und die Validation ein fester Bestandteil. Verifikation bedeutet, die Sicherstellung dass das entwickelte Produkt mit den Spezifikationen übereinstimmt. Die Validation ist die Eignung des Produkts bezogen auf seinen Einsatzzweck. Durch die Sicherstellung beider Qualitätsmerkmale wird das Projekt erfolgreich zu seinem Ziel, die Antennennachführung für Satelliten, geführt. Aus diesem Grund ist das V-Modell die richtige Vorgehensweise für dieses Projekt.

2.1 Zeitplan

Bevor ein Projekt starten kann, ist wichtig zu definieren, wie viel Zeit für die Durchführung zu Verfügung steht. Für diese Studienarbeit wird drei Studierenden einen Zeitrahmen von zwei Semester zugeschrieben. Dafür sind pro Studenten 300 Stunden kalkuliert. Beginn der Studienarbeit ist der 14. Oktober 2015 und endet am 15. Juli 2016. Für die Studierenden der Kurse TEN, TEK und TLE ist im Studienplan jeder Mittwoch von 14 Uhr bis 17.15 Uhr Studienarbeitszeit vorgesehen. Dies ist eine grobe Richtlinie für die Durchführung der Studienarbeit und kann nach Bedarf angepasst werden.

2.2 Anforderungsdefinition

Implementierung einer quelloffenen, kostenfreien Software zur Antennennachführung für das Verfolgen von erdnahen Satelliten. Die zur Antennennachführung benötigten Schnittstellen werden von der Hochschule zur Verfügung gestellt und sollen für diese Arbeit verwendet werden. Die Software soll die Satellitenposition mit Hilfe der Keplerelemente und den Doppler-Shift berechnen und ausgeben. Die Ausgabe soll grafisch erfolgen und anzeigen, wann der gewählte Satellit verfügbar ist. Ist der gewählte Satellit verfügbar, soll die Antenne in die richtige Position nachgeführt werden.

2.3 Arbeitspakete

Arbeitspakete brechen das Projekt in einzelne Aufgaben und vergeben diese an einen zuständigen Mitarbeiter. Daher ergibt sich ein detaillierter Projektablauf.

3 Amateurfunksatelliten

In dieser Studienarbeit soll die Bodenstation erdnahe Satelliten verfolgen können. Erdnah bedeutet auf einer niedrigen Umflaubahn bis etwa 1200 km. Dazu gehören die Amateurfunksatelliten, welche der Satellitenkommunikation zwischen Funkamateuren und auch zu experimentellen Zwecken dienen. Diese kommunizieren im 2-Meter- und 70-Zentimeterband [2] und ermöglichen einen internationalen Sprech- und Datenfunk. Außerdem senden diese Satelliten auch Messwerte der Betriebsdaten des Satelliten. Diese Satelliten werden meist von Hochschulen oder Amateurfunkvereinigungen gebaut und mit weiteren Satelliten an Board einer Rakete in das All geschossen. Dabei handelt es sich um kleine, leichte Satelliten, auch “Cubesat” genannt. Die Grafik 3.1 zeigt einen

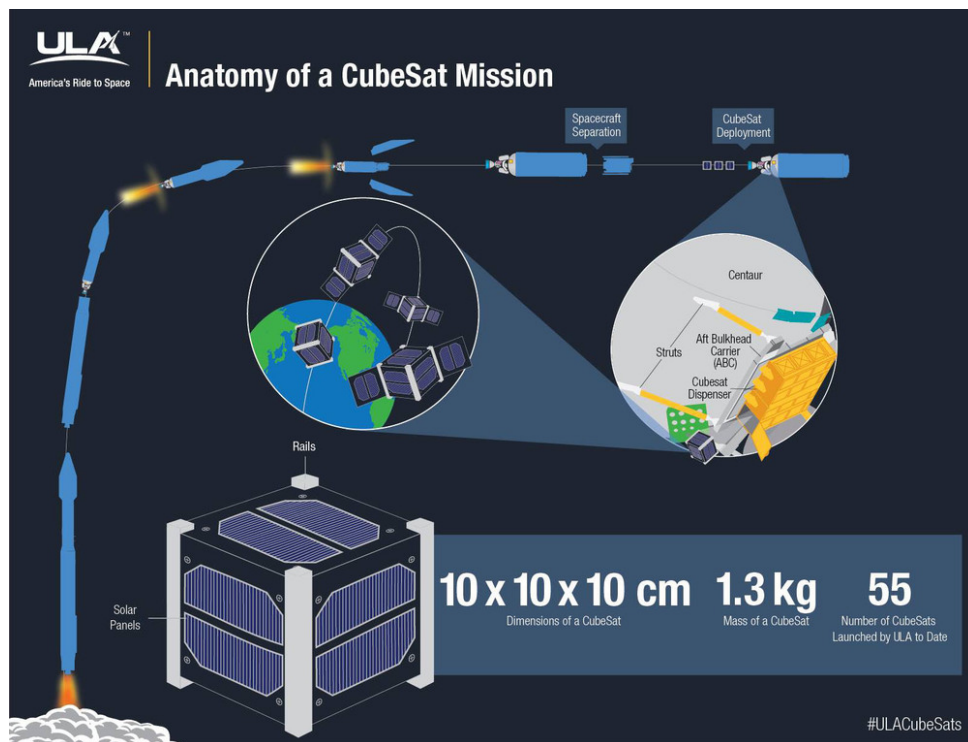


Abbildung 3.1: Cubesat, Quelle: [3]

beispielhaften Cubesat und wie dieser von einer Rakete in das All geschossen wird. Im Vergleich, der ASTRA 1L Satellit mit einer Spannweite von 20 Meter und einem Gewicht von 4,5 Tonnen. Erdnahe Satelliten umkreisen die Erde ständig und haben daher nur ein beschränktes Zeitfenster, in dem man mit einem solchen Satelliten arbeiten kann. Davor befindet sich der Satellit durch die Erdkrümmung hinter dem Horizont und ist daher unerreichbar. Daher ist es für die Bodenstation wichtig zu wissen wann der Satellit am

Horizont auftaucht um die Antenne in die richtige Position nachzuführen. Nach der korrekten Nachführung folgt die Kontaktaufnahme. Die Amateurfunksatelliten arbeiten in einem bestimmten Modus. Der Modus gibt an, in welchem Frequenzbereich der Satellit hört und auf welchem er sendet. Ein Signal, das der Satellit zur Erde schickt nennt man “Downlink” und von der Basisstation zum Satelliten “Uplink”. Das für diese Studienarbeit verwendete Funkgerät IC-9100 kann mit den Modis B und J arbeiten [4, S.153]

- Mode B: Uplink: 70 cm - Downlink: 2 m
- Mode J: Uplink: 2 m - Downlink: 70 cm

Einige Cubesats senden in Continuous Wave (CW) das Rufzeichen sowie ihre Telemetriedaten aus und dient der Funktionskontrolle.

4 Hintergründe

4.1 Bahnmechanik

Es sei zu Beginn dieses Absatzes darauf hingewiesen

4.1.1 Die Keplerschen Gesetze

Seit der Antike galt die Erklärung der Bewegung der Planeten und die Vorhersage dieser als eine große Herausforderung. Theorien von Ptolemaios mit seinem geozentrischen Weltbild und Kopernikus mit seinem heliozentrischen Weltbild führten bereits im 16. Jahrhundert zu brauchbaren Modellen zur Vorhersage der Planetenbewegungen. Diese Modelle unterlagen jedoch Ungenauigkeiten, „die in mit Instrumenten des 16. Jahrhunderts bereits messbaren Breichen lagen“ (siehe S. 20 in [5]). Der mathematische Aufwand hinter diesen



Abbildung 4.1: Johannes Kepler (1571-1630), Quelle: [6]

Modellen war enorm. Selbst das kopernikanische Weltbild, das einige Vereinfachungen mit sich brachte, bediente sich der Überlagerung einer Vielzahl von Kreisbewegungen, um das Verhalten der Planeten zu erklären. Resignierend zog sich zu der Zeit die katholische Kirche und mit ihr viele Gelehrte auf den Standpunkt zurück, dass „die Frage, welche der Theorien die korrekte sei, [...] schlicht unbeantwortbar“ wäre (siehe S. 21 in [5]).

Ein deutscher Mathematiker und Astronom, Johannes Kepler, war hier anderer Auffassung. Er war überzeugter Kopernikaner und stand im Dienste des Kaisers Rudolph II.

Schließlich gelang es ihm aus seinen Beobachtungen drei einfache Gesetze herzuleiten. Seine Gesetze führten zu Vorhersagen der Planetenbewegungen nie da gewesener Präzision, welche er seinem Dienstherrn widmend in den Rudolphinischen Tabellen niederschrieb. Steiner und Schlagerl schreiben in Ihrem Buch „Raumflugmechanik“, dass ohne die Vorarbeit Keplers keine Weltraumtechnik je existiert hätte (vgl. S. 21 in [5]). Die drei Gesetze lauten:

1. Keplersches Gesetz: Die Planeten umlaufen die Sonne auf elliptischen Bahnen. In einem der Brennpunkte dieser Ellipsen befindet sich die Sonne.
2. Keplersches Gesetz: Die Linie von der Sonne zu einem Planeten überstreicht in gleichen Zeiten gleiche Flächen.
3. Keplersches Gesetz: Die Quadrate der Umlaufzeiten zweier Planeten verhalten sich zueinander so wie die Kuben der großen Halbachsen ihrer Bahnellipsen.

Kepler starb 1630 und damit 12 Jahre vor Isaacs Newtons (1642-1726) Geburt. Mit seinen Werken hinterließ Kepler Newton alles, um das Gravitationsgesetz später herleiten zu können.

Das erste Keplersche Gesetz

Durch die Annahme Planeten bewegen sich auf Ellipsen anstatt auf Kreisen, brach Kepler ein tausende Jahre altes Paradigma. Das mit der Ellipse war allerdings nicht seine Idee. Bereits im 11. Jahrhundert nahm ein arabischer Gelehrter namens Al-Zarkali (1029-1087) eine elliptische Bahn an, um die unregelmäßige Bewegung des Merkurs erklären zu können. Kepler kannte diese Idee durch die Lehren des Mathematikers und Astronomen Peurbach (1423-1461), welcher die Ellipsen-Theorie im Abendland verbreitete.

Zunächst soll die Ellipse an sich betrachtet werden. Die einfachste Möglichkeit eine Ellipse zu konstruieren besteht darin zwei Nägel in einer Holzplatte mit einem Stück Schnur mit einer Schlaufe zu verbinden. Das Stück Schnur muss länger sein als der Abstand zwischen beiden Nägeln. Nimmt man nun einen Bleistift und drückt ihn in der Schlaufe gegen die Schnur, kann man die beiden Nägel mit Kontakt der Bleistiftspitze zum Holzbrett umrunden. Hält man die Schnur konstant auf Spannung, so ergibt sich eine Ellipse. Darüber hinaus muss der Punkt auf welchem die Schlaufe am Bleistift anliegt höher sein, als der

$$E = \{P | \overline{F_1 P} + \overline{F_2 P} = 2a = \textit{konstant}\} \quad (4.1)$$

Man spricht bei der Länge dieser Strecke von der **großen Halbachse** **a**. Beide Strecken ergeben zusammen die Hauptachse $\overline{S_1S_2}$. Analog gibt es hierzu die Nebenachse, welche durch die Strecke $\overline{S_3S_4}$ bestimmt wird. Die **kleinen Halbachsen** sind $\overline{MS_3}$ und $\overline{MS_4}$. Diese haben die Längen **b**. Eine Ellipse kann auch als Stauchung eines Kreises mit dem Faktor $\frac{b}{a}$ angesehen werden.

$$e' = \frac{e}{a} \quad (4.2)$$

9

M liegen. Für $e' = 1$ entartet die Ellipse zu einer Geraden, da die kleine Halbachse b zu 0 wird. Um das zu zeigen wird die obige Gleichung noch mal herangezogen.

$$e'^2 = \frac{e^2}{a^2} = \frac{a^2 - b^2}{a^2} = 1 - \left(\frac{b}{a}\right)^2 \quad (4.3)$$

Des Weiteren besitzt jede Ellipse einen Halbparameter p . Geht man davon aus, dass es einen Abstand p' gibt, welcher p bis zu einer die Ellipse umschließende Kreislinie verlängert, so gelten folgende Gleichungen

$$\frac{p}{p'} = \frac{b}{a} \quad (4.4)$$

Mit dem Satz eines alten Freudes folgt

$$p' = \sqrt{a^2 - e^2} \quad (4.5)$$

Jetzt ist klar, dass gilt

$$p = \frac{b}{a} \cdot p' = \frac{b}{a} \sqrt{a^2 - e^2} = \frac{b^2}{a} \quad (4.6)$$

Was nun noch fehlt ist „eine Gleichung, also eine analytische Beschreibung der Punkte einer Ellipse“ (siehe S.22 in [5]). Eine solche Gleichung ergibt sich mit dem Schnitt einer Ebene mit einem Kegel. Die Neigung der Schnittebene zur Kegelgrundfläche sei α . Der Öffnungswinkel des Kegels sei ϵ . Jetzt passiert etwas, dass das räumliche Denkvermögen herausfordert. In den Kegel wird eine (Dandelin'sche) Kugel eingeschrieben. Diese Kugel berühre die Ebene im Punkt F und tangiere den Kegelmantel entlang eines Breitenkreises. Es ist einzusehen, dass der Punkt F auf der Hauptachse der Ellipse liegt. Der Schnittpunkt der entstehenden Ellipse und der Normale zur Hauptachse im Punkt F ist der Punkt Q . P stellt einen beliebigen Punkt auf der Ellipse dar. Die Verbindungslinie zwischen F und P hat zur Hauptachse die Neigung θ . Der Abstand zwischen F und P ist r . s und s_0 sind die Abstände der Punkte P und Q zum Berührkreis der Kugel gemessen entlang des Kegelmantels.

Man sieht nun, dass die Größen r , s und θ von P abhängig sind. Es interessiere nun die mathematische Funktion $r(\theta)$, welche die Ellipsenbahn beschreibe (vgl. S.23 in [5]). Betrachtet wird nun zunächst die zweidimensionale Zeichnung rechts oben in Abb. 4.4. Mit ein bisschen Nachdenken sieht man, dass

$$(s_0 - s) \cdot \cos(\epsilon) = r \cdot \cos(\theta) \cdot \cos(\alpha) \quad (4.7)$$

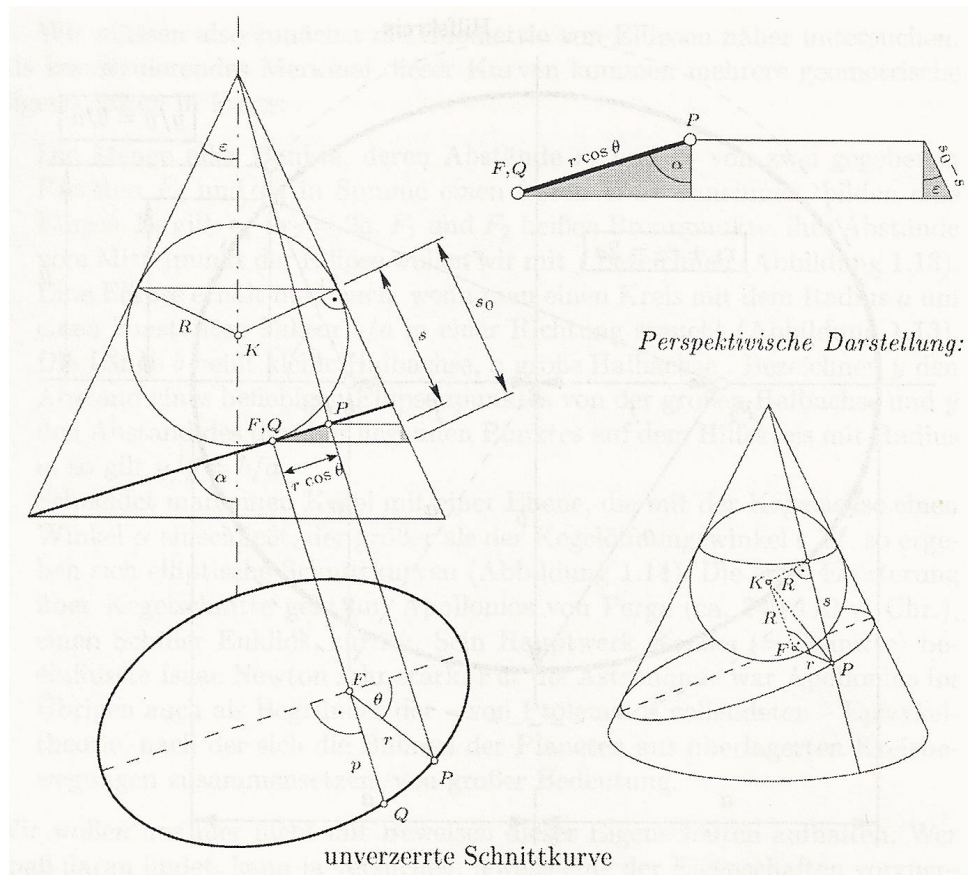


Abbildung 4.3: Kegelschnitt, Quelle: S.24 in [5]

Doch woher kommt der Ausdruck $r \cdot \cos(\theta)$? Hierzu werfe man einen Blick auf die zweidimensionale Abbildung der Schnittfläche/Ellipse links unten im Bild. Dieses Bild setze man nun in Relation zum Bild darüber. Der Abstand $r \cdot \cos(\theta)$ lässt sich nun auf die Hauptachse der Ellipse projizieren. r , die Projektionslinie für P die Hauptachse und F bilden nun eine rechtwinkliges Dreieck aus. Der Rest ist Trigonometrie.

In der Darstellung rechts unten in Abb. 4.4 ist folgende Beziehung auffindbar.

$$R^2 + s^2 = R^2 + r^2 \quad (4.8)$$

Das bedeutet, dass s durch r in Gleichung 4.7 ersetzt werden kann.

$$(s_0 - r) \cdot \cos(\epsilon) = r \cdot \cos(\theta) \cdot \cos(\alpha) \quad (4.9)$$

Ausmultiplizieren ergibt

$$s_0 \cos(\epsilon) - r \cos(\epsilon) = r \cos(\theta) \cos(\alpha) \quad (4.10)$$

Sortieren führt zu

$$s_0 \cos(\epsilon) = r \cos(\theta) \cos(\alpha) + r \cos(\epsilon) \quad (4.11)$$

Ausklammern und auflösen bringt

$$r = \frac{s_0 \cos(\epsilon)}{\cos(\theta) \cos(\alpha) + \cos(\epsilon)} \quad (4.12)$$

Die entstandene Gleichung 4.12 kann nun noch durch die Zusammenhänge $p = s_0$ (Halbparameter) und $e' = \frac{\cos(\alpha)}{\cos(\epsilon)}$ vereinfacht werden (vgl. S. 24 in [5]). Hierzu dividiert man Gleichung 4.12 durch $\cos(\epsilon)$.

$$r = \frac{s_0}{1 + \frac{\cos(\alpha)}{\cos(\epsilon)} \cos(\theta)} = \frac{p}{1 + e' \cos(\theta)} \quad (4.13)$$

Fertig ist die mathematische Version des ersten Keplerschen Gesetzes.

Das zweite Keplersche Gesetz

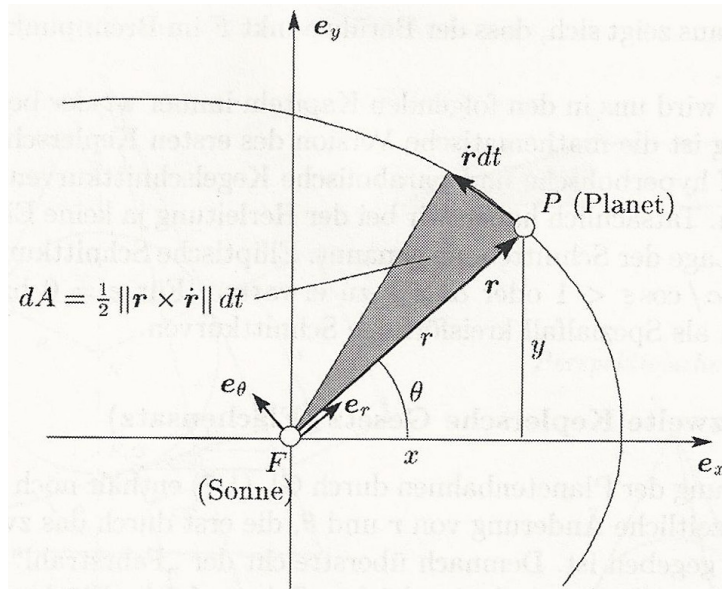


Abbildung 4.4: Kegelschnitt, Quelle: S.26 in [5]

Gleichung 4.13 liefert noch keine Aussage über die zeitliche Änderung von r und θ . Um eine Aussage über die zeitliche Änderung dieser Variablen treffen zu können, kann Keplers zweites Gesetz herangezogen werden: Die Fläche, welche die Verbindungslinie zwischen Sonne und einem Planet überstreicht ist zeitlich konstant. Die Fläche ΔA , die in einem Zeitintervall Δt durchstrichen wird ist genau

$$\Delta A = \frac{1}{2} \left| \vec{r} \times \dot{\vec{r}} \right| \Delta t + F(\Delta t^2) \quad (4.14)$$

Die beiden Vektoren r und $\dot{r} \Delta t$ spannen eine Fläche auf, welche ein Parallelogramm beschreibt. Das Kreuzprodukt ergibt einen Vektor dessen Länge dieser Fläche entspricht.

Die Hälfte davon ist die Fläche des Dreiecks, die gesucht wird. Der Ausdruck $\dot{r}\Delta t$ ist dabei sehr ungenau und beschreibt eigentlich nur die Änderung des Vektors r . Aus diesem Grund kommt noch der Fehlerterm F hinzu, der die Krümmung der Ellipse berücksichtigt. Bezieht man sich im nächsten Schritt auf infinitesimale Elemente, die wirklich gegen Null gehen, so erreicht man die gewünschte Genauigkeit. Der Fehlerterm wird überflüssig.

$$dA = \frac{1}{2} \left| \vec{r} \times \dot{\vec{r}} \right| dt \quad (4.15)$$

Man führt nun eine für jeden Planeten individuelle Konstante h ein, da sich das Verhältnis $\frac{dA}{dt}$ nicht verändern darf.

$$2dA = h \cdot dt \quad (4.16)$$

Um das mathematische Äquivalent zu dem sprachlich formulierten zweiten Gesetz zu erhalten, soll wie beim ersten Gesetz eine Abhängigkeit zu r und θ hergestellt werden. Zu diesem Zweck wird die Gleichung einer Koordinatentransformation in Zylinderkoordinaten unterworfen (vgl. S. 25 f. in [5]). Es gilt also

$$x = r \cos(\theta), y = r \sin(\theta) \text{ und } z = z.$$

Gemäß der Definition von Zylinderkoordinaten darf man jetzt den Vektor \vec{r} auch anders schreiben:

$$\vec{r} = r \vec{e}_r + z \vec{e}_z \quad (4.17)$$

wobei folgendes generell über Zylinderkoordinatensysteme bekannt ist:

$$\vec{e}_r = \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \\ 0 \end{pmatrix}, \vec{e}_\theta = \begin{pmatrix} -\sin(\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} \text{ und } \vec{e}_z = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Die Vektoren stehen allesamt senkrecht aufeinander, was man sieht, wenn man das Skalarprodukt bildet. Das liegt daran, dass das Skalarprodukt über die Summe der Längen der Vektoren multipliziert mit dem Kosinus des Winkels den sie einschließen definiert wird, welcher bei $\frac{\pi}{2}$ bekanntlich Null ist. Da auch die Ableitung des Vektors \vec{r} (Geschwindigkeit) gesucht ist beginnt man zu differenzieren. Man wende hier zunächst die Summenregel, dann auf den ersten Ausdruck noch die Produkt- und die Kettenregel, da \vec{e}_r von θ abhängt und diese wiederum von t . Es folgt

$$\dot{\vec{r}} = \dot{r} \vec{e}_r + r \dot{\vec{e}}_r + \dot{z} \vec{e}_z \quad (4.18)$$

Setzt man nun

$$\dot{\vec{e}}_r = \dot{\theta} \begin{pmatrix} -\sin(\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} = \dot{\theta} \vec{e}_\theta$$

in Gleichung 4.18 ein, so ergibt sich

$$\dot{\vec{r}} = \dot{r} \vec{e}_r + r \dot{\theta} \vec{e}_\theta + \dot{z} \vec{e}_z \quad (4.19)$$

Man wählt nun die z-Achse geschickt, so dass diese senkrecht auf der Trägerebene der Ellipse steht (vgl. S.26 in [5]). Durch diesen Schachzug gilt für die zu betrachtenden Gleichungen $z = 0$. Jetzt fällt Gleichung 4.15 in sich zusammen

$$\frac{dA}{dt} = \frac{1}{2} \left| \vec{r} \times \dot{\vec{r}} \right| = \frac{1}{2} \left| r \vec{e}_r \times (\dot{r} \vec{e}_r + r \dot{\theta} \vec{e}_\theta) \right| \quad (4.20)$$

Durch die Bilinearität des Kreuzprodukts folgt

$$\frac{dA}{dt} = \frac{1}{2} \left| \dot{r} r (\vec{e}_r \times \vec{e}_r) + r r \dot{\theta} (\vec{e}_r \times \vec{e}_\theta) \right| \quad (4.21)$$

Die Tatsache, dass das Kreuzprodukt eines Vektors mit sich selbst den Nullvektor ergibt und dem Umstand, dass $\vec{e}_r \perp \vec{e}_\theta \perp \vec{e}_z$ ist, führt zu

$$\frac{dA}{dt} = \frac{1}{2} \left| r^2 \dot{\theta} \vec{e}_z \right| = \frac{1}{2} r^2 \dot{\theta} = h = \text{konstant} \quad (4.22)$$

Das dritte Keplersche Gesetz

Auf sein drittes Gesetz soll Kepler besonders stolz gewesen sein (vgl. S.27 in [5]). Nach seinem Gesetz stehen die Halbachsen a_1 und a_2 und die Umlaufzeiten T_1 und T_2 in dem Zusammenhang

$$\frac{T_1^2}{T_2^2} = \frac{a_1^3}{a_2^3} \quad (4.23)$$

Die Umlaufzeit lässt sich nun leicht mit der Konstanten h , welche für das zweite Keplersche Gesetz eingeführt wurde ausdrücken. Es wird hierfür die gesamte Ellipsenfläche ($A_{\text{Ellipse}} = ab\pi$) und konsequenterweise dann auch die gesamte Umlaufzeit T angenommen. Eingesetzt in Gleichung 4.22 ergibt sich

$$2ab\pi = hT. \quad (4.24)$$

Daraus folgt

$$T = \frac{2ab\pi}{h}. \quad (4.25)$$

Eingesetzt in Gleichung 4.23 ergibt sich

$$\frac{T_1^2}{T_2^2} = \frac{a_1^3}{a_2^3} = \frac{a_1^2 b_1^2 h_2^2}{a_2^2 b_2^2 h_1^2} \quad (4.26)$$

Bringt man nun $a_1^2 b_1$ und $a_2^2 b_2^2$ einmal durch Division und einmal durch Multiplikation eins weiter nach links, so kann man die Beziehung aus Gleichung 4.6 ausnutzen und schreiben

$$\frac{a_1 b_2^2}{a_2 b_1^2} = \frac{h_2^2}{h_1^2} \quad (4.27)$$

Oben und unten multipliziert man jetzt noch jeweils mit $1/a_1$ und $1/a_2$.

$$\frac{b_2^2/a_2}{b_1^2/a_1} = \frac{h_2^2}{h_1^2} = \frac{p_2}{p_1} \quad (4.28)$$

Das Resultat ist ein „Zusammenhang zwischen den Halbparametern p_1, p_2 und den Bewegungskonstanten h_1, h_2 “ (siehe S.27 in [5]). Diese Größen sind aus dem zweiten Keplerschen Gesetz bekannt.

4.1.2 Die Bahnelemente

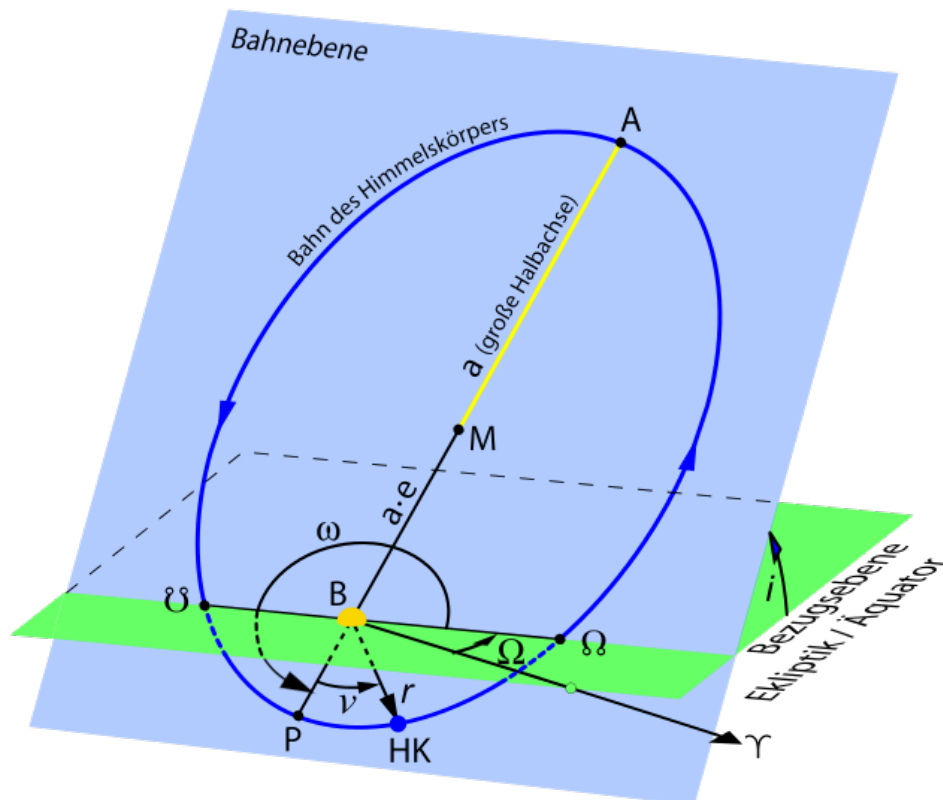


Abbildung 4.5: Bahnelemente, Quelle: [7]

Die Bahnelemente dienen der Beschreibung einer Bewegung eines Himmelskörpers auf einer Umlaufbahn (meist einer Ellipse). Dieser Körper unterliegt den Keplerschen Gesetzen. Wird die Bewegung eines Himmelskörpers durch äußere Einflüsse (z.B. Gravitationskraft der Sonne) nicht gestört, so kann sie durch sechs Größen beschrieben werden. Diese Größen sind die Bahnelemente. Zwei Bahnelemente beschreiben die **Form der Bahn**, drei legen die **Lage der Bahn** im dreidimensionalen Raum fest und ein Bahnelement gibt an zu welcher **Zeit** sich der Himmelskörper wo auf der Bahn befunden hat.

Diese Bahnelemente reichen in der Praxis nicht aus, um die Position eines Himmelskörpers z.B. eines Satelliten mit einem Vorhersagemodell berechnen zu können. Aus diesem Grund werden die Bahnelemente meist um von Vorhersagemodellen benötigten Informationen ergänzt. Im Folgenden werden die Bahnelemente in Ihrer Bedeutung anhand der Abbildung 4.5 erläutert.

Gestalt der Bahn

Um die Gestalt der Bahn zu beschreiben wird die **numerische Exzentrizität (1)** e' und die Angabe der Länge der **großen Halbachse (2)** a benötigt. Diese beiden Größen werden ausführlich im Absatz 4.1.1 beschrieben. Es braucht nicht mehr, um eine elliptische Bahn zu definieren. Zur Erinnerung: Die große Halbachse a ist die Länge der Strecke vom Mittelpunkt zu den Hauptscheiteln. Die (lineare) Exzentrizität e gibt die Schlankeit der Ellipse an. Wobei die numerische Exzentrizität e' auf die große Halbachse normiert ist.

Die Abhängigkeit der Umlaufzeit eines Satelliten von den Bahndimensionen (v.a. Länge der großen Halbachse) und die elliptische Bahnform sind der „quadratischen Abnahme der Gravitationsbeschleunigung [geschuldet]“ (siehe S. 87 in [8]).

$$\ddot{\vec{r}} = -\frac{GM_{Erde}}{r^2} \cdot \frac{\vec{r}}{r} \quad (4.29)$$

$GM_{Erde} \approx 398600,4 \frac{km^3}{s^2}$ ist dabei das konstante Produkt aus Gravitationskonstante und Masse der Erde. Das Stichwort lautet hier Impulserhaltungssatz. In einem geschlossenen System kann Impuls weder vernichtet noch erschaffen werden. Bis zum Perigäumsdurchgang nimmt ein Satellit Impuls von der Erde über die Kraft, die zum Erdmittelpunkt hin wirkt auf. Danach wird Impuls an die Erde abgegeben \rightarrow der Satellit verlangsamt sich. Nach dem Apogäumsdurchgang fängt der Satellit wieder an schneller zu werden. Er nimmt wieder Impuls auf.

Lage der Bahn

Es erscheint sinnvoll, dass die Lage im Raum als nächstes festgelegt wird. Auch scheint es sinnig, dass man drei Bahnellemente benötigt, um die Lage der Bahn im dreidimensionalen Raum festzulegen.

Man stelle sich nun vor, dass die Ellipse, auf welcher der Satellit sich bewegt, in einer Ebene liege. Diese Ebene schneide man mit einer Referenzebene (z.B. Schnittebene durch den Erdäquator). Die **Inklination (3)** i ist nun ein Winkel, „um den die Bahnebene gegenüber der [Referenzebene] geneigt ist“ (siehe S.88 in [8]). Um die Nebenachse der Ellipse herum kann durch diese Festlegung keine Rotation mehr stattfinden. Die Ellipse, die ja in der Ebene liegt, kann sich jetzt noch in dieser Ebene um eine zur Ebene orthogonalen Achse drehen. Um das zu unterbinden wird ein weiterer Winkel, das **Argument des Perigäums (4)** ω , eingeführt. Das Argument des Perigäums ist der Winkel, den die Knotenlinie und die Perigäumsrichtung (Hauptachse) einschließen. Das Perigäum liegt am Ende der Hauptachse, auf welches sich der Satellit zu bewegt, nachdem er den aufsteigenden Knoten passiert hat. Der aufsteigende Knoten ist der Punkt in dem der Satellit die Referenzebene auf seiner elliptischen Umlaufbahn (von Süd nach Nord) durchstößt. Die Knotenlinie ist die Schnittgerade der Ellipse mit der Referenzebene. Darüber hinaus gibt es das Apogäum, welches nach dem absteigenden Knoten durchlaufen wird, der am anderen Ende der Nebenachse beheimatet ist. Eine andere Definition für das Perigäum ist der Punkt des geringsten Abstands des Satelliten zur Erde. Analog lässt sich das Apogäum als Punkt des größten Abstands des Satelliten zur Erde festlegen. Die Ellipse dreht sich nun nicht mehr in der Ebene. Eine Sache wurde jetzt noch nicht betrachtet. Die Ebene kann sich immer noch entlang Ihrer Hauptachse drehen. Die **Rektaszension des aufsteigenden Knotens (5)** Ω gibt nun den Winkel an, den die Knotenlinie und der Vektor vom Ellipsenmittelpunkt zum Frühlingspunkt γ einschließen. Doch was ist der Frühlingspunkt? „Als Frühlingspunkt (auch Widderpunkt) wird in der Astronomie der Schnittpunkt des Himmelsäquators mit der Ekliptik bezeichnet, an dem die Sonne zum Frühlingsanfang der Nordhalbkugel (= Herbstanfang der Südhalbkugel) steht“ (siehe [9]). Die Drehung um die Hauptachse ist jetzt also auch nicht mehr möglich. Der Bahnlage wurden somit alle drei Freiheitsgrade genommen.

Die Bewegung eines Satelliten erfolgt in einer nicht veränderbaren Bahnebene mit dem

Bahnmittelpunkt gleich dem Erdmittelpunkt. Dies impliziert das erste Keplersche Gesetz. Der Grund hierfür ist „dass die Anziehung der Erde (in erster Näherung) immer zum Erdmittelpunkt gerichtet ist“ (siehe S. 87 in [8]). Zu keinem Zeitpunkt gibt es somit einen Kraftvektor, der Senkrecht zum Ortsvektor \vec{r} wirkt und somit die Bahnebene verschieben könnte (Annahme: Mittelpunkt des Koordinatensystems ist der Erdmittelpunkt). Daraus resultiert wiederum, dass eine einmal eingeschlagene Bahnebene ohne eine äußere Kraft nicht wieder verlassen werden kann.

Zeitlicher Bezug und Kepler-Gleichung

Bis jetzt ist alles recht schön und anschaulich. Es ist klar wo die Bahnebene im Raum ist und wie diese aussieht. Wir haben durch das zweite Keplersche Gesetz eine Gleichung, die uns die Position des Satelliten in Abhängigkeit des Winkels θ verrät. Diese ist

$$r = \frac{p}{1 + e' \cos(\theta)} \quad (4.30)$$

θ ist der Winkel, den die Verbindungslinie zwischen einem Brennpunkt und dem Satellit mit der Hauptachse einschließt. Es gibt allerdings noch ein Haar in der Suppe. Aufgrund des Flächensatzes ist die Winkelgeschwindigkeit des Satelliten nicht konstant (bzw. nur für eine in der Natur nicht vorkommende Kreisbahn konstant). θ ist eine Funktion von t . Die sog. wahre Anomalie $\theta(t)$ schwankt um die mittlere Anomalie M (nicht zu verwechseln mit der Erdmasse). Man stößt nun auf das Zweikörperproblem (Keplerproblem). Dies ist die Angabe der wahren Anomalie $\theta(t = t_x)$ zu einem vorgegebenen Zeitpunkt t_x . Wie kommt die mittlere Anomalie nun zustande? Zunächst sollte ein genauer Blick auf folgende Abbildung 4.6 geworfen werden. Die Abbildung stellt eine Momentaufnahme einer Situation dar. Wie schon bei früheren Betrachtungen wird die Ellipse (Orbit) in einen Umkreis eingeschrieben. Ein fiktiver (mittlerer) Satellit Y laufe entlang des Umkreises um den Punkt C (Mittelpunkt der Ellipse und des Kreises, kein Ellipsenbrennpunkt). Der wahre Satellit P bewege sich entlang der Ellipse. Die Lage des fiktiven Satelliten Y wird zum Zeitpunkt t mittels des Winkels M dargestellt. Hierbei wird unterstellt, dass der Satellit zum Zeitpunkt t_P im Perigäum (oft auch Periapsis genannt) $M = 0$ war. Es gilt nun

$$M = 2\pi \frac{t - t_P}{U} = n(t - t_P) \quad (4.31)$$

Hierbei ist n die Winkelgeschwindigkeit. S ist der Erdmittelpunkt (Schwerezentrum). Um nun die Kurve zu der Ellipse und dem tatsächlichen Satellit zu bekommen wird das

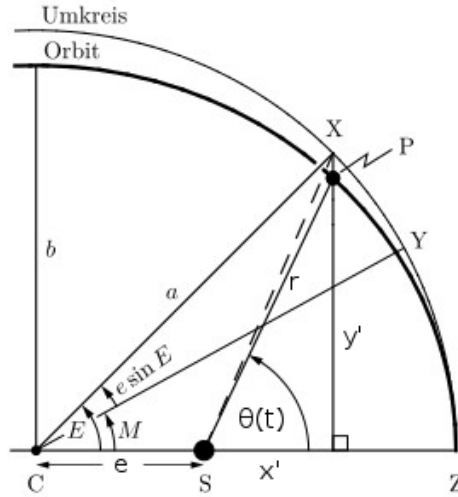


Abbildung 4.6: Kepler-Gleichung, Quelle: [10], Grafik ist eigens modifiziert

zweite Keplersche Gesetz für diesen konkreten Fall herangezogen. In gleichen Zeiträumen werden gleiche Flächen von den Fahrstrahlen der Satelliten unabhängig von der Bahnform (egal ob elliptisch oder kreisförmig mit $e' = e = 0$) überstrichen. Hier spricht man von einer bijektiven Abbildung der Ellipsen- und Kreisflächensegmente. Letztere sind leicht zu berechnen. Es wird also praktisch ein Umweg über den Umkreis gegangen. Das zweite Keplersche Gesetz angewandt auf den vorliegenden Fall, führt bezüglich der Zeitspanne einer Überstreichung der gesamten Fläche zu

$$\frac{A_{Kreis}}{A_{Ellipse}} = \frac{\pi a^2}{\pi a b} \quad (4.32)$$

Nach dem zweiten Keplerschen Gesetz ist dieses Verhältnis dasselbe für eine infinitesimale Fläche. So gilt

$$\frac{A_{CZY}}{A_{SPZ}} = \frac{A_{Kreis}}{A_{Ellipse}} = \frac{a}{b} \quad (4.33)$$

Kepler führte als eine Hilfsgröße die exzentrische Anomalie E ein. Hierzu wird P auf den Hilfskreis projiziert. Es entsteht der Punkt X . Der Winkel zwischen Hauptachse und X ist die exzentrische Anomalie. Durch den in Gleichung 4.33 hergeleiteten Zusammenhang kann folgende Aussage getroffen werden.

$$A_{SXZ} = \frac{a}{b} A_{SPZ} \quad (4.34)$$

Setzt man Gleichung 4.33 in Gleichung 4.34 ein, so ergibt sich

$$A_{SXZ} = A_{CZY} \quad (4.35)$$

Mit der Gleichung 4.35 ist ein indirekter Zusammenhang zwischen der mittleren Anomalie (Punkt Y) und der exzentrischen Anomalie (Punkt X) gefunden. Der Fahrstrahl \overline{CY} in der Zeit einer Periode U eine Fläche von πa^2 überstreicht, so überstreicht er in einer Zeit t eine Fläche, die um den Faktor $\frac{M}{2\pi}$ kleiner ist. Es folgt

$$A_{CYZ} = \frac{M}{2\pi} \pi a^2 = \frac{a^2}{2} M \quad (4.36)$$

Analog gilt dies für den Fahrstrahl \overline{CX} .

$$A_{CXZ} = \frac{a^2}{2} E \quad (4.37)$$

Weiter geht's mit noch mehr Geometrie. Die Fläche A_{CXZ} kann in zwei Teilflächen zerlegt werden.

$$A_{CXZ} = A_{CSX} + A_{SXZ} \quad (4.38)$$

Die Fläche A_{CSX} kann als die Fläche eines Dreiecks aufgefasst werden. Dabei ist $a \cdot e'$ die Basis und $a \sin(E)$ die Höhe des Dreiecks.

$$A_{CSX} = \frac{1}{2} \cdot a \cdot e' \cdot a \sin(E) = \frac{a^2}{2} e' \sin(E) \quad (4.39)$$

Als letzten Schritt setze man nun noch Gleichung 4.35 (unter Beachtung von Gleichung 4.36), Gleichung 4.37 und Gleichung 4.39 in Gleichung 4.38 ein.

$$\frac{a^2}{2} E = \frac{a^2}{2} e' \sin(E) + \frac{a^2}{2} M \quad (4.40)$$

Mit ein paar Tricks aus der Grundschule entsteht die **Kepler-Gleichung** wie sie im Buche steht.

$$E - e' \sin(E) = M = U \cdot t \quad (4.41)$$

Diese Gleichung kann nicht nach E aufgelöst werden und so keine schöne Funktion $E(t)$ gewonnen werden. Doch Gott sei dank, schenke er Newton die Eingebung des Newton-Verfahrens zur Nullstellensuche. Das Newton-Verfahren ist im Grunde eine rekursive Folge, wobei jedes Folgenglied allgemein mit

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (4.42)$$

zu bestimmen ist. Man stellt also Gleichung 4.41 entsprechend um und leite ab.

$$f(E) = E - e' \sin(E) - M \stackrel{!}{=} 0 \quad (4.43)$$

$$f'^E(E) = 1 - e' \cos(E) \quad (4.44)$$

Es folgt die rekursive Folge für die Kepler-Gleichung.

$$E_{n+1} = E_n - \frac{E - e' \sin(E) - M}{1 - e' \cos(E)} \quad (4.45)$$

Als Startwert wird in [8] $E_0 = M$ empfohlen. Die rekursive Folge Verhält sich dabei so, dass sie gegen einen entsprechenden Wert für die exzentrische Anomalie E konvergiert. Wenn sich entsprechende Nachkommastellen nach gewisser Genauigkeit nicht mehr ändern, so hat man die Nullstelle gefunden. Mit diesem Wert für $E(t = t_x)$ kann die wahre Anomalie $\theta(t = t_x)$ bestimmt werden. Zu beachten ist, dass in die Kepler-Gleichung natürlich vor der Anwendung des Newton-Verfahrens auch der gewünschte Zeitpunkt t_p eingesetzt wird.

Wie bestimmt man nun die wahre Anomalie, also den Winkel, der zu dem Zeitpunkt gehört aus welchem wir dann wiederum einen Vektor r gewinnen können? Hierzu werden die Größen x' und y' in Abbildung 4.6 herangezogen. Man sucht nun nach einer Beziehung zwischen diesen Größen und bekannten Größen aus dem Fundus der Bahnelemente die Form der Bahn betreffend (dazu gehören natürlich auch abgeleitete Größen). Es ist nachvollziehbar, dass

$$x' = r \cos(\theta) = \cos(E) a - e = a (\cos(E) - e') \quad (4.46)$$

ist. Für y' muss man etwas tiefer in die Trickkiste greifen. Man geht aus von Gleichung 4.4. Diese stelle man nach p um.

$$p = p' \frac{b}{a} \quad (4.47)$$

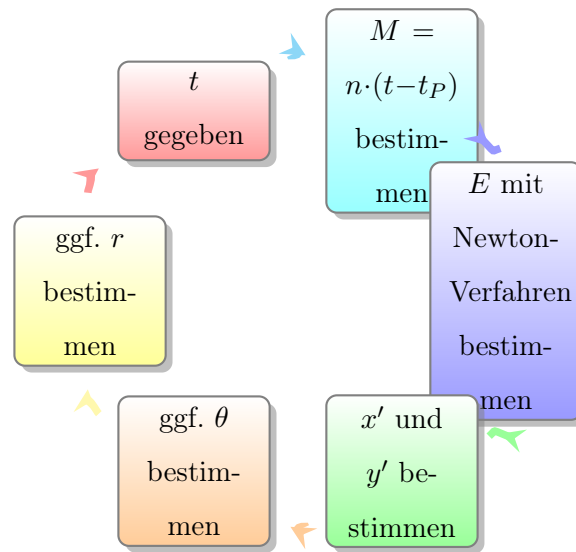
Für p' kann der Term $\sin(E) a$ eingesetzt werden. Mit dem Satz des Pythagoras kann b auch mit den Größen a und e ausgedrückt werden. Es folgt

$$p = y' = \sin(E) a \frac{\sqrt{a^2 - e^2}}{a} = \sin(E) a \sqrt{1 - e'^2} \quad (4.48)$$

Aus diesen beiden Größen kann unmittelbar die wahre Anomalie θ zu einem gegebenen Zeitpunkt bestimmt werden.

$$\theta = \arctan\left(\frac{y'}{x'}\right) = \arctan\left(\frac{\sin(E) a \sqrt{1 - e'^2}}{a (\cos(E) - e')}\right) = \arctan\left(\frac{\sin(E) \sqrt{1 - e'^2}}{\cos(E) - e'}\right) \quad (4.49)$$

Nun ist also der zeitliche Bezug bekannt. Hier nochmal eine kleine Darstellung wie vorzugehen ist.



Führt man den Vorgang wie oben gezeigt iterativ aus, so erhält man einen zweidimensionalen Vektor \vec{r} . Wenn man nun scharf nachdenkt, so fällt einem auf, dass man nochmal eine Information benötigt. Es braucht einen Zeitpunkt t_P . Dies muss ein Zeitpunkt sein, bei welchem der Satelliten im Perigäum stand. Als Analogie könnte man die Phase eines Signals in der Elektrotechnik angeben. Genau diesen Zeitpunkt liefern die Bahnelemente als zeitlichen Bezug.

Die Bahnelemente sind also

- Gestalt der Bahn
 - die große Halbachse a
 - die numerische Exzentrizität e'
- Lage der Bahn
 - die Inklination i
 - die Rektaszension des aufsteigenden Knotens Ω
 - das Argument des Perigäums ω
- Zeitlicher Bezug
 - die Epoche (wann ist der Satellit im Perigäum gewesen) t_P

Zum Schluss trägt man alle Informationen aus den Bahnelementen zusammen, um einen Vektor \vec{r} im dreidimensionalen kartesischen Koordinatensystem zu erhalten. Der Ursprung des Koordinatensystems soll im Erdzentrum liegen und der Vektor \vec{r} soll ein Ortsvektor des Satelliten sein. Die x-y-Ebene dieses Koordinatensystems ist der Äquator. Das Ergebnis der bisherigen Berechnungen ist der Vektor

$$\vec{r}' = a \cdot \begin{pmatrix} \cos(E) - e' \\ \sin(E) \sqrt{1 - e'^2} \\ 0 \end{pmatrix} \quad (4.50)$$

Die Satellitenbahn liegt aktuell in der äquatorialen Ebene. Der Erdmittelpunkt liegt im Brennpunkt der Ellipse. Anstatt die Bahnebene zu drehen, dreht man nun das Koordinatensystem. Allerdings in die mathematisch entgegengesetzte Richtung. Hierfür können natürlich nur die Winkel verwendet werden, welche in den Bahnelementen für die Lage der Bahn mitgeliefert werden. Bei der Drehung wird auf Drehmatrizen zurückgegriffen. Eine gute Veranschaulichung dieser Drehungen findet man *in dieser Animation* (<https://www.youtube.com/watch?v=QZrYaKwZwhI>). Der gesuchte Vektor ergibt sich nun zu

$$\vec{r} = \mathbf{R}_Z(-\Omega) \mathbf{R}_X(-i) \mathbf{R}_Z(-\omega) \cdot a \cdot \begin{pmatrix} \cos(E) - e' \\ \sin(E) \sqrt{1 - e'^2} \\ 0 \end{pmatrix} \quad (4.51)$$

Hierbei stellen \mathbf{R}_X und \mathbf{R}_Z folgende Drehmatrizen dar (vgl. S.89 in [8]).

$$\mathbf{R}_X(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & +\cos \alpha & +\sin \alpha \\ 0 & -\sin \alpha & +\cos \alpha \end{pmatrix} \quad (4.52)$$

$$\mathbf{R}_Z(\alpha) = \begin{pmatrix} +\cos \alpha & +\sin \alpha & 0 \\ -\sin \alpha & +\cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4.53)$$

Eine Frage, die sich aufdrängen mag ist: Warum kann die Drehung der Ellipse in der Bahnebene (Argument des Perigäums ω) einfach auf eine Drehung der z-Achse zurückgeführt werden? Man kann es sich auch so vorstellen, dass zunächst die Ellipse noch in der äquatorialen Ebene liegt, die z-Achse dann um $-\omega$ gedreht wird, dann die x-Achse um $-i$ und letztendlich die z-Achse nochmal um $-\Omega$ gedreht wird.⁶

Nach [8] ergibt sich nach der Ausführung der Drehungen der Vektor

$$\vec{r} = r \cdot \begin{pmatrix} \cos(\omega + \theta) \cos(\Omega) - \sin(\omega + \theta) \cos(i) \sin(\Omega) \\ \cos(\omega + \theta) \sin(\Omega) + \sin(\omega + \theta) \cos(i) \cos(\Omega) \\ \sin(\omega + \theta) \sin(i) \end{pmatrix} \quad (4.54)$$

Durch Ableitung nach der Zeit kommt man wie in [8] angegeben auf

$$\vec{v} = \mathbf{R}_Z(-\Omega) \mathbf{R}_X(-i) \mathbf{R}_Z(-\omega) \cdot \frac{\sqrt{GM_{Erde} a}}{r} \cdot \begin{pmatrix} -\sin(E) \\ \cos(E) \sqrt{1 - e'^2} \\ 0 \end{pmatrix} \quad (4.55)$$

Gleichung 4.54 lässt sich so interpretieren, dass der Term $\sqrt{\frac{GM_{Erde}}{r^2 a}}$ die zeitliche Ableitung der exzentrischen Anomalie E sein muss, da die Kettenregel der Differentialrechnung greift und die Rotationsmatrizen keine zeitlichen Abhängigkeiten aufweisen.

Mit den Gleichungen 4.54 und 4.55 lässt sich nun die Antenne fast ausrichten und nachführen.

4.1.3 Bahnstörung

5 GPredict

Eine sehr detaillierte Beschreibung der Software GPredict ist unter [11] in englischer Sprache verfügbar. Dieses Kapitel stellt eine kompakte Beschreibung mit ausschließlich für dieses Projekt relevanten Informationen dar. Insofern können gewisse Parallelen nicht ausgeschlossen werden, wobei darauf hingewiesen sei, dass der Mehrwert dieser Zusammenfassung in der Kürze und Relevanz im Vergleich zum englischen Pendant liegt.

5.1 Übersicht

GPredict ist eine freie Software zur Satellitenverfolgung und Orbitvorhersage und steht als Quellcode oder bereits fertig kompiliertes Programm für Windows, Mac OS und Linux zur Verfügung. Die Software ist in C geschrieben und unter der GNU General Public License (GPL) lizenziert, somit kann sie frei verändert und an die entsprechenden Nutzervoraussetzungen angepasst werden. Im Rahmen dieser Projektarbeit wird Version 1.3 von GPredict verwendet (verfügbar unter [12]).

In Abbildung 5.1 ist das Prinzip eines Satellitenverfolgungsprogramms zu sehen (die blauen Blöcke stellen hierbei die Funktionalität des Programms dar). Zunächst wird an Hand der Keplerschen Bahnelemente und dem aktuellen Zeitpunkt die absolute Position des Satelliten berechnet. Daraufhin wird der Vektor, der von der Bodenstation zum Satelliten zeigt, bestimmt. Nun können Azimut und Elevation dieses Vektors für die Ansteuerung der Antenne verwendet werden.

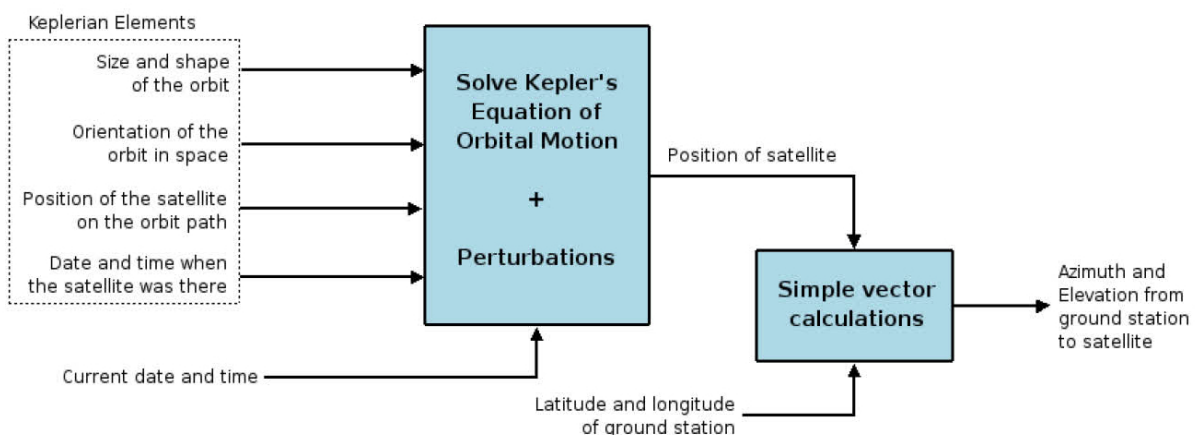


Abbildung 5.1: Prinzip eines Satellitenverfolgungsprogramms, Quelle: [11]

Zur Berechnung der Satellitenposition wird auf den NORAD SGP4/SDP4 Algorithmus zurückgegriffen (siehe Abschnitt XXX). Um hierfür zu jedem Zeitpunkt die aktuellen Kepler-Elemente des zu verfolgenden Satelliten zu kennen, gibt es unter GPredict die Möglichkeit einer automatischen Aktualisierung über HTTP, FTP oder aus dem lokalen Verzeichnis.

Bei GPredict ist im Gegensatz zu anderen Satellitenverfolgungsprogrammen wie SatPC32 kein Limit an zu verfolgenden Satelliten und Bodenstationen gegeben. Durch die Verwendung von Modulen kann außerdem unkompliziert zwischen verschiedenen Konfigurationen gewechselt werden. Die Orbitvorhersage eines Satelliten lässt sich sowohl grafisch als auch tabellarisch darstellen, wobei durch die Einstellungen verschiedener Parameter eine sehr individuelle Anzeige erreicht werden kann.

5.2 Grafische Oberfläche

In Abbildung 5.2 ist die grafische Oberfläche von GPredict zu sehen. In der Standardkonfiguration ist dort zunächst die Kartenansicht bzw. *Map View* (oben), die Polaransicht bzw. *Polar View* (links unten) und die Einzelsatellitenansicht bzw. *Single-Satellite View* (rechts unten) zu sehen.

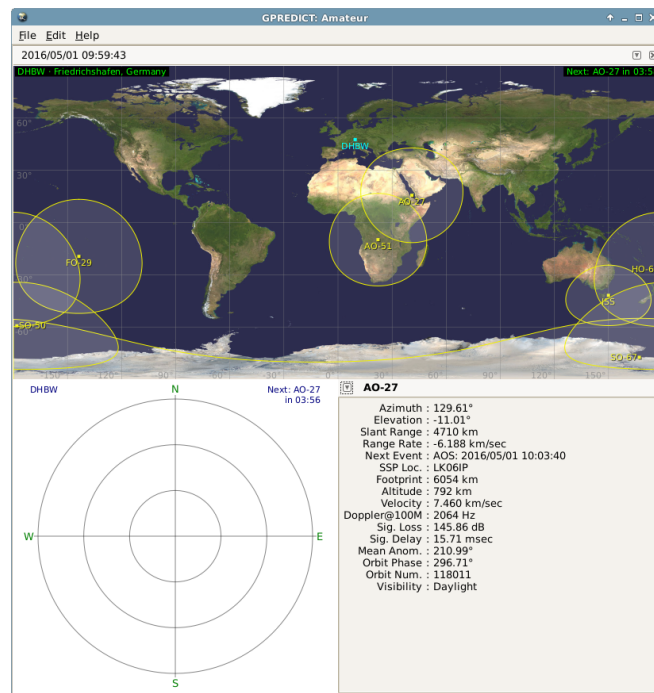


Abbildung 5.2: Standardoberfläche von GPredict

5.2.1 Grundansichten

Zu den oben genannten Ansichten kommen noch zwei Weitere hinzu, die Listenansicht bzw. *List View* und eine Ansicht für bevorstehende Durchläufe, die sogenannte *Upcoming Passes View*. Im Folgenden werden die verschiedenen Ansichten genauer beschrieben:


Map View

Diese Ansicht besteht, wie in Abbildung 5.2 zu sehen, aus einer Weltkarte auf der die aktuellen Standorte der für das aktuelle Modul ausgewählten Satelliten zu sehen ist. Das heißt der Punkt auf dem der entsprechende Satellit senkrecht bezogen auf den Erdmittelpunkt steht. Außerdem ist um diesen Punkt die Fläche umrahmt, von der der Satellit von der Erde aus sichtbar ist. Mit einem Rechtsklick auf einen Satellitennamen kann außerdem die Option *Ground Track* aktiviert werden, mit welcher die Spur des Satelliten für mehrere Orbits angezeigt wird.

Polar View

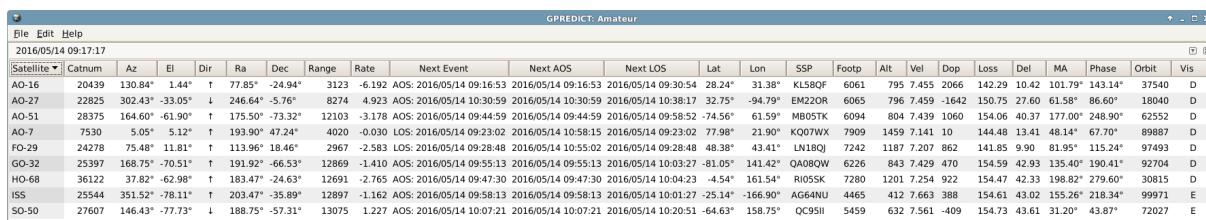
Die *Polar View* (siehe Abbildung 5.2) stellt eine Draufsicht auf die Bodenstation dar, bei der die Polarachse den Azimutwinkel darstellt und die Radialachse den Elevationswinkel. Mit einem Rechtsklick auf einen Satelliten lässt sich mit der Option *Show sky track* aktivieren, das die Spur des entsprechenden Satelliten angezeigt wird. Zusätzlich wird das aktuelle Modul links oben angezeigt, der nächste sichtbare Satellit (rechts oben) und die genauen Werte für Azimut und Elevation (links unten) sobald sich der Mauszeiger auf der *Polar View* befindet.

Single-Satellite View

In dieser Ansicht (siehe Abbildung 5.2) werden detaillierte Informationen zu einem ausgewählten Satelliten angezeigt, z.B. Azimut, Elevation, Entfernung der direkten Sichtverbindung (*Slant Range*), Höhe, Geschwindigkeit, Dopplerverschiebung oder Signaldämpfung. Mit einem Klick auf das -Symbol links neben dem Satellitennamen kann zwischen den für dieses Modul ausgewählten Satelliten gewechselt werden.

List View

Die Listenansicht zeigt eine tabellarische Auflistung aller für das aktuelle Modul ausgewählten Satelliten mit verschiedenen Details, mit je einem Satelliten pro Zeile. In Abbildung 5.3 ist die Listenansicht mit allen verfügbaren Details zu sehen. Mit einem Klick auf eine entsprechende Kategorie lässt sich das Sortierkriterium ändern. Falls hier ein variables Kriterium wie die Geschwindigkeit eingestellt wird, ändert sich die Sortierreihenfolge mit der eingestellten Auffrischrate (*Refresh Rate*). Die Bezeichnung des jeweiligen Details ist in dieser Ansicht abgekürzt, z.B. Az für *Azimut*. Unter den Moduleinstellungen beim Reiter *List View* kann ausgewählt werden, welches Detail angezeigt wird. Dort ist außerdem erkenntlich für was die entsprechenden Abkürzungen stehen.



Satellite	Catnum	Az	El	Dir	Ra	Dec	Range	Rate	Next Event	Next AOS	Next LOS	Lat	Lon	SSP	Footp	Alt	Vel	Dop	Loss	Del	MA	Phase	Orbit	Vis
AO-16	20439	130.84°	1.44°	↑	77.85°	-24.94°	3123	-6.192	AOS: 2016/05/14 09:16:53	2016/05/14 09:16:53	2016/05/14 09:30:54	28.24°	31.38°	KL58QF	6061	795	7.455	2066	142.29	10.42	101.79°	143.14°	37540	D
AO-27	22825	302.43°	-33.05°	↓	246.64°	-5.76°	8274	4.923	AOS: 2016/05/14 10:30:59	2016/05/14 10:30:59	2016/05/14 10:38:17	32.75°	-94.79°	EM22OR	6065	796	7.459	-1642	150.75	27.60	61.58°	86.60°	18040	D
AO-51	28375	164.60°	-61.90°	↑	175.50°	-73.32°	12103	-3.178	AOS: 2016/05/14 09:44:59	2016/05/14 09:44:59	2016/05/14 09:58:52	-74.56°	61.59°	MB05TK	6094	804	7.439	1060	154.06	40.37	177.00°	248.90°	62552	D
AO-7	7530	5.05°	5.12°	↑	193.90°	47.24°	4020	-0.030	LOS: 2016/05/14 09:23:02	2016/05/14 10:58:15	2016/05/14 09:23:02	77.98°	21.90°	K007WX	7909	1459	7.141	10	144.48	13.41	48.14°	67.70°	89887	D
FO-29	24278	75.48°	11.81°	↑	113.96°	18.46°	2967	-2.583	LOS: 2016/05/14 09:28:48	2016/05/14 10:55:02	2016/05/14 09:28:48	48.38°	43.41°	LN18QJ	7242	1187	7.207	862	141.85	9.90	81.95°	115.24°	97493	D
GO-32	25397	168.75°	-70.51°	↑	191.92°	-66.53°	12869	-1.410	AOS: 2016/05/14 09:55:13	2016/05/14 09:55:13	2016/05/14 10:03:27	-81.05°	141.42°	QA08QW	6226	843	7.429	470	154.59	42.93	135.40°	190.41°	92704	D
HO-68	36122	37.82°	-62.98°	↑	183.47°	-24.63°	12691	-2.765	AOS: 2016/05/14 09:47:30	2016/05/14 09:47:30	2016/05/14 10:04:23	-4.54°	161.54°	RI05SK	7280	1201	7.254	922	154.47	42.33	198.82°	279.60°	30815	D
ISS	25544	351.52°	-78.11°	↑	203.47°	-35.89°	12897	-1.162	AOS: 2016/05/14 09:58:13	2016/05/14 09:58:13	2016/05/14 10:01:27	-25.14°	-166.90°	AG64NU	4465	412	7.663	388	154.61	43.02	155.26°	218.34°	99971	E
SO-50	27607	146.43°	-77.73°	↓	188.75°	-57.31°	13075	1.227	AOS: 2016/05/14 10:07:21	2016/05/14 10:07:21	2016/05/14 10:20:51	-64.63°	158.75°	QC95II	5459	632	7.561	-409	154.73	43.61	31.20°	43.87°	72027	E

Abbildung 5.3: List View

Upcoming Passes View

Die *Upcoming Passes View* (siehe Abbildung 5.4) zeigt alle Satelliten des aktuellen Moduls, deren Azimut und Elevation, sowie die Zeit bis zum nächsten Verschwinden des Satelliten, dem sogenannten *Loss of Signal (LOS)* bzw. dem nächsten Auftauchen, auch *Acquisition of Signal (AOS)* genannt. Wie bei der *List View* ist es auch hier möglich nach den verschiedenen Spalten zu sortieren.

Satellite	Az	El	▼	AOS/LOS
AO-27	29.48°	21.85°		05:04
AO-16	182.57°	-49.98°		25:16
FO-29	170.06°	-60.09°		26:43
AO-7	157.48°	-65.08°		32:06
GO-32	339.04°	-64.81°		55:41
HO-68	237.62°	-52.19°		01:08:45
SO-50	152.56°	-31.48°		01:15:43
AO-51	47.69°	-72.15°		09:34:23
ISS	114.72°	-65.28°		12:46:49

Abbildung 5.4: Upcoming Passes View

5.2.2 Weitere Ansichten

Bei allen Ansichten kann durch einen Klick auf den Satellitennamen ein kleines Pop-Up Menü geöffnet werden, welches den entsprechenden Satellitennamen, die Option *Show next pass* und die Option *Future passes* anzeigt. Bei einem Klick auf den Satellitennamen öffnet sich ein Fenster mit dem Titel *Satellite Info*, wie in Abbildung 5.5 zu sehen. Dort sind unter dem Reiter *Orbit Info* verschiedene Informationen zum Satellitenorbit und unter dem Reiter *Transponders* die verfügbaren Transponder zu sehen.

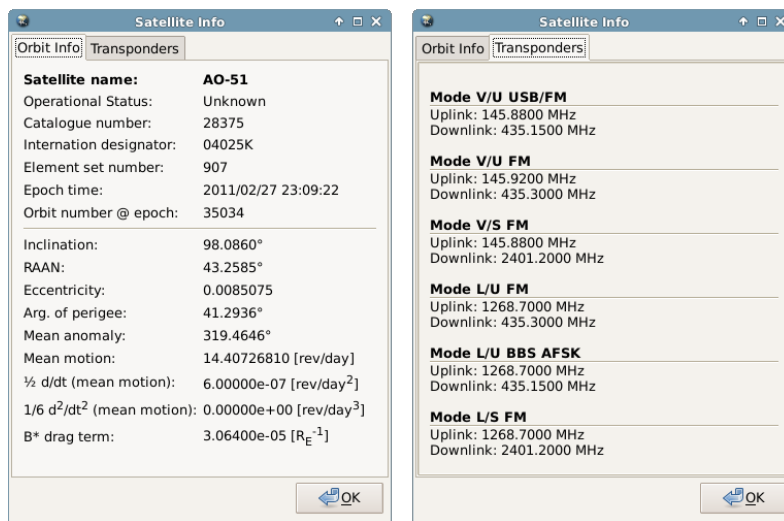


Abbildung 5.5: *Satellite Info*

Mit einem Klick auf die Option *Show next pass* gelangt man zu einer Übersicht über den nächsten Durchlauf des entsprechenden Satelliten. Die Details sind tabellarisch, als Polaransicht und als Verlauf des Azimut- und Elevationswinkels über der Zeit zu sehen (siehe Abbildung 5.6).

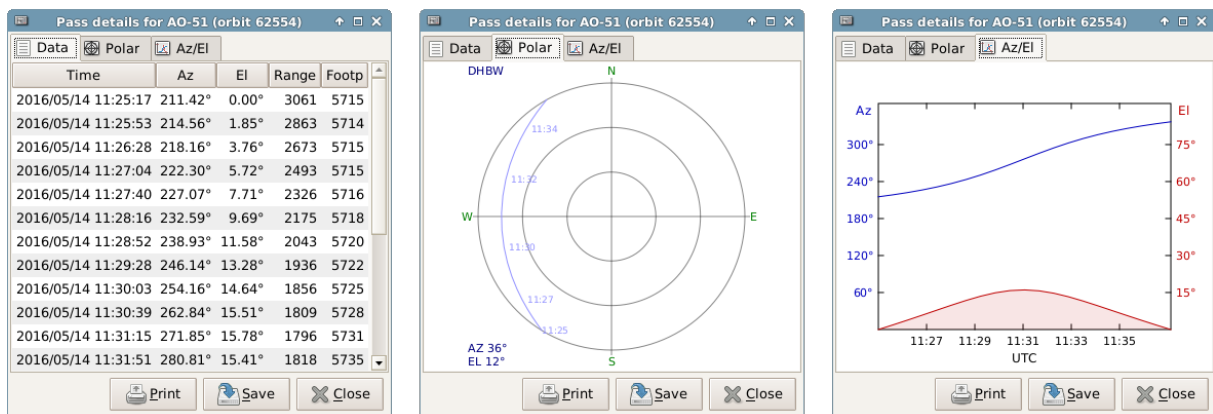
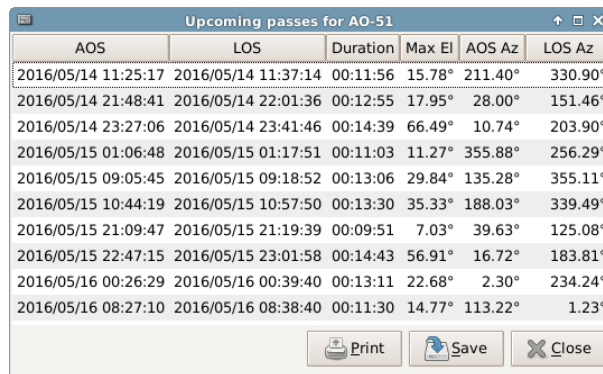


Abbildung 5.6: *Pass Details*


Die Option *Future passes* öffnet ein Fenster, in welchem die nächsten Durchläufe des entsprechenden Satelliten tabellarisch dargestellt sind (siehe Abbildung 5.7). Hierbei ist die Anzahl der darzustellenden Durchläufe in den GPredict-Einstellungen unter *Predict* als *Number of passes to predict* einstellbar.



AOS	LOS	Duration	Max El	AOS Az	LOS Az
2016/05/14 11:25:17	2016/05/14 11:37:14	00:11:56	15.78°	211.40°	330.90°
2016/05/14 21:48:41	2016/05/14 22:01:36	00:12:55	17.95°	28.00°	151.46°
2016/05/14 23:27:06	2016/05/14 23:41:46	00:14:39	66.49°	10.74°	203.90°
2016/05/15 01:06:48	2016/05/15 01:17:51	00:11:03	11.27°	355.88°	256.29°
2016/05/15 09:05:45	2016/05/15 09:18:52	00:13:06	29.84°	135.28°	355.11°
2016/05/15 10:44:19	2016/05/15 10:57:50	00:13:30	35.33°	188.03°	339.49°
2016/05/15 21:09:47	2016/05/15 21:19:39	00:09:51	7.03°	39.63°	125.08°
2016/05/15 22:47:15	2016/05/15 23:01:58	00:14:43	56.91°	16.72°	183.81°
2016/05/16 00:26:29	2016/05/16 00:39:40	00:13:11	22.68°	2.30°	234.24°
2016/05/16 08:27:10	2016/05/16 08:38:40	00:11:30	14.77°	113.22°	1.23°

Abbildung 5.7: *Upcoming Passes*

5.2.3 Modul Pop-Up Menü

Um das Modul Pop-Up Menü zu öffnen, klickt man ganz rechts oben im GPredict-Fenster auf das -Symbol. Im daraufhin erscheinenden Pop-Up Menü ist es möglich die Positionierung eines Moduls innerhalb des GPredict-Fensters einzustellen, ein Modul zu kopieren, zu löschen, zu schließen oder genauer zu konfigurieren. Außerdem sind dort weitere Funktionen, welche im Folgenden genauer beschrieben werden, zugänglich.

Wie in Abbildung 5.8 zu sehen, bietet die Funktion *Sky at a glance* eine Übersicht darüber, wann welche Satelliten innerhalb der nächsten acht Stunden sichtbar sind. Dieser Zeitraum lässt sich in den GPredict-Einstellungen bei *Predict* unter dem Reiter *Sky at a Glance* zwischen einer und 24 Stunden einstellen.

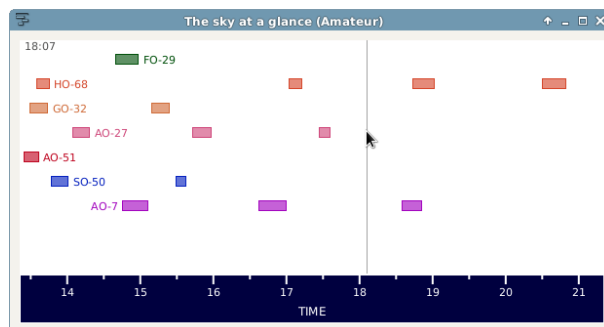
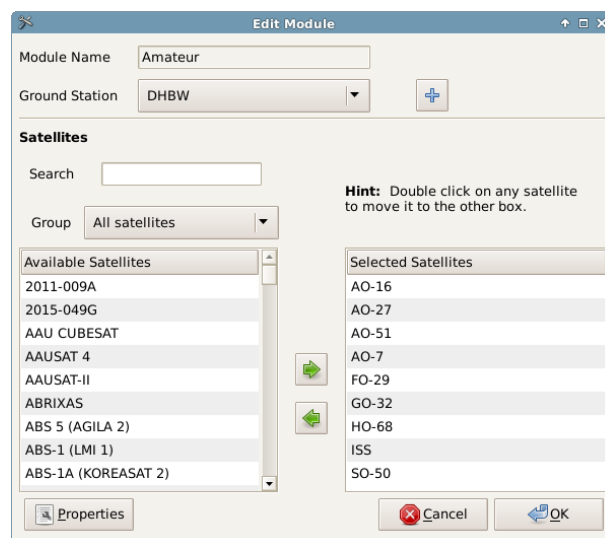


Abbildung 5.8: *The sky at a glance*

Über die Funktion *Time Controller* (siehe Abbildung 5.9) lässt sich die Zeit, auf die sich die Berechnungen von GPredict beziehen, ändern. Hierbei ist standardmäßig das aktuelle Datum und die aktuelle Uhrzeit eingestellt. Außerdem kann hier die Geschwindigkeit, mit der die eingestellte Zeit fortschreitet, auf maximal ein Hundertfaches erhöht werden. Die eingestellte Zeit wird im GPredict-Fenster ganz links oben im ausgewählten Format angezeigt. Mit dem Schieberegler kann die Zeit zwischen $-2,5$ und $+2,5$ Stunden eingestellt werden.


Abbildung 5.9: *Time Controller*

Klickt man auf *Configure*, öffnet sich ein Fenster wie in Abbildung 5.10 zu sehen. Hier lassen sich die zu verfolgenden Satelliten und die Bodenstation für das aktuelle Modul auswählen. Außerdem gelangt man mit einem Klick auf das Feld *Properties* in die Modul-Einstellungen. Diese gelten im Gegensatz zu den in den allgemeinen GPredict-Einstellungen zu findenden Modul-Einstellungen nur für das aktuelle Modul. Auf Seite 37 sind nähere Infos zu den Modul-Einstellungen zu finden.


Abbildung 5.10: *Edit Module*

Hinter der Funktion *Antenna Control* (siehe Abbildung 5.11) verbirgt sich ein Bedienfeld zur Steuerung der Antennenrotoren. Bevor dieses geöffnet werden kann, muss zunächst in den GPredict-Einstellungen unter *Interfaces* mindestens eine Schnittstelle zur Rotorensteuerung konfiguriert werden (siehe Abschnitt XXX).

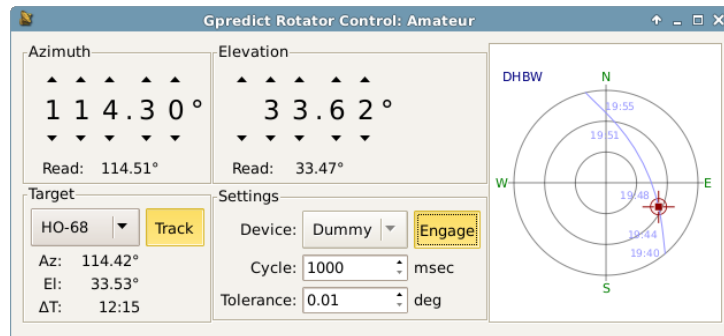


Abbildung 5.11: Rotorensteuerungs-Bedienfeld

Das Bedienfeld beinhaltet zum Einen eine Polaransicht, auf der die Spur und der aktuelle Ort des zu verfolgenden Satelliten (dargestellt durch ein Viereck) und die gegenwärtige Ausrichtung der Antenne (dargestellt durch ein Fadenkreuz) zu sehen ist. Zum Anderen sind folgende vier Bereiche verfügbar:

- **Azimuth:** In diesem Feld lässt sich die Ausrichtung der Antenne in Azimut-Richtung steuern, vorausgesetzt, dass die *Track*-Funktion nicht aktiviert ist. Am unteren Ende des Feldes wird unter *Read* der aktuelle Azimut-Winkel der Antenne angezeigt. Ist keine Verbindung zum Rotor aufgebaut, wird hier „- -“ angezeigt. Liegt ein Verbindungsproblem vor, erscheint „ERROR“.
- **Elevation:** In diesem Feld lässt sich die Ausrichtung der Antenne in Elevations-Richtung steuern, vorausgesetzt, dass die *Track*-Funktion nicht aktiviert ist. Am unteren Ende des Feldes wird unter *Read* der aktuelle Elevations-Winkel der Antenne angezeigt. Ist keine Verbindung zum Rotor aufgebaut, wird hier „- -“ angezeigt. Liegt ein Verbindungsproblem vor, erscheint „ERROR“.
- **Target:** Hier lässt sich der zu verfolgende Satellit auswählen. Es stehen hierbei nur die für das aktuelle Modul ausgewählten Satelliten zu Verfügung. Aktiviert man die Schaltfläche *Track*, wird der ausgewählte Satellit verfolgt. Unter dem Satellitennamen werden die jeweiligen Winkel in Echtzeit dargestellt und hinter ΔT wird die Zeit bis zum nächsten AOS bzw. LOS angezeigt.

- **Settings:** Hier lässt sich die in den GPredict-Einstellungen festgelegte Schnittstelle zur Kommunikation mit den Rotoren auswählen. Mit einem Klick auf die Schaltfläche *Engage* wird die Verbindung zu dieser Schnittstelle aufgebaut bzw. unterbrochen. Unter *Cycle* kann dabei der Zyklus eingestellt werden, in welchem Kommandos an die Rotor-Schnittstelle gesendet und Winkelwerte von dieser abgefragt werden. Ein sinnvoller Wert liegt hierbei zwischen zwei und fünf Sekunden. Bei *Tolerance* wird die tolerierte Differenz zwischen abgefragtem und eingestelltem Winkel eingetragen. Sobald diese überschritten wird, wird ein Kommando an die Rotor-Schnittstelle geschickt. Hierbei sollte sowohl die Winkelauflösung der Rotorensteuerung, als auch die Keulenbreite der Antenne berücksichtigt werden. Nach fünf aufeinanderfolgenden Fehlern bei der Kommunikation mit den Rotoren, wird die Verbindung automatisch unterbrochen.

In Tabelle 5.1 sind alle möglichen Kombinationen der Schaltflächen *Track* und *Engage* und deren Auswirkung beschrieben.

Track	Engage	Beschreibung
Inaktiv	Inaktiv	Es werden weder Kommandos an die Rotoren gesendet, noch wird die aktuelle Ausrichtung der Antenne ausgelesen. Die aktuellen Winkel des zu verfolgenden Satelliten werden nicht in die Winkelsteuerungs-Eingabefelder übertragen.
Aktiv	Inaktiv	Die aktuellen Winkel des zu verfolgenden Satelliten werden in die Winkelsteuerungs-Eingabefelder übertragen, es werden aber keine Kommandos an die Rotoren geschickt und die aktuelle Ausrichtung der Antenne wird nicht ausgelesen.
Aktiv	Aktiv	Die aktuellen Winkel des zu verfolgenden Satelliten werden in die Winkelsteuerungs-Eingabefelder übertragen und diese werden an die Rotoren geschickt. Die aktuelle Ausrichtung der Antenne wird ausgelesen.
Inaktiv	Aktiv	Die Winkel, die in den Winkelsteuerungs-Eingabefelder eingestellt sind, werden an die Rotoren gesendet und die aktuelle Ausrichtung der Antenne wird ausgelesen.

Tabelle 5.1: Betriebsmodi des *Antenna Control*-Bedienfelds, Quelle: [11]

Um dem Funkgerät die entsprechenden Up- und Downlink-Frequenzen inklusive Dopplerschiftkorrektur zu übermitteln, wird das *Radio Control*-Bedienfeld (siehe Abbildung 5.12) verwendet. Dieses kann nur geöffnet werden, wenn mindestens eine Funkgerät-Schnittstelle in den GPredict-Einstellungen unter *Interfaces* konfiguriert ist (siehe Abschnitt XXX).

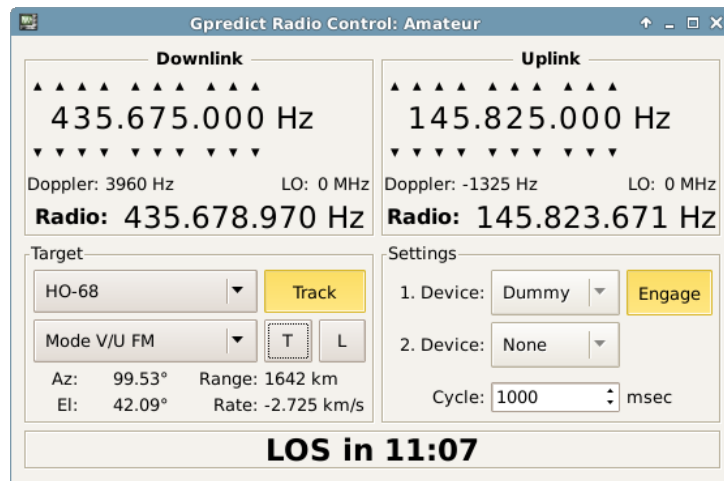


Abbildung 5.12: Funkgerät-Steuerung

Neben einer Anzeige die die Zeit bis zum nächsten AOS bzw. LOS darstellt (ganz unten im Bedienfeld in fettgedruckten Buchstaben) kann das *Radio Control*-Bedienfeld in folgende vier Bereiche untergliedert werden:

- **Downlink:** In diesem Bereich kann die Frequenzeinstellung für die Downlink-Frequenz vorgenommen werden. Außerdem ist die aktuelle Dopplerfrequenz und ein Feld namens *LO*, welches für eine Offsetfrequenz steht, die im Funkgerät eingestellt werden kann, steht. Die korrigierte Frequenz welche nach *Radio* angezeigt wird, setzt sich damit beispielsweise wie folgt zusammen: $f_{Radio} = f_{Satellite} + f_{Doppler} - f_{LO}$
- **Uplink:** Dieser Bereich beinhaltet die Frequenzeinstellungen für die Uplink-Frequenz und besitzt die gleichen Eigenschaften wie der Downlink-Bereich.
- **Target:** Hier kann der zu verfolgende Satellit und der zu verwendende Transponder eingestellt werden. Mit einem Klick auf *Track* wird der Dopplerschift des ausgewählten Satelliten bei den Frequenzeinstellungen im Downlink- und Uplink-Bereich korrigiert. Eine wichtige Rolle spielt hierbei die Schaltfläche *T* (*Tune*), da die Frequenzen des Transponders nur durch einen Klick auf diese in die Frequenzeinstellungen übertragen werden. Sollte der Transponder beispielsweise nur einen Downlink besitzen, wird auch nur die Frequenz im Downlink-Bereich eingestellt.

Mit der Schaltfläche *L* (*Lock*) lässt sich die Differenz der Downlink- und Uplink-Frequenz sperren, das heißt Änderungen an der einen wirken sich auch auf die andere Frequenz aus. Dies ist nicht für Transponder möglich, die nur einen Up- oder Downlink besitzen.

- **Settings:** In diesem Bereich lassen sich bis zu zwei Funkgeräte auswählen die in den GPredict-Einstellungen unter *Interfaces* eingerichtet wurden. Das obere Gerät stellt das Primäre dar und kann für Up- und Downlink verwendet werden. Das Untere kann als sekundäres Gerät verwendet werden, welches nur für den Uplink verwendet werden kann. Mit einem Klick auf *Engage* wird die Kommunikation zwischen GPredict und Funkgerät(en) aufgebaut. Hierbei kann im Feld neben *Cycle* der Zyklus eingetragen werden, nach welchem Kommandos an das Funkgerät geschickt werden sollen.

In Tabelle 5.2 ist eine Übersicht über die verschiedenen Betriebsmodi des *Radio Control*-Bedienfelds zu sehen.

Track	Engage	Beschreibung
Inaktiv	Inaktiv	Es wird keine Dopplerschiftkorrektur durchgeführt, keine Befehle an das Funkgerät gesendet und die aktuelle Frequenz des Funkgeräts nicht ausgelesen.
Aktiv	Inaktiv	Die Dopplerschiftkorrektur wird durchgeführt, es werden aber weder Befehle an das Funkgerät geschickt noch wird die aktuelle Frequenz ausgelesen.
Aktiv	Aktiv	Der Dopplerschift wird korrigiert und die eingestellte Frequenz wird zum Funkgerät geschickt. Die aktuelle Frequenz des Funkgeräts wird ausgelesen.
Inaktiv	Aktiv	Die Dopplerschiftkorrektur wird nicht ausgeführt. Die eingestellte Frequenz wird an das Funkgerät geschickt und die dort eingestellte Frequenz wird ausgelesen.

Tabelle 5.2: Betriebsmodi des *Radio Control*-Bedienfelds, Quelle: [11]

Wenn die *Engage*-Schaltfläche aktiviert ist, wird die Dopplerschiftkorrektur durchgeführt, egal ob der ausgewählte Satellit sichtbar ist oder nicht.

5.2.4 GPredict-Einstellungen

Um in die GPredict-Einstellungen zu gelangen, klickt man links oben unterhalb der Titelleiste auf *Edit* und anschließend auf *Preferences*. Nun öffnet sich ein Fenster bei dem standardmäßig die Erste der vier Kategorien zu sehen ist, die Kategorie *General* (siehe Abbildung 5.13). Nach einer Veränderung in den GPredict-Einstellungen muss GPredict zunächst neugestartet werden, damit diese Änderung wirksam wird. Überall wo die Schaltfläche *Reset* zu finden ist, können mit einem Klick auf diese, die Standard-Einstellungen wiederhergestellt werden. Bei den meisten Optionen erscheint ein kleines Informations-Fenster, sobald man mit dem Mauszeiger einige Sekunden über der entsprechenden Option verharret.

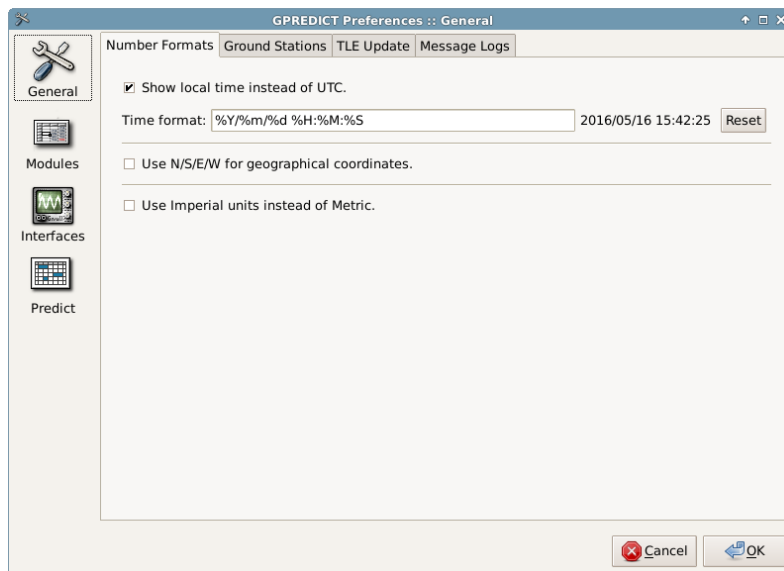


Abbildung 5.13: GPredict-Einstellungen, Kategorie *General*

Die Kategorie *General* ist in folgende vier Reiter untergliedert:

- **Number Formats:** Hier lassen sich Einstellungen zum Zeitformat, zu den geographischen Koordinaten und zu den Längen- und Geschwindigkeits-Einheiten vornehmen.
- **Ground Stations:** Unter diesem Reiter können beliebig viele Bodenstationen eingerichtet werden. Mindestens eine muss jedoch zu jeder Zeit vorhanden sein.
- **TLE Update:** Hier können Einstellungen zur Aktualisierung der Kepler-Elemente vorgenommen werden.
- **Message Logs:** Hier können Einstellungen bzgl. des GPredict-Protokolls vorgenommen werden. Unter *Log browser* im Menü *File* kann dieses dargestellt werden.

Zur Kategorie *Modules* gelangt man über zwei Wege. Der erste führt über die allgemeinen GPredict-Einstellungen, wie in diesem Abschnitt beschrieben. Der Zweite führt über das Modul Pop-Up Menü, wie auf Seite 31 beschrieben. Der Unterschied ist, dass bei den Modul-Einstellungen in den GPredict-Einstellungen die Standardwerte für alle Module eingestellt werden können und beim zweiten Weg nur die für das entsprechende Modul. Die Modul-Einstellungen sind in folgende sechs Reiter untergliedert:

- **Layout:** Bei diesem Reiter lässt sich konfigurieren, welche Ansichten in welcher Anordnung im GPredict-Fenster zu sehen sind. Hierbei kann aus verschiedenen Anordnungen ausgewählt werden oder eine völlig benutzerspezifische Anordnung erstellt werden (Hierfür wird an dieser Stelle auf [11] Seite 67f verwiesen).
- **Refresh Rates:** Hier lässt sich die Zeitspanne auswählen, nach welcher GPredict die Positions-Berechnungen periodisch für jeden Satelliten durchführt, der für das geöffnete Modul ausgewählt wurde. Außerdem lässt sich einstellen mit welchem ganzzahligen Vielfachen dieser Zeitspanne die einzelnen Ansichten aufgefrischt werden.
- **List View:** Hier lassen sich die Parameter der *List View* auswählen.
- **Map View:** Hier lassen sich Darstellungsoptionen für die *Map View* konfigurieren.
- **Polar View:** Hier lassen sich Darstellungsoptionen für die *Polar View* konfigurieren.
- **Single Sat View:** Hier lassen sich die Parameter der *Single Sat View* auswählen.

Die Kategorie *Interfaces* ist in zwei Reiter untergliedert, *Radios* und *Rotators*. Dort wird jeweils eine Liste der bereits eingerichteten Funkgeräte bzw. Rotoren angezeigt, welche standardmäßig leer ist. Die Anzahl der Geräte ist dabei nach oben nicht beschränkt. Die detaillierte Einrichtung ist in Abschnitt 5.4.5 und 5.4.6 genauer beschrieben.

Beim Öffnen der Kategorie *Predict* wird standardmäßig der Reiter *Pass Conditions* angezeigt. Unter diesem lassen sich folgende Parameter einstellen:

- **Minimum elevation:** Dieser Parameter gibt an, ab welcher Elevation von einem Durchlauf eines Satelliten ausgegangen wird. Das heißt, übersteigt die maximale Elevation diesen Wert, wird dieser Durchlauf in die *Upcoming Passes*, die *Pass Details* und die Ansicht *The sky at a glance* aufgenommen.
- **Number of passes to predict:** Anzahl der angezeigten zukünftigen Durchläufe.

- **Passes should occur within:** Dieser Parameter definiert den Zeitraum, in welchem zukünftige Durchläufe von GPredict berücksichtigt werden. So tritt entweder erst die Anzahl der angezeigten zukünftigen Durchläufe ein oder der Zeitraum in welchem zukünftige Durchläufe berücksichtigt werden sollen.
- **Time resolution:** Hier kann die Zeitauflösung eingetragen werden, mit der der nächste Durchlauf in den *Pass Details* dargestellt wird. Je geringer die Auflösung, desto mehr Einträge werden angezeigt.
- **Number of entries:** Mit diesem Parameter kann die Anzahl der Einträge des nächsten Durchlaufs in den *Pass Details* eingestellt werden. Hierbei ist zu beachten, dass diese Einstellung eine höhere Priorität als die Zeitauflösung besitzt.
- **Twilight threshold:** Dieser Parameter gibt an, ab welcher Elevation ein Satellit als sichtbar gilt. Für die Satellitenverfolgung spielt dieser Parameter keine Rolle.

Unter den Reitern *Multiple Passes* und *Single Pass* lassen sich die Parameter auswählen, die in den Ansichten *Upcoming Passes* und *Pass Details* dargestellt werden, wohingegen beim Reiter *Sky at a Glance* Darstellungsoptionen für die Funktion *The sky at a glance* eingestellt werden können.

Eine detailliertere Beschreibung der GPredict-Einstellungen kann aus [11] Seite 23ff entnommen werden.

5.3 HamLib-Programmierschnittstelle

5.3.1 Übersicht

Da es keinen einheitlichen Kommunikationsstandard für die zahlreichen Funkgeräte und Rotoren unterschiedlicher Hersteller gibt, ist für die Verwendung von GPredict eine applikationsspezifische Programmierschnittstelle oder auch Application Programming Interface (API) erforderlich. Mit den *Ham Radio Control Libraries* (englisch für Amateurfunk-Kontrollbibliotheken), kurz HamLib, steht dem Benutzer eine solche API zur Verfügung. HamLib ist unter der GPL lizenziert und ist unter [13] in der Version 3.0.1 als Download für Linux und Windows kostenlos verfügbar. Wie in Abbildung 5.14 zu sehen, ermöglicht HamLib einer Software wie GPredict die Kommunikation mit verschiedenen Funkgeräten

und Rotoren, in dem es für jedes dieser Geräte einen eigenen Treiber bzw. ein eigenes Backend (BE) zur Verfügung stellt.

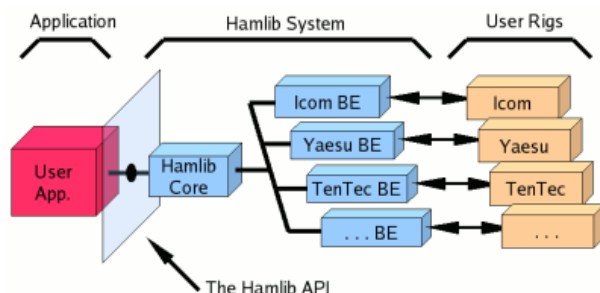


Abbildung 5.14: HamLib Design, Quelle: [14]

Dabei verwendet man entweder den Quellcode um eine benutzerspezifische Anwendung zu erstellen oder man greift auf die bereits fertig kompilierten Programme zurück, welche im Folgenden aufgelistet sind:

- **rigctl**: Ein Kommandozeilenprogramm, mit welchem man Befehle über die Kommandozeile an das Funkgerät senden kann. (Unter Windows: *rigctl.exe*)
- **rotctl**: Ein Kommandozeilenprogramm, mit welchem man Befehle über die Kommandozeile an die Antennenrotoren senden kann. (Unter Windows: *rotctl.exe*)
- **rigctld**: Ein Kommandozeilenprogramm, mit welchem man Befehle über das TCP/IP-Protokoll an das Funkgerät senden kann (Unter Windows: *rigctld.exe*)
- **rotctld**: Ein Kommandozeilenprogramm, mit welchem man Befehle über das TCP/IP-Protokoll an die Antennenrotoren senden kann (Unter Windows: *rotctld.exe*)

Hierbei steht „rot“ für „Rotator“ (deutsch: Rotor), „rig“ für „Rig“ (deutsch: Amateurfunkgerät) und das „d am Ende von *rigctld* und *rotctld* für „Deamon“ (deutsch: Hintergrundprozess).

5.3.2 Parameter-Konfiguration

Für die erfolgreiche Nutzung der oben genannten Programme müssen einige Optionen und Parameter berücksichtigt werden. Hierfür ist es notwendig die erforderlichen Informationen zu den Rotoren und zum Funkgerät zu kennen. Um eine Übersicht der möglichen Optionen und Befehle zu erhalten, kann das jeweilige Programm einfach mit der Option **-h** oder **-help** ausgeführt werden (siehe Anhang XXX und YYY).

Die erste zentrale Information ist die *Hamlib ID*. Hierfür gibt man folgenden Befehl in die Kommandozeile ein und erhält eine Liste mit allen unterstützten Funkgeräten (für *rigctl.exe* und *rotctl(d).exe* analog durchführbar):

```
rigctl.exe -l
```

Im Folgenden sind die beiden Zeilen dargestellt, die für die in diesem Projekt verwendeten Funkgerät und Rotoren relevant sind. Hierbei steht der erste Eintrag für die *Hamlib ID*, der Zweite für den Hersteller, der Dritte für die Version und der Vierte für den Test-Status:

368	Icom	IC-9100	0.7	Untested
601	Yaesu	GS-232A	0.3	Beta

Als Nächstes muss die Schnittstelle zum Funkgerät bzw. zur Rotorensteuerung angegeben werden. Bei der im Rahmen diesen Projekts verwendeten Software (siehe Abschnitt 5.4.1) werden hierfür serielle Schnittstellen verwendet. Wie diese konfiguriert werden, ist im folgenden Auszug aus der Hilfe, die bei Eingabe der Option *-h* oder *--help* erscheint (für *rigctl.exe*), zu sehen:

<i>-r, -rig-file=DEVICE</i>	select radio model number. See model list
<i>-s, -serial-speed=BAUD</i>	set serial speed of the serial port
<i>-C, -set-conf=PARAM=VAL</i>	set config parameters

Um eine Liste der möglichen Konfigurationsparameter (*config parameters*) auszugeben, führt man *rigctl(d).exe* bzw. *rotctl(d).exe* mit der Option *-L* oder *--show-conf* aus. Die genauen Werte sind in Abschnitt 5.4.1 aufgeführt.

Im Testbetrieb ist die Ausgabe von Statusmeldungen (z.B. Warnungen oder Fehler) oft sehr hilfreich. Hierfür lässt sich mit der Option *-v* die Intensität der Ausgabe einstellen. Bei *-v* werden nur grobe Fehler ausgegeben, wohingegen bei *-vvvvv* alle verfügbaren Statusmeldungen der API ausgegeben werden. Bei Weglassen der Option erfolgt keine Ausgabe. Mit diesem Wissen sieht die Ausführung des Programms *rotctl.exe* im Rahmen dieses Projekts wie folgt aus:

```
rotctl.exe -vvvv -m 601 -r COM10 -s 9600 -C stop_bits=1,data_bits=8
```

Da beim Funkgerät IC-9100 das Icom-spezifische Protokoll CI-V (das „V“ steht hierbei für die römische Zahl) für die Fernsteuerung verwendet wird, kommt bei der Verwendung von *rigctl.exe* noch die CI-V-Adresse als Parameter hinzu. Somit sieht die Ausführung des Programms *rigctl.exe* wie folgt aus:

```
rigctl.exe -vvvv -m 368 -r COM5 -c 0x7C -s 19200 -C stop_bits=1,data_bits=8
```

5.3.3 Verwendung

Nachdem das Programm *rigctl.exe* bzw. *rotctl.exe* mit den unter Abschnitt 5.3.2 dargestellten Parametern gestartet wurde, hat man die Möglichkeit Befehle an das Funkgerät bzw. an die Rotorensteuerung zu senden. Eine Übersicht der Befehle lässt sich der Ausgabe der Programmausführung mit dem Parameter *-h* bzw. *-help* entnehmen.

Da nicht jedes Funkgerät bzw. nicht jede Rotorensteuerung alle möglichen Befehle unterstützt, ist es wissenswert welche Befehle von den in diesem Projekt verwendeten Geräten unterstützt werden. Unter Angabe der *Hamlib ID* und der Option *-u* bzw. *-dump-caps* kann eine Übersicht der *capabilities* (Fähigkeiten), z.B. der unterstützten Befehle oder der Standard-Parameter ausgegeben werden (siehe Anhang XXX und YYY):

```
rigctl.exe -m 368 -u  
rotctl.exe -m 601 -u
```

Um die Verwendung dieser Kommandozeilenprogramme zu vereinfachen und um gleichzeitig die notwendige Konfiguration festzuhalten, wurde für die Programme *rigctl*, *rigctld*, *rotctl* und *rotctld* jeweils ein Batch-Skript erstellt (siehe Anhang A, B, C und D). Bei der Verwendung unter Linux können diese ohne großen Aufwand in Bash-Skripte umgeschrieben werden.

5.4 Inbetriebnahme unter Windows

5.4.1 Zusätzliche Software

Wie in Abschnitt XXX zu sehen, werden sowohl das Funkgerät als auch die Rotorensteuerung über das Hochschul-Netzwerk angesteuert. Für die Kommunikation mit den notwendigen Hardwarekomponenten (Banana Pi und *NetCOM 211*) stehen bereits Programme für Windows zur Verfügung. In Abbildung 5.15 ist eine Übersicht zu sehen, wie GPredict mit dem Funkgerät und den Rotoren kommuniziert. Hierbei sind reine Software-Komponenten in blau und Hardware-Komponenten in rot dargestellt. Der Banana Pi liegt zwar als Hardware-Komponente vor, jedoch ist die Software selbst erstellt worden und kann jederzeit angepasst werden, deshalb ist er in blau-rot dargestellt.

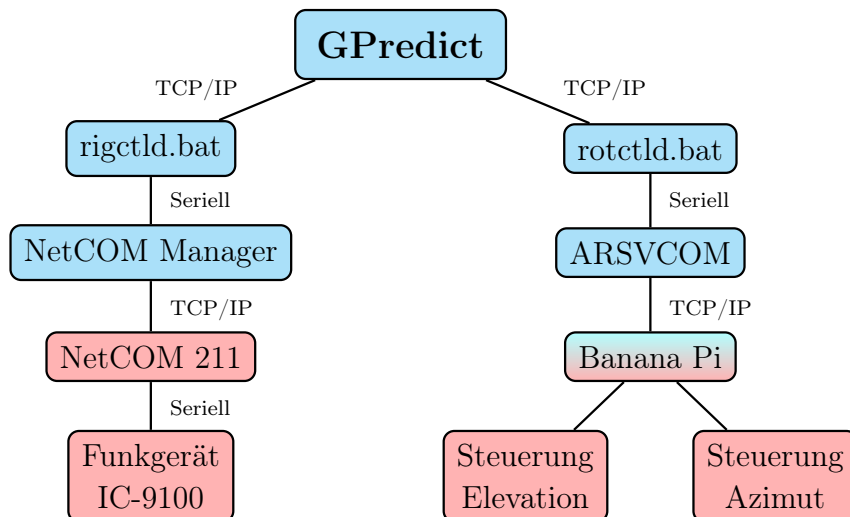


Abbildung 5.15: Kommunikation zwischen Hard- und Software

Hierbei kommuniziert GPredict über eine IP-basierte Schnittstelle nur mit den HamLib-Programmen *rigctl(d).exe* und *rotctl(d).exe*. Diese werden unter Berücksichtigung der notwendigen Parameter als Batch-Skripte ausgeführt und müssen solange eine Kommunikation zwischen GPredict und Funkgeräten bzw. Rotoren bestehen soll, aktiv sein.

Die Software *NetCOM Manager* stellt eine virtuelle serielle Schnittstelle (COM5) zur Verfügung, auf die von *rigctl(d).exe* zugegriffen werden kann. Das *NetCOM 211*-Modul ist wiederum per serieller Schnittstelle mit dem Funkgerät verbunden und ermöglicht die Kommunikation über das Netzwerk mit der Software *NetCOM Manager*. Somit muss man sich lediglich im Hochschul-Netzwerk befinden, um mit dem Funkgerät kommunizieren zu

können. Wird hierbei die Möglichkeit einer VPN-Verbindung wahrgenommen, kann quasi von jedem beliebigen Ort mit Internetanbindung mit dem Funkgerät kommuniziert werden.

Für die Kommunikation mit den Rotoren, wird auf die Software *ARSVCOM* zurückgegriffen. Diese emuliert eine Rotorensteuerung vom Typ *Yaesu RS232A*, wofür ebenfalls ein virtueller, beliebig einstellbarer COM-Port erstellt wird. Der COM-Port und der Rotorensteuerungs-Typ werden als Parameter bei *rotctl(d).exe* angegeben. Die Software *ARSVCOM* kommuniziert über das Netzwerk mit dem Banana Pi, an welchem wiederum die Steuerungseinheiten des Azimut- und Elevations-Rotors angeschlossen sind (siehe Abschnitt XXX).

5.4.2 Test der Bedienfelder

Um sich mit den Bedienfeldern zur Funkgerät- bzw. Rotorensteuerung vertraut zu machen, kann auf zwei sogenannte *Dummy*-Schnittstellen zugegriffen werden. Diese sind sowohl bei *rigctl.exe*, als auch bei *rotctl.exe* unter der *Hamlib ID* 1 erreichbar. In Abbildung 5.16 ist eine beispielhafte Konfiguration zu sehen, die in den *Interface*-Einstellungen zur Kommunikation mit den *Dummy*-Schnittstellen vorgenommen werden kann.

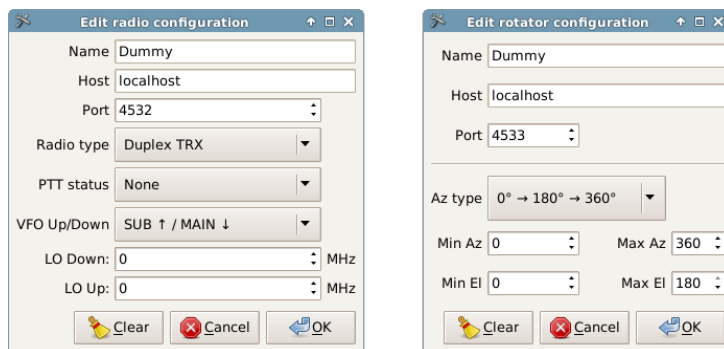


Abbildung 5.16: Schnittstellenkonfiguration für das Dummy-Interface

Für diesen Test wurden für die Programme *rigctld.exe* und *rotctld.exe* ebenfalls eigene Batch-Skripte erstellt (siehe Anhang E und F). Die hierfür verwendeten Optionen unterscheiden sich lediglich im Port, für *rigctld.exe* ist standardmäßig der Port 4532 und für *rotctld.exe* der Port 4533 vorgesehen. Dieser kann jedoch nach Belieben angepasst werden.

Sobald ein Fehler bei der Kommunikation zwischen den Bedienfeldern und den HamLib-Programmen fünf Mal in Folge aufgetreten ist, wird aus Sicherheitsgründen die Kommunikation automatisch abgebrochen.

5.4.3 Test der Kommunikation zum Funkgerät

Um die Kommunikation zwischen HamLib und dem Funkgerät zu testen, werden zunächst einzelne Kommandos mit Hilfe des Skripts *rigctl.bat* an das Funkgerät geschickt. Somit kann sichergestellt werden, ob das HamLib-Backend für das Funkgerät IC-9100 richtig implementiert wurde. Hierbei kann nicht davon ausgegangen werden, dass ein reibungsloser Betrieb möglich ist, da als Test-Status des HamLib-Backends für das Funkgerät IC-9100 „Untested“ angegeben ist (siehe Seite 40).

In der Betriebsanleitung des Funkgeräts IC-9100 (siehe [4]) ist auf Seite 183ff eine Übersicht der Befehlsstruktur und der zulässigen Kommandos des ICOM-spezifischen Protokolls CI-V dargestellt. Zur Kontrolle, ob die HamLib-Befehle auch den richtigen CI-V-Kommandos entsprechen, wird mit dem Windows-Programm *Serial Port Monitor* (hierfür kann auch ein anderes Programm verwendet werden) der serielle Port COM5 abgehört, während über *rigctl.bat* verschiedene Befehle an das Funkgerät geschickt werden.

Bei folgende Befehlen konnte eine Übereinstimmung der von HamLib verwendeten mit den in [4] zu findenden CI-V Befehlen und damit eine einwandfreie Funktionalität festgestellt werden:

- F: `set_freq` (Setzen der Frequenz des *Main*-Bandes)
- f: `get_freq` (Auslesen der Frequenz des *Main*-Bandes)
- M: `set_mode ()`
- m: `get_mode ()`

Bei einigen Befehlen wurde festgestellt, dass die CI-V-Kommandos, die von HamLib verwendet werden, nicht mit den tatsächlichen Befehlen aus [4] übereinstimmen. Da es mit dem fertig kompilierten Programm *rigctl(d).exe* keine Möglichkeit gibt einen „rohen“ Befehl zu versenden, ist es somit nicht möglich, die Funktionalität der betroffenen Befehle ohne Eingriff in den Quellcode von HamLib zu erreichen. In Tabelle 5.3 sind die betroffenen Befehle aufgelistet, wobei die Ausschnitte, die nicht übereinstimmen, rot markiert sind.

HamLib Befehl (rigctl)	CI-V (Verwendet)	CI-V (Tatsächlich)
U SATMODE 1	fe fe 7c e0 1a 07 01 fd	fe fe 7c 00 16 5a 01 fd

Tabelle 5.3: Fehlerhafte CI-V Kommandos von *rigctl(d).exe*

5.4.4 Test der Kommunikation zu den Rotoren

Beim Test der Bedienelemente kann der Ausgabe des Skripts *rotctld-dummy.bat* bei Verwendung der höchsten Ausgabeintensität *-vvvvv* entnommen werden, dass lediglich die Befehle *set_pos* und *get_pos* beim Verfolgen eines Satelliten verwendet werden. Diese beiden Befehle konnten mit dem Skript *rotctl.bat* erfolgreich für die Steuerung der beiden Rotoren getestet werden. Grundsätzlich sind bei der Steuerung der Rotoren einige Aspekte zu beachten. Beispielsweise, dass die Rotorgeschwindigkeit reduziert wird, sobald der Ist-Winkel in gewisser Nähe zum Soll-Winkel. Diese Punkte werden jedoch alle entweder von den Steuerungsmodulen oder von der Software *ARSVCOM* übernommen, sodass in GPredict lediglich die jeweiligen Winkel ausgelesen und gesetzt werden müssen.

5.4.5 Konfiguration des Funkgeräts

Dem Abschnitt XXX kann entnommen werden, dass das Funkgerät IC-9100 vollduplexfähig ist und im *Sub*-Band sendet und im *Main*-Band empfängt. Diese Informationen können in den *Interface*-Einstellungen wie in Abbildung 5.17 zu sehen, eingestellt werden.

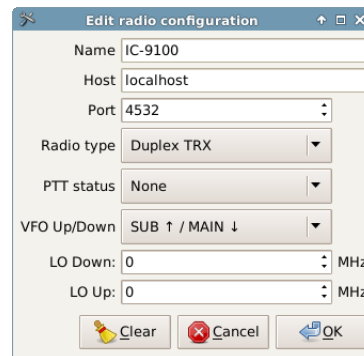


Abbildung 5.17: Funkgerät-Konfiguration

Bei *Host* muss die IP-Adresse angegeben werden unter der aktive *rigctld.exe*-Prozess zu erreichen ist. Hier ist standardmäßig *localhost* eingetragen, was für den Computer steht, auf dem GPredict ausgeführt wird. Da im Rahmen dieses Projekts GPredict und *rigctld.bat* immer auf dem gleichen Computer ausgeführt werden, wird sowohl unter GPredict als auch beim Skript *rigctld.bat* die IP-Adresse 127.0.0.1 (*localhost*) eingestellt. Hierbei ist darauf zu achten, dass die Verwendung der Zeichenkette *localhost* unter Windows bei der Verwendung von *rigctld.exe* nicht funktioniert. Die Angabe des Ports der *Interface*-Einstellungen muss ebenfalls dem in *rigctld.bat* eingestellten Port entsprechen. Die Felder *LO Down* und *LO Up* ermöglichen die Berücksichtigung eines Transverters, welcher zur Veränderung der Empfangs- oder Sendefrequenz eingesetzt wird. Somit kann beispielsweise mit einem 28 MHz-Empfänger ein Signal von einem Satellit der mit 144 MHz sendet, empfangen werden, in dem $144 \text{ MHz} - 28 \text{ MHz} = 116 \text{ MHz}$ bei *LO Down* eingetragen wird. Beim Feld *PTT status* wird *None* eingetragen, da bei vollduplexfähigen Funkgeräten die Abfrage des Push-to-Talk (PTT)-Status nicht nötig ist (siehe [11] Seite 58).

5.4.6 Konfiguration der Rotoren

Bei der Konfiguration der Rotorensteuerung ist die Angabe der IP-Adresse und des Ports analog wie bei der Funkgerät-Konfiguration vorzunehmen, bis auf die Ausnahme, dass hier ein anderer Port eingestellt werden muss (siehe Abbildung 5.18).

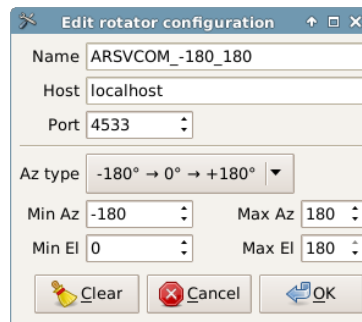


Abbildung 5.18: Konfiguration der Rotorensteuerung

Wie in Abschnitt XXX zu sehen, besitzt der Azimut-Rotor seinen Anschlag bei 180° (Süden), das heißt er lässt sich von 0° (Norden) aus jeweils um 180° in Richtung Süden drehen. Deshalb muss beim Feld *Az type* der Bereich von -180° bis $+180^\circ$ ausgewählt werden. Der Elevations-Rotor lässt sich von 0° bis 180° steuern, dementsprechend wird die Einstellung bei *Min El* und *Max El* gewählt.

5.4.7 Probelauf und Testergebnisse

Probleme:

- Wenn die Frequenzen im Main- und Sub-Band vertauscht werden müssten, bricht die Kommunikation ab. Nach manuellem Vertauschen ist Steuerung möglich.
- Nicht möglich den Satmodus zu aktivieren (muss manuell aktiviert werden)
- Rotor-Flip: Möglich bei -180 bis 180 ; Nicht möglich bei 0 bis 360

6 Zusammenfassung und Ausblick

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean porttitor mi purus, sit amet efficitur velit semper sit amet. Mauris et pulvinar nunc, id maximus metus. Suspendisse convallis sapien nisi, sed maximus quam gravida eu. Duis faucibus elit non nunc posuere dignissim. Sed ullamcorper fringilla felis ac mollis. Nunc blandit tristique auctor. Praesent elementum dictum nulla et vulputate. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer sed rutrum lacus. Morbi tincidunt dui at augue molestie rhoncus. Curabitur sagittis sed sapien quis vestibulum. Cras dictum sem quam, nec tincidunt augue tempor et. Morbi consectetur, dui id scelerisque consequat, velit tortor gravida eros, ac hendrerit massa magna sit amet lacus. Quisque mattis nulla diam, sed efficitur mi fermentum vitae. Nam vestibulum iaculis rhoncus.

Donec feugiat augue leo, a malesuada tortor laoreet eu. Praesent dictum tortor eu egestas sodales. Maecenas rhoncus pretium leo ut sodales. Praesent blandit sit amet ante posuere malesuada. Vestibulum egestas sit amet ex et placerat. In sed dapibus sem, placerat interdum mi. Sed eget eros ante. Duis varius molestie eleifend. Suspendisse eu aliquet velit. Nulla facilisi. Morbi eu sapien in odio viverra luctus.

Abbildungsverzeichnis

2.1	V-Modell, Quelle: [1]	2
3.1	Cubesat, Quelle: [3]	5
4.1	Bahnelemente	7
4.2	Ellipse	9
4.3	Kegelschnitt	11
4.4	Kegelschnitt	12
4.5	Bahnelemente	15
4.6	Kepler-Gleichung	19
5.1	Prinzip eines Satellitenverfolgungsprogramms, Quelle: [11]	25
5.2	Standardoberfläche von GPredict	26
5.3	<i>List View</i>	28
5.4	<i>Upcoming Passes View</i>	28
5.5	<i>Satellite Info</i>	29
5.6	<i>Pass Details</i>	29
5.7	<i>Upcoming Passes</i>	30
5.8	<i>The sky at a glance</i>	30
5.9	<i>Time Controller</i>	31
5.10	<i>Edit Module</i>	31
5.11	Rotorensteuerungs-Bedienfeld	32
5.12	Funkgerät-Steuerung	34
5.13	GPredict-Einstellungen, Kategorie <i>General</i>	36
5.14	HamLib Design, Quelle: [14]	39
5.15	Kommunikation zwischen Hard- und Software	42
5.16	Schnittstellenkonfiguration für das Dummy-Interface	43
5.17	Funkgerät-Konfiguration	46
5.18	Konfiguration der Rotorensteuerung	47

Alle nicht eigens nachgewiesenen Abbildungen stammen von den Autoren.

Tabellenverzeichnis

5.1	Betriebsmodi des <i>Antenna Control</i> -Bedienfelds, Quelle: [11]	33
5.2	Betriebsmodi des <i>Radio Control</i> -Bedienfelds, Quelle: [11]	35
5.3	Fehlerhafte CI-V Kommandos von <i>rigctl(d).exe</i>	45

Literatur- und Quellenverzeichnis

- [1] P. D. I. habil. Klaus-Peter Fährnich, *Softwaretechnik - Vorgehensmodelle*, Deutsch. Universität Leipzig, Institut für Informatik, Betriebliche Informationssysteme, 2009.
- [2] W. F. Inc. (2015). Amateurfunksatellit, Adresse: <https://de.wikipedia.org/wiki/Amateurfunksatellit> (besucht am 24.05.2016).
- [3] C. Lab. (2016). ULA Releases Application for University CubeSat Competition, Adresse: <http://www.cubesat.org/> (besucht am 24.05.2016).
- [4] *IC-9100 Instruction Manual*. Adresse: http://www.icom-australia.com/products/amateur/ic-9100/Amateur_IC-9100_Instruction_Manual.pdf (besucht am 24.05.2016).
- [5] W. Steiner und M. Schlagerl, *Raumflugmechanik*, Deutsch, 1. Aufl. Berlin Heidelberg: Springer Verlag, 2004.
- [6] W. F. Inc. (2016). Johannes Kepler, Adresse: https://en.wikipedia.org/wiki/Johannes_Kepler (besucht am 05.05.2016).
- [7] —, (2016). Bahnelement, Adresse: <https://de.wikipedia.org/wiki/Bahnelement> (besucht am 01.05.2016).
- [8] W. Ley, K. Wittmann, W. Hallmann u. a., *Handbuch der Raumfahrttechnik*, Deutsch, 3. Aufl. München: Carl Hanser Verlag, 2008.
- [9] W. F. Inc. (2016). Frühlingspunkt, Adresse: <https://de.wikipedia.org/wiki/Fr%C3%BChlingspunkt> (besucht am 25.05.2016).
- [10] —, (2016). Kepler-Gleichung, Adresse: <https://de.wikipedia.org/wiki/Kepler-Gleichung> (besucht am 27.05.2016).
- [11] *GPredict User Manual*. Adresse: <https://sourceforge.net/projects/gpredict/files/Gpredict/1.3/gpredict-user-manual-1.3.pdf/download> (besucht am 01.05.2016).
- [12] *Sourceforge.net: GPredict 1.3*. Adresse: <https://sourceforge.net/projects/gpredict/files/Gpredict/1.3/> (besucht am 01.05.2016).
- [13] *HamLib Download*. Adresse: <https://sourceforge.net/projects/hamlib/> (besucht am 31.05.2016).
- [14] *HamLib Manual*. Adresse: <http://hamlib.sourceforge.net/manuals/hamlib.html> (besucht am 13.05.2016).

A Batch-Skript: *rigctld.bat*

File: /home/hannes/Dropbox/02_Sarah...ib/Skripte/Windows/rigctld.bat

Page 1 of 1

```
@rem Dies ist ein Skript zur Ausführung von rigctld.exe.
@title Batch Skript: rigctld.exe
@echo off

echo =====
echo #
echo #          Batch Skript zur Ausführung von          #
echo #
echo #          rigctld.exe                                #
echo #
echo =====
echo.
echo # Ausführung von rigctld.exe mit folgenden Parametern:
echo.
set mycmd=rigctld.exe -vvvv -m 368 -r COM5 -c 0x7C -s 19200 -C stop_bits=1,data_bits=8,write_delay=5,po
st_write_delay=5,retry=3 -T 127.0.0.1 -t 4532 &
echo %mycmd%
echo.
echo =====
echo.
echo # Ausgabe von rigctld.exe:
echo.
%mycmd%
```

— !!! —

AKTUALISIEREN

— !!! —

B Batch-Skript: *rigctl.bat*

File: /home/hannes/Dropbox/02_Sarah...Lib/Skripte/Windows/rigctl.bat

Page 1 of 1

```
@rem Dies ist ein Skript zur Ausführung von rigctl.exe.
@title Batch Skript: rigctl.exe
@echo off

echo =====
echo #
echo #           Batch Skript zur Ausführung von           #
echo #                                           #
echo #           rigctl.exe                           #
echo #                                           #
echo =====
echo.
echo # Ausführung von rigctl.exe mit folgenden Parametern:
echo.
set mycmd=rigctl.exe -vvvv -m 368 -r COM5 -c 0x7C -s 19200 -C stop_bits=1,data_bits=8,post_write_delay=40
echo %mycmd%
echo.
echo =====
echo.
echo # Ausgabe von rigctl.exe:
echo.
%mycmd%
```

— !!! —

AKTUALISIEREN

— !!! —

C Batch-Skript: *rotctld.bat*

File: /home/hannes/Dropbox/02_Sarah...ib/Skripte/Windows/rotctld.bat

Page 1 of 1

```
@rem Dies ist ein Skript zur Ausführung von rotctld.exe.
@title Batch Skript: rotctld.exe
@echo off

echo =====
echo #                                     #
echo #           Batch Skript zur Ausfuehrung von           #
echo #                                     #
echo #           rotctld.exe                               #
echo #                                     #
echo =====
echo.
echo # Ausfuehrung von rotctld.exe mit folgenden Parametern:
echo.
set mycmd=rotctld.exe -vvvvv -m 601 -r COM10 -T 127.0.0.1 -t 4533 &
echo %mycmd%
echo.
echo =====
echo.
echo # Ausgabe von rotctld.exe:
echo.
%mycmd%
```

— !!! —

AKTUALISIEREN

— !!! —

D Batch-Skript: *rotctl.bat*

File: /home/hannes/Dropbox/02_Sarah...Lib/Skripte/Windows/rotctl.bat

Page 1 of 1

```
@rem Dies ist ein Skript zur Ausführung von rotctl.exe.
@title Batch Skript: rotctl.exe
@echo off

echo =====
echo #
echo #           Batch Skript zur Ausführung von           #
echo #                                           #
echo #           rotctl.exe                               #
echo #                                           #
echo =====
echo.
echo # Ausführung von rotctld.exe mit folgenden Parametern:
echo.
set mycmd=rotctl.exe -vvvv -m 601 -r COM10 -s 9600 -C stop_bits=1,data_bits=8,min_az=-180,max_az=180,min_el=0,max_el=180
echo %mycmd%
echo.
echo =====
echo.
echo # Ausgabe von rotctld.exe:
echo.
%mycmd%
```

— !!! —

AKTUALISIEREN

— !!! —

E Batch-Skript: *rigctld-dummy.bat*

File: /home/hannes/Dropbox/02_Sarah...ipte/Windows/rigctld-dummy.bat

Page 1 of 1

```
@rem Dies ist ein Skript zur Ausführung von rigctld.exe mit der Dummy-Schnittstelle
@title Batch Skript: rigctld.exe - Dummy
@echo off

echo =====
echo #
echo #          Batch Skript zur Ausfuehrung von          #
echo #
echo #          rigctld.exe - Dummy                      #
echo #
echo =====
echo.
echo # Ausfuehrung von rigctld.exe mit folgenden Parametern:
echo.
set mycmd=rigctld.exe -vvvv -m 1 -T 127.0.0.1 -t 4532 &
echo %mycmd%
echo.
echo =====
echo.
echo # Ausgabe von rigctld.exe:
echo.
%mycmd%
```

— !!! —

AKTUALISIEREN

— !!! —

F Batch-Skript: *rotctld-dummy.bat*

File: /home/hannes/Dropbox/02_Sarah...ipte/Windows/rotctld-dummy.bat

Page 1 of 1

@rem Dies ist ein Skript zur Ausführung von rotctld.exe mit der Dummy-Schnittstelle.

@**title** Batch Skript: rotctld.exe - Dummy

@**echo** off

```
echo =====  
echo #                                     #  
echo #           Batch Skript zur Ausfuehrung von           #  
echo #                                     #  
echo #           rotctld.exe - Dummy                         #  
echo #                                     #  
echo =====  
echo .  
echo # Ausfuehrung von rotctld.exe mit folgenden Parametern:  
echo .  
set mycmd=rotctld.exe -vvvv -m 1 -T 127.0.0.1 -t 4533 &  
echo %mycmd%  
echo .  
echo =====  
echo .  
echo # Ausgabe von rotctld.exe:  
echo .  
%mycmd%
```

— !!! —

AKTUALISIEREN

— !!! —