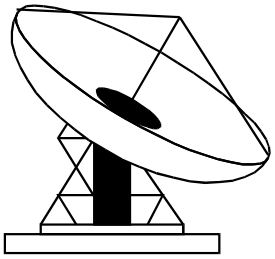
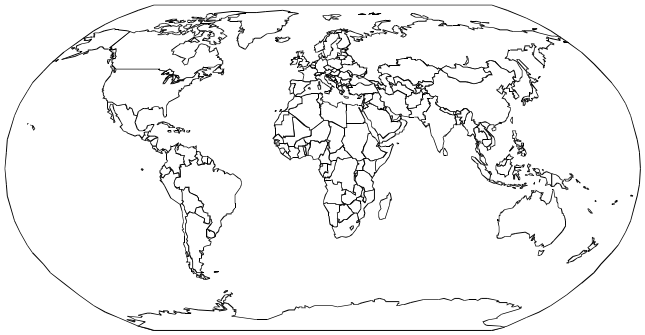
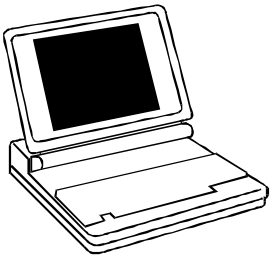
Diplomarbeit

Sommersemester 1998

Auf µC 80C517A basierende Erweiterung zum GPS-OEM-Modul der Fa. Rockwell







Ingenieurschule Zürich
Technikerschule

Abteilung	: Elektronik	
Klasse	: 6 N	
Aufgabenstellung und	:	
Betreuung	: Dipl. El. Ing. ETH J. Bazali	
Ausgabedatum	: Di. 21.04.1998	16^{45}
Abgabedatum	: Mo. 10.08.1998	18^{00}
Diplomand	: S. Schallenberger	

Allgemeines 1

Aufgabenstellung 2

Jupiter GPS-Receiver 3

Konzept 4

Hardware 5

Software 6

Inbetriebnahme 7

Zusammenfassung 8

Literaturverzeichnis 9

Anhang 10

Inhaltsverzeichnis

1. Allgemeines

1.1 Einleitung **1-1**
1.2 Kurze Einführung in das NAVSTAR-GPS **1-1**
1.3 Prototypen-Baugruppe MCB-517AC **1-2**
1.3.1 Der 80C517A 1-2
1.3.1.1 Features 1-2
1.4 “Jupiter”-GPS-Receiver-Board **1-3**
1.4.1 GPS-Adaptor-Card 1-3
1.5 Bestimmungen für die Diplomarbeit **1-4**

2. Aufgabenstellung

2.1 Original **2-1**
2.2 Detaillierung **2-2**
2.2.1 Handbuch zu Jupiter-GPS-Receiver 2-2
2.2.2 Hardware 2-2
2.2.2.1 Aufbau 2-2
2.2.2.2 Benötigte Versorgungs-Spannungen 2-3
2.2.2.3 Prototypen-Board MCB-517AC 2-3
2.2.2.4 LCD-Display mit 4x20 Zeichen 2-3
2.2.2.5 Tastatur mit 4x4 Tasten 2-3
2.2.2.6 Adressdecodierung 2-4
2.2.3 Software 2-4
2.2.4 Ziel 2-4

3. Handbuch zu Jupiter-GPS-Receiver

3.1 LABMON Evaluation-Software **3-3**
3.1.1 Einleitung 3-3
3.1.2 LABMON installieren 3-4
3.1.3 LABMON unter DOS installieren 3-4
3.1.4 LABMON unter Windows installieren 3-4
3.1.5 LABMON konfigurieren 3-5
3.1.5.1 Die Datei LABMON.CFG 3-5
3.1.5.2 Die Datei LABMON.INI 3-7
3.1.6 LABMON starten 3-8
3.1.7 LABMON unter DOS starten 3-8
3.1.8 LABMON unter Windows 3.x aus einer DOS-Box starten 3-8

3.1.9 LABMON unter Windows 3.x starten ... 3-8
3.1.10 LABMON beenden ... 3-9
3.1.11 LABMON-Bildschirm ... 3-9
3.1.12 LABMON Menü-Tasten ... 3-11
3.1.12.1 <Shift>+Funktionstaste ... 3-12
3.1.12.2 <Ctrl>+Funktionstaste ... 3-13
3.1.12.3 <Alt>+Funktionstaste ... 3-14
3.1.13 Features, Hinweise und Bemerkungen ... 3-15

3.2 Development Kit Setup & Operation ... 3-17
3.2.1 Einleitung ... 3-17
3.2.2 Mitgelieferte Geräte ... 3-17
3.2.3 Benötigte Geräte ... 3-18
3.2.4 Beschreibung ... 3-18
3.2.5 Konfiguration ... 3-20
3.2.5.1 DIP-Schalter ... 3-21
3.2.5.2 Jumper-Blocks ... 3-22

3.3 Zodiac-Family Designer's Guide ... 3-25
3.3.1 Einleitung ... 3-25
3.3.2 Kurze Produkte-Übersicht ... 3-25
3.3.2.1 Features ... 3-26
3.3.3 GPS-System-Beschreibung ... 3-27
3.3.3.1 Space Segment (Raumsegment) ... 3-28
3.3.3.2 Control Segment (Kontrollstationen) ... 3-29
3.3.3.3 User Segment (Anwenderbereich) ... 3-29
3.3.3.4 Systemfunktion und Positionsbestimmung ... 3-30
3.3.3.5 Differential GPS (DGPS) ... 3-31
3.3.4 HARDWARE Interface ... 3-31
3.3.5 SOFTWARE Interface ... 3-31
3.3.6 BETRIEB ... 3-32
3.3.6.1 Interne (onboard) Datenquellen ... 3-32
3.3.7 Initialisation ... 3-33
3.3.7.1 Definition ... 3-33
3.3.7.2 Positions-, Geschwindigkeits- und Zeitdaten (PVT) ... 3-33
3.3.7.3 Satelliten-Ephemeriden ... 3-33
3.3.7.4 Satelliten-Almanach ... 3-33
3.3.7.5 UTC- und Ionosphären-Parameter ... 3-33
3.3.8 Konfiguration ... 3-34
3.3.8.1 Definition ... 3-34
3.3.8.2 Geodetic Datums (Geodätisches Datum) ... 3-35
3.3.8.3 Satellite Selection (Satelliten-Auswahl) ... 3-35
3.3.8.4 Differential GPS (DGPS) Control (DGPS-Kontrolle) ... 3-35
3.3.8.5 Cold Start Control (Kaltstart-Kontrolle) ... 3-35
3.3.8.6 Solution Validity Criteria (gültige Lösungskriterien) ... 3-35
3.3.8.7 Antenna Type Select (Antennen-Auswahl) ... 3-36

3.3.8.8 User-Entered Altitude (Anwenderdef. Höhenlage) 3-36
3.3.8.9 Vehicle Platform Select (Plattform-Klasse) 3-36
3.3.8.10 Navigation Control (Navigations-Kontrolle) 3-36
3.3.8.11 NMEA Select (NMEA-Auswahl) 3-36
3.3.8.12 ROM Defaults (ROM-Standard-Werte) 3-36
3.3.9 Start-Up Modes (Start-Modus) 3-37
3.3.9.1 Warm Start 3-37
3.3.9.2 Initialized Start 3-37
3.3.9.3 Cold Start 3-37
3.3.9.4 Frozen Start 3-37
3.3.10 Satelliten-Management 3-38
3.3.10.1 Liste der sichtbaren Satelliten 3-38
3.3.10.2 Erfassungs-Modus 3-38
3.3.10.3 Daten-Sammlung 3-40
3.3.11 Navigation 3-41
3.3.11.1 Geodätisches Datum 3-41
3.3.11.2 Platform Class (Plattform-Klasse) 3-42
3.3.11.3 Navigation Cycle (Navigations-Zyklus) 3-42
3.3.11.4 Solution Validity (Gültigkeit der Lösung) 3-44
3.3.11.5 Mean Sea Level (Meereshöhe) 3-45
3.3.11.6 Magnetic Variation (Magnetische Abweichung) 3-45
3.3.12 Unterstützende Funktionen 3-45
3.3.12.1 Serielle Kommunikations-Schnittstellen 3-45
3.3.12.2 EEPROM Services 3-46
3.3.12.3 RTC Services 3-47
3.3.12.4 Differential GPS (DGPS) 3-47
3.3.12.4.1 RTCM-Nachrichten-Typen 3-47
3.3.12.4.2 DGPS-Initialisation und Konfiguration 3-48
3.3.12.5 Built-In-Test (BIT) 3-49
3.3.12.5.1 Interpretation der BIT-Resultate 3-50

3.4 Zodiac Serial Data Interface 3-51
3.4.1 Zodiac Datentypen und Nachrichtenformate 3-51
3.4.1.1 Binäres Nachrichten-Format und Wort-Struktur 3-51
3.4.1.2 Binary Message Header 3-52
3.4.1.3 Binary Message Data 3-55
3.4.1.4 NMEA-Nachrichten-Format und Satz-Struktur 3-56
3.4.2 Zodiac binary Daten-Nachrichten 3-59
3.4.2.1 Ausgabe-Nachrichten Beschreibung 3-60
3.4.2.2 Eingabe-Nachrichten Beschreibung 3-61
3.4.3 Zodiac NMEA-Daten-Nachrichten 3-64
3.4.3.1 Ausgabe-Nachrichten Beschreibung 3-64
3.4.3.2 Eingabe-Nachrichten Beschreibung 3-65

3.5 VISYS® GPS ADAPTOR 3-67
3.5.1 Serial and Power Adaptor Card for Rockwell Jupiter™ Board 3-67
3.5.1.1 Allgemeines 3-67
3.5.1.2 Eigenschaften 3-67
3.5.1.3 Interface zu Jupiter™-Board 3-68

3.6 Abkürzungen & Glossar **3-69**
3.7 Allg. Hinweise und Bemerkungen **3-72**
3.7.1 Probleme 3-72
3.7.2 Behebung 3-72
3.8 Anhang zu **3-74**
3.8.1 LABMON Operation Guide for the “Jupiter” GPS Receiver [rosa]
3.8.2 Development Kit Setup and Operation Guide for the “Jupiter” GPS Receiver [lachs]
3.8.3 Zodiac GPS Receiver Family Designer’s Guide [blau]
3.8.4 Zodiac Seriel Data Interface Specification [gelb]
3.8.5 VISYS® GPS ADAPTOR [grün]

4. Konzept

4.1 Handbuch zu Jupiter-GPS-Receiver **4-1**
4.2 Blockschaltbild **4-1**
4.3 Hardware **4-2**
4.3.1 Aufbau und Realisierung 4-2
4.3.1.1 Grund- und Frontplatte 4-3
4.3.1.2 Laborkarten 1 und 2 4-3
4.3.1.3 Prototypen-Board MCB-517AC 4-3
4.4 Software **4-4**
4.4.1 Struktur und Programmablauf 4-4
4.4.2 Einschränkung 4-5
4.4.3 Funktion und Anzeige 4-5
4.5 Entwicklungsvorgang **4-7**
4.5.1 Entwicklungsumgebung 4-7

5. Hardware

5.1 Mikrocontroller 80C517A **5-1**
5.1.1 Blockschaltbild 5-1
5.1.2 Speicherkonzept 5-2
5.1.2.1 Einleitung 5-2
5.1.2.2 Speicherorganisation 5-2
5.1.2.3 Zugriff auf den externen Speicherbereich 5-3
5.1.3 Serielle Schnittstellen (COM0 und COM1) 5-5
5.1.3.1 Einleitung 5-5
5.1.3.2 Serielle Schnittstelle Nr. 0 5-5
5.1.3.3 Serielle Schnittstelle Nr. 1 5-5
5.1.3.4 Senden und empfangen 5-5
5.2 Prototypen-Baugruppe MCB-517AC **5-6**
5.2.1 Monitor Memory-Map 5-6

5.3 Mechanik **5-7**
5.4 Spannungsversorgung **5-7**
5.4.1 Beschreibung Schema 5-7
5.4.2 Dimensionierungen 5-8
5.5 Schaltung zur Adressdecodierung **5-9**
5.5.1 Adressbereich 5-9
5.5.2 Prinzip und 1. Ansatz 5-9
5.5.3 Das Problem mit dem Timing 5-10
5.5.3.1 Lösung 5-10
5.5.4 Modifizierte Adressdecodierung und Logikschaltung 5-11
5.5.4.1 Timing-Diagramm 5-13
5.5.4.2 Verwendete IC's 5-14
5.6 LCD-Display-Modul **5-15**
5.6.1 Ansteuerung 5-15
5.6.2 Dimensionierungen 5-15
5.6.3 LCD-Controller HD44780 5-16
5.6.3.1 Schreib-/Lesezugriff 5-18
5.6.3.2 Befehlssatz 5-18
5.7 Tastatur 4x4 Tasten **5-18**
5.7.1 16-Key Encoder MM74C922 5-19
5.7.2 Ansteuerung 5-20
5.7.3 Tastencodierung 5-20
5.7.4 Dimensionierungen 5-20

6. Software

6.1 GPS-Protokolle **6-1**
6.1.1 Das NMEA-Nachrichten-Format 6-1
6.1.2 Das Binary Message Format 6-2
6.1.3 Unterschiede 6-2
6.2 Der GPRMC-Datensatz **6-3**
6.2.1 Auszuwertende Daten 6-4
6.2.1.1 Darstellungsform 6-4
6.3 Schweizer Landeskoordinaten **6-4**
6.3.1 Umrechnungs-Formeln 6-5
6.4 Programm **6-6**
6.4.1 Einleitung 6-6
6.4.2 Programm-Architektur 6-6
6.4.3 Allgemeines zum Source-Code 6-7

6.5 Programm-Module **6-8**
6.5.1 Interrupt serielle Schnittstelle COM1 6-8
6.5.2 Externer Interrupt 4 6-9
6.5.3 Hauptprogramm (MAINLOOP) 6-9
6.5.4 Abfrage Tasten (TASTE) 6-10
6.5.5 Abfrage Daten gültig (STATUS) 6-10
6.5.5.1 Aufbereiten der Rohdaten (L_B_DAT) 6-11
6.5.6 Verbindung COM1 ⇨ COM0 (TRANSFER) 6-11
6.5.7 “allgemeine” Subroutinen 6-12
6.6 Datenablage im RAM-Speicher **6-12**
6.6.1 Memory-Map 6-14
6.7 Flussdiagramme **6-14**

7. Inbetriebnahme

7.1 der Hardware-Komponenten **7-1**
7.1.1 GPS-Empfänger-Modul 7-1
7.1.2 Spannungsversorgung 7-1
7.2 Adressdecodierschaltung **7-2**
7.3 Bedienungsanleitung **7-3**
7.3.1 Benötigte Komponenten 7-3
7.3.2 Gerätebeschreibung (1) 7-3
7.3.2.1 Übersicht und Anschluss 7-4
7.3.3 Bedienung 7-5

8. Zusammenfassung

8.1 Randprobleme **8-2**
8.2 Ausstehendes / Verbesserungen **8-2**
8.3 Kommentar **8-3**
8.4 Verdankung **8-4**
8.5 Schlusswort **8-5**

9. Literaturverzeichnis

9.1 Literatur “Papier” **9-1**
9.2 Literatur “Bauteile” **9-2**
9.3 Literatur “elektronisch” **9-2**

10. Anhang

10.1 Benutzte Hilfsmittel

10.2 Kopie der erwähnten Berichte [5,6,9]

10.3 Zeitplan

10.4 Schemata Hardware

10.5 Stückliste

10.6 Software-Listings

10.7 Datenblätter

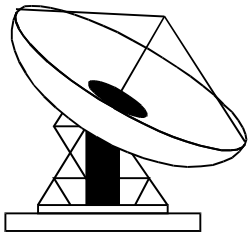
10.8 Diskette / CD-ROM

1. Allgemeines

1.1 Einleitung

Basierend auf einem Rockwell *GPS-Receiver-OEM-Modul* wurde unter Verwendung der Prototypen-Baugruppe MCB-517AC eine Hardware-Plattform für den Empfang von GPS-Daten realisiert. Die zum *GPS-OEM-Modul* mitgelieferten technischen Unterlagen wurden in eine deutschsprachige “Kurzanleitung” abgefasst, welche integrierender Bestandteil dieser Dokumentation ist.
Die folgenden Abschnitte sollen dem Leser einen Überblick über die in dieser Diplomarbeit enthaltenen Komponenten verschaffen.

1.2 Kurze Einführung in das NAVSTAR-GPS



Das **GPS** (**G**lobal **P**ositioning **S**ystem) ist ein vom Verteidigungsministerium der U.S.A **DOD** (**D**epartment **o**f **D**efense) installiertes und betriebenes, weltweit verfügbares Navigationssystem. Es besteht aus 24 Satelliten, wovon 21 aktiv sind und 3 als Reserve-Satelliten dienen. In einer Höhe von etwa 20’200 km umkreisen diese **SV** (**S**pace **V**ehicles) die Erdkugel auf 6 verschiedenen, zum Äquator um 55° geneigten Umlaufbahnen in genau 11 Stunden und 58 Minuten.
Fünf über die gesamte Erdkugel verteilte Kontrollstationen unterhalten das System laufend.

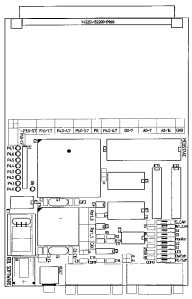
Damit nun ein solcher Empfänger seine Position bestimmen kann, müssen mindestens 4 Satelliten gleichzeitig am Himmel sichtbar sein, um das Gleichungssystem mit den 4 Unbekannten X, Y, Z und t zu lösen. Mit Hilfe der an jedem Satellit in 4-facher Ausführung vorhandenen Atomuhren wird über die Laufzeit des gesendeten Satellitensignales die Entfernung des jeweiligen Satelliten berechnet ($s = c \cdot t$).
Die Abweichung der weniger genauen Empfänger-Uhr wird im Laufe der Positionsbestimmung über den vierten Satelliten ermittelt, welche von da an auf die Satellitenzeit synchronisiert ist.

Die Antenne weist in den meisten Fällen eine halbkugelförmige Charakteristik auf, um einerseits die schwachen Signale (ca. 70nV !) der sichtbaren Satelliten gleich gut zu empfangen, und andererseits die von reflektiven Gegenständen unerwünschten Störsignale unterdrücken zu können.

Der Gesamtkomplex eines satellitengestützten Ortungs- und Navigationssystems umfasst im wesentlichen drei Hauptgruppen:

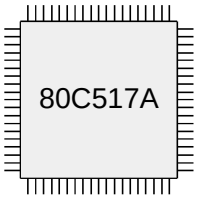
- Die Satelliten als Sendernetz *(Space Segment)*
- Die Boden- oder Kontrollstationen *(Control Segment)*
- Den Empfänger *(User Segment)*

1.3 Prototypen-Baugruppe MCB-517AC



Die Micro-Controller-Baugruppe MCB-517AC ermöglicht das Testen und Entwickeln von 8051-Programmen, in diesem Fall auf der Basis des Prozessor-Typs 80C517A. Zudem verfügt die Baugruppe über zwei RS232-Schnittstellen, welche zur Kommunikation mit anderen Computersystemen dienen.

1.3.1 Der 80C517A



Eine detaillierte Beschreibung des 80C517A (im folgenden *Mikrocontroller* genannt) würde an dieser Stelle den Rahmen sprengen. Kurz zusammengefasst kann dieser Prozessor-Typ wie folgt charakterisiert werden:

“Der 80C517A der Fa. SIEMENS ist ein äusserst leistungsstarker 8-Bit-Single-Chip-Mikrocontroller, der eine Vielzahl von Funktionen auf dem Chip integriert hat. Er basiert auf der 8051-Architektur (weitverbreiteter Industriestandard, der von der Firma Intel entwickelt wurde) und ist vollständig software-kompatibel zum Grundbaustein 80(C)51.” (aus *“Mikrocontroller Praxis”* [1])

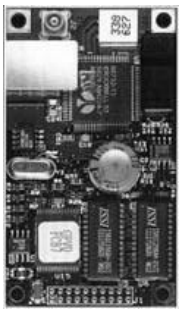
1.3.1.1 Features

Features

Die folgende Aufzählung gibt einen ersten Überblick über einige Leistungsmerkmale dieses Controllers:

- 8051-Core mit einer Taktfrequenz von 12 MHz
- 8 kByte grosser interner Programmspeicher (on-chip code memory)
- 256 Byte grosser interner Arbeitsspeicher (on-chip-RAM), davon 256 direkt adressierbare Bits
- 64 der total 111 Software-Befehle werden innerhalb eines Zyklus abgearbeitet
- 8-Bit A/D-Wandler mit 12 gemultiplexten Eingängen und programmierbaren Referenzspannungen
- Zwei 16-Bit Timer / Counter
- Eine leistungsstarke Compare / Capture-Einheit
- Eine spezielle Arithmetik-Einheit für div. 16 bis 32-Bit Operationen
- Acht Datenpointer
- Sieben bidirektionale, digitale 8-Bit-Ports (Port 0 bis Port 6)
- Zwei serielle Schnittstellen (voll-duplex) mit eigenen Baudraten-Generatoren (COM 0 und COM 1)
- Ein Interrupt-System mit 4 Prioritäts-Stufen und 14 Interrupt-Quellen
- Drei verschiedende Power Saving Modes, mit denen softwaremässig die Leistungsaufnahme des Controllers deutlich reduziert werden kann

1.4 "Jupiter"-GPS-Receiver-Board



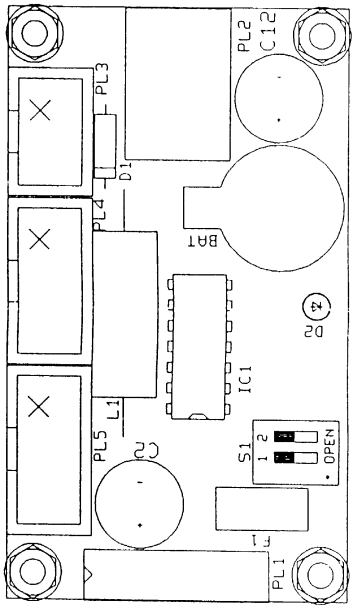
Wie in *Kapitel 3* noch ausführlicher beschrieben, ist von der Firma Rockwell ein **OEM**-Modul (**O**riginal **E**quipment **M**anufacterer) erhältlich, welches auf dem Zodiac Chipsatz basiert. Dieser, als "Jupiter"-Board bezeichneter, Empfänger ist ein 12-Kanal-Empfänger mit sehr schneller Reaquasitionszeit, **NMEA** (**N**ational **M**arine **E**lectronics **A**ssociation)-Protokoll und **DGPS** (**D**ifferential **G**lobal **P**ositioning **S**ystem)-Funktionen.

1.4.1 GPS-Adaptor-Card

Obwohl ich in Kapitel 3 das gesamte "Development Kit" vollumfänglich beschrieben habe, benutzte ich selbst nur die GPS-Adapter-Karte zu dem "Jupiter"-Board. Diese, als "Serial and Power Adaptor Card" bezeichnete, Printplatte mit den geringen Abmessungen von 70 x 40 x 20 mm wurde mir zusammen mit dem gleich grossen "Jupiter"-Empfänger-Board von Herrn Bazali zur Verfügung gestellt. Diese Adapter-Karte besitzt, abgesehen von den beim Development Kit reichlich vorhandenen Konfigurationseinstellungen, sämtliche Eigenschaften um eine Positionsbestimmung durchzuführen (Techn. Daten siehe *Anhang*).

Bemerkung:
Die GPS-Adapter-Karte wird im folgenden vereinfachend als GPS-Empfänger(-Modul) bezeichnet

Mit dem einzigen vorhandenen Konfigurationsschalter S1 konnte ich (gemäss Datenblatt) zwischen den folgenden Einstellungen wählen:



S1-1	S1-2	Haupt-Port	Initialisierungs-Werte Zeit / letzte Pos.	Nachrichten-Format
ON	ON	SRAM/EEPROM	SRAM/EEPROM	SRAM/EEPROM
ON	OFF	9600-N-8-1	ROM	BINARY
OFF	ON	4800-N-8-1	SRAM/EEPROM	NMEA
OFF	OFF	4800-N-8-1	ROM	NMEA

Da ich erst nach einiger Zeit herausfand, dass sich die Beschreibungen der PL1-Interface-Signale dieser Adapter-Karte im spezifischen *Zodiac GPS Receiver Data Sheet* befinden, mir aber die Zeit fehlte, dieses auch noch zu übersetzen, ist es im *Anhang* zu *Abschnitt 3.3* (blau) ersichtlich.

Auf die interessanten und zum besseren Verständnis dienenden Details wird in den folgenden Kapiteln bei Bedarf noch näher eingegangen.

Für weiterführende, einschlägige Literatur der hier kurz vorgestellten Gebiete sei auf das Literaturverzeichnis im *Kapitel 9* verwiesen.
Drei kurz gehaltene und trotzdem sehr informative Einführungen in das GPS sowie deren Technik möchte ich dem interessierten Leser jedoch nicht vorenthalten und habe sie aus diesem Grund dem *Anhang* beigelegt ([5], [6] u. [9]).

Kapitel 9: Literaturverzeichnis

1.5 Bestimmungen für die Diplomarbeit

2. Aufgabenstellung

2.1 Original

Nachstehend ist die Aufgabenstellung im Original zu finden.

2.2 Detaillierung

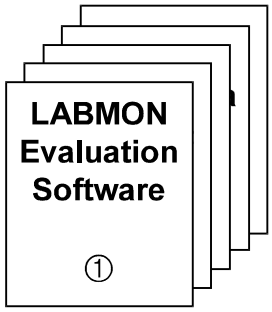
Besprechungen

Wie sich nach der zweiten Besprechung mit Herrn Bazali herausstellte, lag das Schwergewicht vorerst in der Aufarbeitung der englischen Manuals sowie der Kommunikation des GPS-Empfänger-Moduls über die RS 232-Schnittstelle mit einem PC. Dies führte zu einer ersten Modifizierung des von mir erstellten Zeitplanes (siehe *Kapitel 4*).
Da die Wichtigkeit und der (Zeit)-Aufwand dieser ersten Teilaufgabe nicht zu unterschätzen ist, trafen wir die Vereinbarung, dass die Implementierung einer Tastatur sowie eines LCD-Displays zweitrangig und somit abhängig von der noch vorhandenen Zeit zu realisieren sei.

Bei der 4. Besprechung (Woche 6) wurde diese Abmachung jedoch wieder vergessen, da ich ohnehin mindestens die Spannungsversorgung und einen Aufbau der verschiedenen Komponenten gemäss dem Konzept vorsehen musste. Somit wurde die Speisung und der ungefähre mechanische Aufbau festgelegt und das zu verwendende Display sowie eine Tastatur evaluiert.

Im folgenden sind die wichtigsten Punkte zur weiterführenden Detaillierung der Aufgabenstellung aufgeführt:

2.2.1 Handbuch zu Jupiter-GPS-Receiver



- Übersetzen und aufbereiten der gesamten mitgelieferten technischen Dokumentation des “Rockwell Jupiter-GPS-Receiver Development Kit” in eine geeignete, prakmatische Form.
- Die 4 technischen Manuals sollen zusammenfassend in einem dafür vorgesehenen Register als integrierender Bestandteil der gesamten Dokumentation enthalten sein.
- Ebenso soll dieses Kapitel aber auch als eigenständiges Dokument mit eigenem Inhaltsverzeichnis verwendet werden können.
- Bezüglich Auswertung der Handbuch-Informationen soll auch kurz auf die Protokolle **NMEA** und **Rockwell binary** eingegangen und die wesentlichen Unterschiede dargestellt werden.

2.2.2 Hardware

2.2.2.1 Aufbau

Die Hardware soll in geeigneter Form auf einer Platte o. ä. angeordnet werden. Ein Gehäuse als Schutz wird in diesem Sinn nicht benötigt.

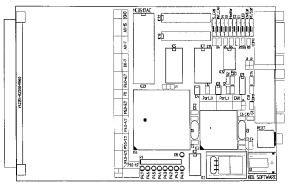
Der Hardware-Aufbau muss die folgenden Komponenten enthalten:

- Benötigte Versorgungs-Spannungen
- Prototypen-Baugruppe MCB-517AC
- LCD-Display mit 4x20 Zeichen
- Tastatur mit 4x4 Tasten

2.2.2.2 Benötigte Versorgungs-Spannungen

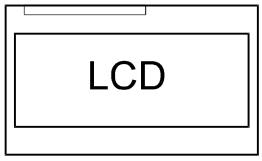
- Die fertige Einrichtung muss mit der Netzspannung (230V AC) sowie mit dem Zigaretten-Anzünder (12V DC) im Auto betrieben werden können um mobilen Betrieb zu ermöglichen.
- Benötigt werden +5V DC für die Logik, unstabilisierte 12V DC für das Prototypen-Board und das GPS-Empfänger-Modul sowie eine Spannung von ca. 9V für die Beleuchtung des LCD-Displays (Anoden/Kathoden-Spannung = 4V).
- Die Speisung soll nach den Vorgaben des Dozenten selbst aufgebaut werden.

2.2.2.3 Prototypen-Board MCB-517AC



- Das Prototypen-Board wird von der Schule zu Verfügung gestellt. Jedoch sollen keine hardwaremässigen Eingriffe auf das Board vorgenommen werden.
- Das Board muss als Modul angesehen werden und daher frei in die Einrichtung integrierbar sein.

2.2.2.4 LCD-Display mit 4x20 Zeichen



- Es soll ein geeignetes LCD-Display mit Hintergrundbeleuchtung evaluiert werden, welche bei Bedarf eingeschaltet werden kann.
- Das Display soll direkt über den Adress-/Datenbus des Mikrocontrollers angesprochen werden, damit die Ports für Experimente frei bleiben.
- Soweit realisierbar, soll das Display zur Kommunikation mit dem Benutzer entsprechende Meldungen und Informationen über den Status anzeigen.

2.2.2.5 Tastatur mit 4x4 Tasten

1	2	3	F
4	5	6	E
7	8	9	D
A	0	B	C

- Die Tastatur soll 16 Tasten enthalten, von denen jedoch vorerst nur einige benötigt werden.
- Bei der Betätigung einer nicht implementierten Taste muss eine entsprechende Fehlermeldung auf dem Display erscheinen.

- Die Tastatur soll wie das Display mit einer geeigneten Decodierschaltung direkt über den Adress-/Datenbus des Mikrocontrollers angesprochen werden.

2.2.2.6 Adressdecodierung

- Das Ansteuern der Peripherie über den Adress-/Datenbus erfordert die Entwicklung einer Adress-Decodierschaltung. Der “User Chip Select”-Bereich ist durch die Verwendung des Prototypen-Boards ab Adresse F600H definiert.

2.2.3 Software

- Einarbeitung in die LABMON-Software zur Inbetriebnahme und Kommunikation des GPS-Empfänger-Moduls mit einem PC.
- Schreiben eines Programmes in Assembler für den 80C517A, welches die von dem GPS-Empfänger-Modul erhaltenen Daten einliest und zur Anzeige auf dem Display entsprechend aufbereitet.
- Allfällige Berechnungsroutinen sollen in einer frei wählbaren, geeigneten Hochsprache implementiert werden und können, sofern vorhanden, von bereits bestehenden Diplomarbeiten übernommen werden.
- Zur Positionsbestimmung sollen in dieser Arbeit zunächst nur die Daten des NMEA-0183-Protokolles ausgewertet und in Form von Längen-und Breitengraden sowie in Schweizer-Landeskoordinaten auf dem Display dargestellt werden.

```
LCD_TXT:     CLR     A
             MOVC    A,@A+DPTR
             JZ      LCD_ENDE
             CALL    LCD_WRITE
             INC     DPL
             JMP     LCD_TXT
LCD_ENDE:    RET
TXT_ENDE:    CALL    LCD_OFF_TXT
             JMP     LCD_ENDE
```

2.2.4 Ziel

- Die fertige Einrichtung soll als Hardware-Plattform für weitere Untersuchungen und Experimente mit dem GPS-Empfänger-Modul verwendet werden können.

3. Handbuch zu Jupiter-GPS-Receiver

HINWEISE ZUR BENÜTZUNG DIESES HANDBUCHES

Gliederung und Inhaltsverzeichnis

Die Gliederung dieser deutschsprachigen Version entspricht im wesentlichen der des Originals, dies erleichtert Quervergleiche.

Abkürzungen und Tabellen

Die in diesem Dokument zusammengefassten Manuals besitzen in der Originalversion alle einen umfangreichen Anhang , welcher jeweils nicht übersetzt wurde, da es sich in den meisten Fällen um Erklärungen von Abkürzungen und in Tabellen übersichtlich zusammengestellten Befehls- oder Konfigurationseinstellungen handelt.
Da im Umgang mit dem Development Kit und der dazugehörigen Software sowieso die englischen Ausdrücke verwendet werden, macht es keinen Sinn diese zu “verdeutschen”.
Die notwendigen, dem besseren Verständis dienenden Übersetzungen und Abkürzungs-Erklärungen wurden jeweils an Ort und Stelle vorgenommen.

Abkürzungen und Glossar

Eine deutsche Beschreibung der zum Verständnis dieses Dokuments notwendigen Begriffe befindet sich im Abschnitt 3.6

Der Anhang

Aus jedem Anhang der 4 Manuals wurden die benötigten Informationen herausgenommen und sind am Schluss dieses Dokumentes als Kopien vorhanden.
Da in den Verweisen auf den Anhang nun Überschneidungen in der Bezeichnung auftreten (Appendix A, B, ect.), ist jeder Anhang durch ein farbiges Blatt vom anderen getrennt.

Symbole

Die in diesem Dokument verwendeten Symbole haben die folgende Bedeutung:

WARNUNGEN sind Informationen über Personen- und Sachschäden, die

- bei nichtbeachten von Hinweisen
- oder unsachgemässem Gebrauch von Geräten entstehen können

WARNUNG

HINWEISE sind Informationen, die

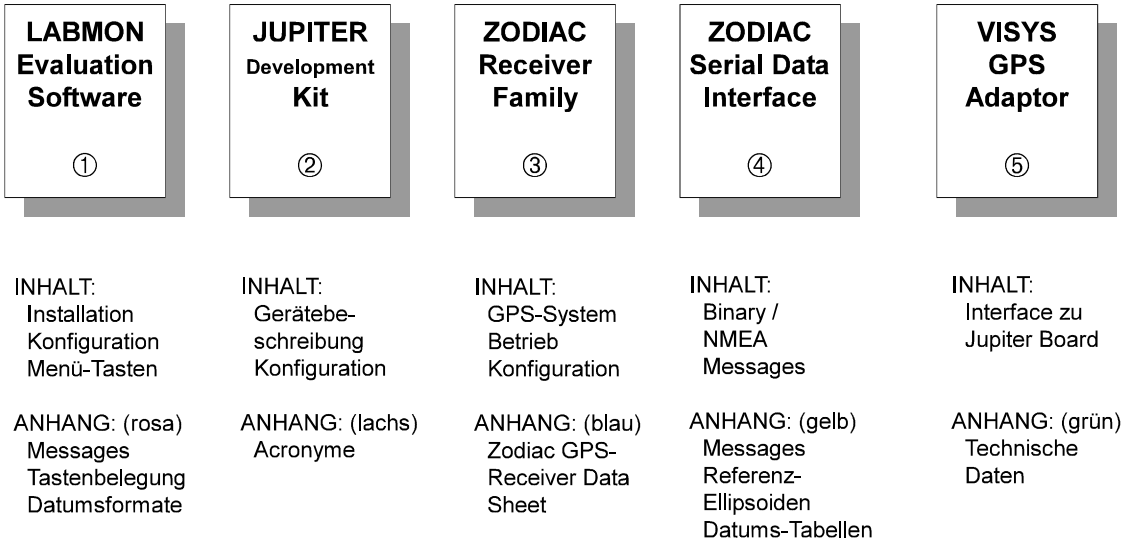
- auf technische Erfordernisse,
- nützliche Tips und / oder allgemeine Bemerkungen hinweisen

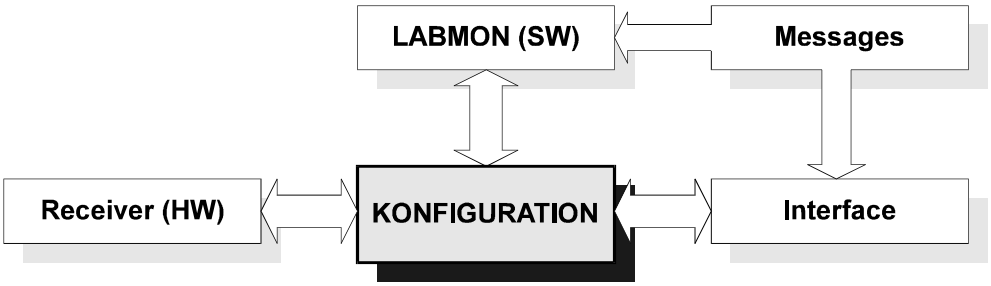
HINWEIS

Der Übersetzer übernimmt keine Garantie für die Vollständigkeit und Richtigkeit der Angaben !

Übersicht

Um den Umfang und den wesentlichen Inhalt dieses Dokumentes auf einen Blick zu erfassen, habe ich versucht dies graphisch darzustellen:





Die 4 Manuals sowie die Kurzbeschreibung der GPS-Adapter-Karte sind in die 5 Abschnitte *3.1-3.5* unterteilt.

Aus den zahlreichen vom GPS-OEM-Modul empfangenen und von der LAB-MON-Software ausgewerteten Daten ist nur ein minimaler Anteil zur Positionsbestimmung nötig. Innerhalb dieser Arbeit wird zur Verarbeitung mit dem Mikrokontroller 80C517A nur der folgende Datensatz verwendet:

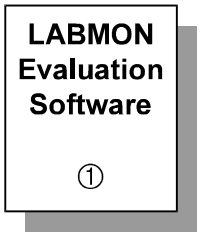
Vom Standard-NMEA-Protokoll:

Recommended Minimum Specific GPS Data RMC *$GPRMC,...,...,...,...,...*41*

Vom Rockwell-binary-Protokoll:

Von diesem Protokoll wurden keine Daten verwendet. Die Auswertung der *"Binary Message"*-Daten bleibt einer nächsten Diplomarbeit zum Thema GPS vorbehalten. Die Voraussetzungen dazu wurden mit der Erstellung dieses deutschsprachigen Handbuches sowie der Hardware-Plattform geschaffen.

3.1 LABMON Evaluation-Software



Der erste Teil dieses Dokumentes beschreibt die Version 4.7 der LABMON Evaluation-Software. Obwohl sie eingeschränkt auch für die älteren NavCore-Empfänger eingesetzt werden kann, ist sie vollumfänglich nur für die Zodiac-Empfänger-Familie konzipiert. Deshalb werden auch nur die für diese Familie relevanten Funktionen beschrieben.

3.1.1 Einleitung

Einleitung

Das Rockwell Zodiac Development-Kit ist entwickelt worden, um die Auswertungen des “Jupiter”-GPS-Empfängers, welcher auf einem Zodiac-Chipsatz basiert, zu erleichtern.
Das Development-Kit beinhaltet Empfänger-Kontroll-Operationen und Ein/Ausgabe-Funktionen (I/O) unter Verwendung eines IBM-AT kompatiblen Personal Computers (PC), einem seriellen Port, einer externen Antenne und einer Stromversorgung.

Dieses Dokument beschreibt die Benutzung der GPS-Monitor- und Kontroller-Software, in Bezug auf LABMON und das Development Kit. LABMON läuft auf einem PC und gestattet dem Benutzer die vom Empfänger ausgewerteten Daten darzustellen.
Externe **R**adio **T**echnical **C**ommission for **M**aritime Services-Daten (RTCM SC-104) können über eine zweite serielle Schnittstelle ebenfalls empfangen werden.

Zur Zeit existieren zwei Serien von kommerziellen Rockwell-GPS-Empfängern: Die erste ist die bereits angesprochene NavCore-Serie und die zweite die Zodiac-Familie. Diese besteht aus dem “Jupiter”-Board und dem Zodiac-Chipsatz.

Der Empfänger kann entweder mit einem seriellen Schnittstellen-Protokoll im binären Format oder mit dem **N**ational **M**arine **E**lectronics **A**ssociation (NMEA-0183)-Protokoll betrieben werden. Die Zodiac und NavCore-Empfänger teilen sich ein ähnliches Rockwell-binary-Format, welche jedoch verschiedene ID’s und Inhalt haben und somit nicht verwechselt werden dürfen. Um Verwechslungen zu vermeiden, werden die beiden Protokolle im folgenden mit *Zodiac binary* und *NavCore binary* bezeichnet.
Genauso muss zwischen *Zodiac NMEA* und *NavCore NMEA* unterschieden werden, da nebst einer Anzahl standardisierten Datenformaten auch herstellerspezifische Formate bestehen, die bei Rockwell nicht für beide Empfängertypen gleich sind.
Die Empfänger der Zodiac-Serie beinhalten standardmässig beide Protokolle, also *Zodiac binary* und *Zodiac NMEA*. Die Auswahl erfolgt softwaremässig oder durch Anlegen einer Spannung an einem externen Pin des Empfängers.

LABMON 4.6 unterstützt beide Empfängertypen. Zur Kommunikation muss jedoch das unter LABMON ausgewählte Protokoll mit dem entsprechenden Empfänger übereinstimmen.

Diese Einstellung ist wichtig, da die beiden binary-Protokolle nicht kompatibel sind. Die NMEA-0183-Protokolle sind zwar soweit ähnlich, dass die Standard-Formate empfangen sowie ausgegeben werden können, jedoch Konflikte bei den Rockwell-eigenen Formaten entstehen.

Detaillierte Informationen zur Konfiguration des Development-Kit befinden sich im Abschnitt *3.2* . Über die Zodiac-GPS-Empfänger-Familie erfahren Sie mehr im Abschnitt *3.3* .

3.1.2 LABMON installieren

installieren

LABMON ist auf jedem DOS-kompatiblen PC lauffähig. Ebenso ist es möglich die Software unter Windows 3.x aus einer DOS-Box zu starten.

Es kann vorkommen, dass Windows und DOS den gleichen Interrupt für die serielle Schnittstelle benützen wollen, und somit ein Konflikt entsteht. Wenn Probleme unter Windows auftreten, schliessen sie Windows und starten sie LABMON erneut unter DOS.

Sollte LABMON aufgrund der Doppelbelegung eines seriellen Ports (z. B. mit der Maus) nicht mehr korrekt arbeiten, ist die Maus auf den anderen Port zu legen.

Obschon der gesamte Inhalt der Diskette auf die Festplatte kopiert werden muss, empfiehlt es sich, diese als Sicherheitskopie aufzubewahren.

3.1.3 LABMON unter DOS installieren

unter DOS installieren

- Erstellen Sie ein Verzeichnis auf der Festplatte und kopieren Sie den gesamten Disketten-Inhalt in dieses.
- Erweitern Sie die Pfad-Angaben in der AUTOEXEC.BAT um dieses Verzeichnis (z.B. PATH=C:\LABMON)
- Entfernen Sie die Diskette aus dem Laufwerk und bewahren Sie sie als Sicherheitskopie auf.
- Starten Sie den PC erneut, um die Einstellungen der AUTOEXEC.BAT zu aktualisieren.

3.1.4 LABMON unter Windows installieren

unter Windows installieren

- Befolgen Sie zunächst die 4 Schritte zur Installation wie unter DOS.
- Starten Sie Windows, öffnen Sie den Programm-Manager und wählen Sie NEU unter dem Menu DATEI.
- Wählen Sie “Programm” und klicken Sie auf OK.
- Geben Sie als Beschreibung “LABMON” ein.

- Als Befehlszeile geben Sie den gesamten Pfad mit dem Namen und Erweiterung (.exe) der Programmdatei an oder suchen diese über die Schaltfläche “Durchsuchen...”.

ACHTUNG: *Die auszuführende Datei heisst in diesem Fall LABMON47.EXE !*

- Geben Sie den korrekten Pfad des LABMON-Verzeichnisses (z. B. C:\LABMON) oder ein von Ihnen gewähltes Verzeichnis als Arbeitsverzeichnis an.
- Um das LABMON Programm-Icon zu wählen, klicken Sie auf “Anderes Symbol...”, dann auf OK und erneut auf die Schaltfläche “Durchsuchen...”. Das Icon befindet sich im LABMON-Installationsverzeichnis mit dem Namen Labmon.ico. Quittieren Sie anschliessend zweimal mit OK.

3.1.5 LABMON konfigurieren

Konfiguration

Über die <Alt>+Funktionstasten (F1-F9) können folgende Einstellungen (CFG OPTIONS) verändert werden:

- Protokoll-Typ (binary oder NMEA)
- UTC-Zeitdifferenz
- Angezeigtes Datum
- Bildschirm-Farben
- Referenz-Position
- Filter-Parameter

HINWEIS:
Die Angabe <Alt>+F1 bedeutet, dass bei gedrückter <Alt>-Taste zusätzlich die Taste F1 betätigt werden muss.

Diese Einstellungen können ebenfalls in der Konfigurations-Datei LABMON.CFG vorgenommen werden.

3.1.5.1 Die Datei LABMON.CFG

Die Datei LABMON.CFG

Sollten beim Einstellen dieser Optionen Schwierigkeiten auftreten, ist die oben genannte Datei auf fehlerhaften Inhalt zu überprüfen. Die Datei LABMON.CFG wird vom ausführenden Programm mit den Standard-Einstellungen automatisch erzeugt, wenn sie einmal gelöscht werden sollte.
Beim starten von LABMON wird geprüft, ob sich die Datei LABMON.CFG im LABMON-Installationsverzeichnis befindet. Wenn ja, werden die Einstellungen übernommen. Wenn nicht, wird eine Datei mit den folgenden Standard-Einstellungen generiert und das Programm in dieser Konfiguration gestartet:

Modifikation

```
DATA TYPE 4
DATUM NUMBER 1012
UTC OFFSET -2
SPD UNITS 0
COLORS 14 10 3 10 0 12
LAT 33.662212 LON -117.862663 ALT -3.310000
FILTERS STATMASK FFFF FOM 5 QUALITY 1 SATS 3 PDOP 6.000000
HDOP 6.000000 VDOP 6.000000
```

ACHTUNG: *Die Einstellungen eines Parameters müssen sich in der Datei alle auf einer Zeile befinden (kein Zeilenumbruch).*

Im folgenden einige Bemerkungen zu den einzelnen Parametern. Eine vollständige Übersicht sämtlicher Tastenkombinationen ist im *Anhang* ersichtlich.

DATA TYPE
<Alt>+F2

- Auswahl des Protokoll-Typs: 1 für Rockwell *NavCore binary*, 2 für *NavCore NMEA*-0183, 4 für Rockwell *Zodiac binary* und 8 für *Zodiac NMEA*-0183

DATUM NUMBER
<Ctrl>+F6

- Bei Benützung eines Zodiac-Empfängers ist das Datum nicht von der Einstellung in der Konfigurations-Datei abhängig. Dafür können mit der Tastenkombination <Ctrl>+ die in *Anhang* C beschriebenen Datum-Nummern angewählt werden.

LABMON ignoriert die Datums-Einstellung in der Konfigurations-Datei bei Benützung eines Zodiac-Empfängers, da die Zodiac-Familie wesentlich mehr Datums-Formate unterstützt als die NavCore-Serie. Der dennoch vorhandene Datum-Eintrag ist auf die Abwärts-kompatibilität von LABMON mit NavCore-Empfängern zurückzuführen.

UTC OFFSET
<F1>

- Die UTC-Zeitdifferenz ist das Ergebnis von der UTC (Universal Time Coordinated) minus der lokalen Zeit. Dieser Wert ist mit der PC-Zeit zu addieren und wird als Standard-Zeit für die Empfänger-Initialisierung benötigt. Dabei ist es wichtig, dass die PC-Zeit richtig eingestellt ist.

COLORS
<Alt>+F4-F9

- Die Bildschirm-Farben können mit den Tastenkombinationen <Alt>+F4-F9 nach den im CFG OPTIONS-Menü angezeigten Attributen verändert werden.

Die Referenz-Position wird als Standard-Wert für die richtige Empfänger-Initialisierung sowie als Referenz-Wert im “Delta”-Anzeige-Modus benötigt. Auch hier ist die richtige Einstellung wichtig, wenn die von LABMON angegebenen Standard-Einstellungen verwendet werden.

LAT, LON, ALT
<Alt>+F3

- Wenn die aktuelle Position unbekannt ist, erhält man sie durch Betätigen der Tastenkombinatione <Alt>+F3 während der Zeit, in welcher der Empfänger die Satelliten sucht.

FILTER
<F>

- Die Datenfiltrierungsparameter STATMASK, FOM, QUALITY, SATS, PDOP, HDOP und VDOP müssen für die gewünschte Lösungsauswertung mit der <F>-Taste angewählt werden. Die vorkommenden Parameter sind vom gerade verwendeten Protokoll abhängig.

Wenn nicht alle Kriterien übereinstimmen, leuchtet die Anzeige FILTER ON auf dem Bildschirm. Wenn in diesem Fall Daten für die Weiterverarbeitung in eine Textdatei exportiert werden, werden die Daten nicht in die Datei geschrieben. Wenn alle Kriterien übereinstimmen, leuchtet die Anzeige nicht, und die Daten werden in die Datei exportiert und geschrieben.

Dies ermöglicht die Filtrierung oder Abschirmung von Datenausgaben, welche unter Konditionen berechnet werden, die nicht den vom Benutzer vorgegebenen Kriterien entsprechen. Wenn also die Anzeige leuchtet, bedeutet dies, dass die aktuellen Daten den Benutzerkriterien (Filtereinstellungen) nicht genügen und somit nicht in die Datei geschrieben werden.

3.1.5.2 Die Datei LABMON.INI

Die Datei LABMON.INI

Diese Datei enthält je eine Zeile mit Port-Einstellungen für das GPS- sowie das RTCM-Protokoll mit den folgenden Parametern:
Serielle Schnittstelle (COM-Port), Interrupt-Nr., Baud-Rate, Parity-Bit, Anzahl Datenbits und Stop-Bit.

Die aktuellen I/O-Adressen sowie die Interrupts werden nach dem Starten von LABMON am unteren Bildschirmrand angezeigt.
Die Auswahl der seriellen Schnittstelle (COM1, COM2) kann wiederum auf zwei Arten erfolgen:
Entweder nach dem Starten des Programmes über das I/O-Menü (Tasten <Alt>+F1) oder aber durch Modifikation der Initialisierungs-Datei LABMON.INI

Sollten beim Einstellen dieser Optionen Schwierigkeiten auftreten, ist die oben genannte Datei auf fehlerhaften Inhalt zu überprüfen. Die Datei LABMON.INI wird vom ausführenden Programm mit den Standard-Einstellungen automatisch erzeugt, wenn sie einmal gelöscht werden sollte.
Beim starten von LABMON wird geprüft, ob sich die Datei LABMON.INI im LABMON-Installationsverzeichnis befindet. Wenn ja, werden die Einstellungen übernommen. Wenn nicht, wird eine Datei mit den folgenden Standard-Einstellungen generiert und das Programm in dieser Konfiguration gestartet:

Standard für Zodiac-Empfäger

```
GPS        COM1 IRQ4 9600 N 8 1

RTCM       COM0 IRQ3 9600 N 8 1
```

Bei Verwendung eines RTCM-Protokolles sind die richtigen Einstellungen der entsprechenden Hardware-Dokumentation zu entnehmen.

Modifikation der Datei LABMON.INI

Modifikation

Zu beachten ist, dass Gross- sowie Kleinbuchstaben verwendet werden können, wobei die einzelnen Parameter durch Leerzeichen (space) zu trennen sind.

Syntax der Parameter:

[Port] [Comn] [IRQm] [Baud] [Parity] [DataBits] [StopBits]

Folgende Angaben sind erlaubt:

- GPS für GPS-Port, RTCM für RTCM-Port — *Port*
- n = 1, 2, 3 oder 4 (Port setzen)
  n = 0 (Port ignorieren) — *COMn*
- m = 3, 4, 5 oder 7 — *IRQm*
- 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800 oder 115200 — *Baud*
- n, o, e (None, odd, even = nicht, ungerade, gerade) — *Parity*
- 6, 7 oder 8 — *DataBits*
- 1 oder 2 — *StopBits*

3.1.6 LABMON starten

starten

Neben dem Standard-Modus kann LABMON noch in 2 weiteren Modi gestartet werden, in denen man die Daten über die serielle Schnittstelle aufzeichnen kann:

Zusatz-Mode 1: Daten können vom und zum Zodiac-Host sowie auf die Festplatte gespeichert werden (1 Port).

Zusatz-Mode 2: Wie Mode 1 und zusätzlich dasselbe mit RTCM-Daten (2 Ports).

LABMON erkennt den Modus nur beim aufstarten, sofern die entsprechenden Zusatz-Parameter übergeben werden:

3.1.7 LABMON unter DOS starten

unter DOS starten

Beispiel

- **Standard-Modus:** Geben Sie LABMON ein
- **Zusatz-Mode 1:** Geben Sie LABMON GPS.DAT ein, um GPS-Daten aufzuzeichnen (die Daten werden in die Datei GPS.DAT geschrieben)
- **Zusatz-Mode 2:** Geben Sie LABMON GPS.DAT RTCM.RTC ein, um GPS-Daten und RTCM-Daten aufzuzeichnen

3.1.8 LABMON unter Windows 3.x aus einer DOS-Box starten

aus einer DOS-Box starten

- Mittels MS DOS-Eingabeaufforderung eine DOS-Box öffnen
- Am DOS-Prompt dieselben Eingaben tätigen wie oben beschrieben
- EXIT eingeben, um die DOS-Box zu schliessen

3.1.9 LABMON unter Windows 3.x starten

unter Windows 3.x starten

- Hier müssen Sie die Befehlszeile um die entsprechenden Einträge (...GPS.DAT RTCM.RTC) erweitern. Aktivieren Sie dazu im Programm-Manager das LABMON-Icon und wählen Sie Eigenschaften aus dem Menü Datei oder betätigen Sie die Tasten <Alt>+Enter.

Um Daten anzuzeigen, müssen der GPS-Protokoll-Typ und die Parameter der seriellen Schnittstelle auf den entsprechenden Empfänger abgestimmt sein. Wenn Daten zwar empfangen, aber nicht angezeigt werden, erscheint eine Meldung am Bildschirm. Wenn die Einstellungen des Empfängers unbekannt und die Kommunikation nicht aufrecht erhalten werden kann, sollte der Empfänger resetet und die richtigen Einstellungen im Abschnitt *3.2* nachgesehen werden.

3.1.10 LABMON beenden

beenden

Vermeiden Sie Tastenkombinationen wie <Ctrl>+B oder <Ctrl>+C !

Diese können andere Programme beeinflussen, welche die gleichen Interrupts benützen wie LABMON. Kommt es zum Absturz, muss der PC in den meisten Fällen neu gebootet werden.

Taste <Q>

- Beenden Sie LABMON deshalb nur mit der <Q>-Taste. Diese beendet das Programm und Sie kehren zurück zum DOS-Prompt oder zu Windows

3.1.11 LABMON-Bildschirm

Bildschirm

Um die vielen neuen Funktionen der Zodiac-Familie zu unterstützen, sind die Bildschirm-Darstellung sowie die Tastenfunktionen gegenüber älteren Versionen überarbeitet worden. Es existieren 2 Typen von Bildschirm-Ausgaben: Das Main-Display (Haupt-Anzeige) und das Built-In-Test (BIT)-Display (Eingebaute Test-Anzeige, zeigt Ergebnisse versch. Tests an).
Abhängig vom Nachrichten-Protokoll-Typ (binary oder NMEA), werden die Daten-Bezeichnungen nur schwach (nicht aktiv) oder hell leuchtend (aktiv) dargestellt.

Dieses Main-Display erscheint, wenn *Zodiac binary*-Daten verarbeitet werden.

Main-Display

Das grau hinterlegte Menu lässt sich mit der Taste <N> ausblenden.

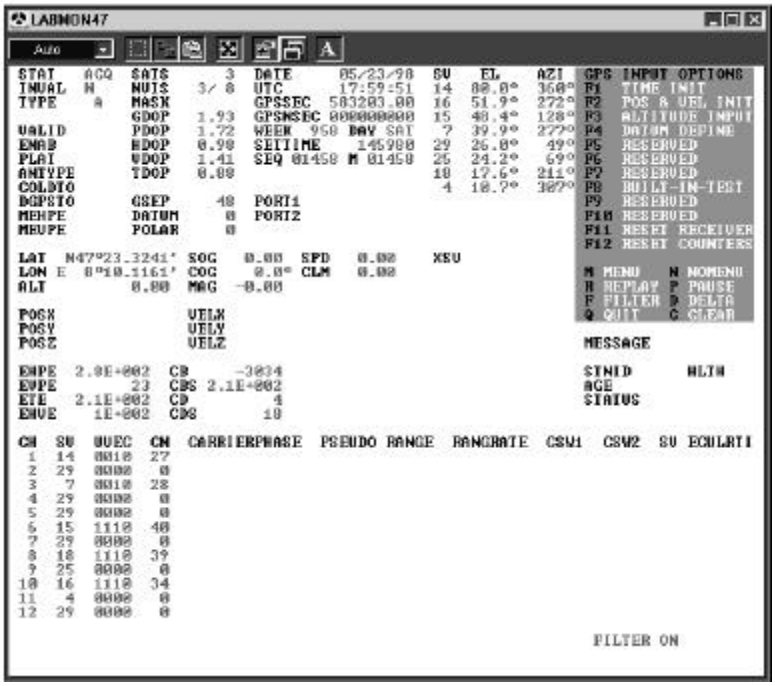


Bild 3.1-1: Main-Display im binary-Mode

Taste <N>

Der Bildschirm ist unterteilt in einen Daten- und einen Menü-Bereich. Da aber auch Daten im Menü-Bereich angezeigt werden können, lässt sich dieser mit der Taste <N> (Kein Menü) ausblenden. Der Datenbereich enthält Satelliten-Daten sowie Status-Informationen des binary oder NMEA-Protokolls.

Zusätzlich zu den Empfänger-Daten werden folgende nützliche Informationen angezeigt:
Anzahl der empfangenen Daten-Bytes, Anzahl der gesendeten oder empfangenen Nachrichten, Anzahl der Checksummen-Fehler und die Anzahl der Nachricht “no acknowledgements” (“keine Erkennung”).

Das NMEA-Display ist ebenfalls abhängig vom verwendeten Empfängertyp und kann sich somit in der Darstellung unterscheiden.

NMEA-Display

Das Menu wurde mit der Taste <N> ausgeblendet.

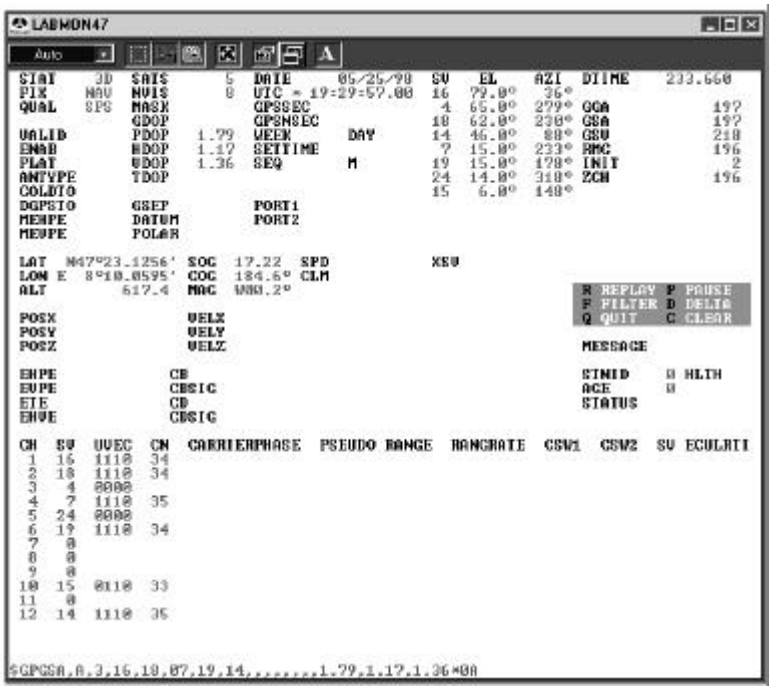


Bild 3.1-2: Main-Display im NMEA-Mode

Das BIT-Display zeigt in beiden Betriebsmodi Informationen über die Nachricht 1100 (Built-In-Test-Resultate). Mehr Informationen zu diesem Test sind im Abschnitt *3.4* ersichtlich.

Empfänger-Ausgabe-Nachrichten

Ausgabe-Nachrichten

LABMON decodiert und stellt die meisten verfügbaren Nachrichten dar. Daten im binary- oder im NMEA-Format können nicht gleichzeitig dargestellt werden, obwohl verschiedene Bildschirm-Bereiche dafür vorgesehen sind. Einige der binary- Daten werden sogleich in übliche Einheiten umgesetzt und entsprechend dargestellt (z.B. in Winkel-Grad anstelle von Radiant).

Einige der in den Nachrichten enthaltenen Daten werden zur weiteren Auswertung nicht benötigt und daher aus Platzgründen nicht am Bildschirm angezeigt.

Jede der am LABMON-Bildschirm ersichtlichen Datenbezeichnung ist mit einem Kurzbeschrieb und Angabe der Herkunft in Tabelle 1 in alphabetischer Reihenfolge aufgelistet. Für mehr Informationen ist das Handbuch des entsprechenden Empfängertyps zu konsultieren.

Tabelle 1: Ausgabedaten am Main-Display. Anhang, Seiten 20-23

Empfänger-Eingabe-Nachrichten

Eingabe-Nachrichten

LABMON macht ausgedehnten Gebrauch von Tastenkombinationen, um Empfänger- und Programm-Funktionen auszuführen, wobei diese vom verwendeten Protokoll-Typ abhängig sind und sich somit in der Ausführung unterscheiden können. Dazu muss zwischen Menü-Befehlen und Befehlen, welche im Online-Betrieb das Senden einer Nachricht zum Empfänger bewirken, unterschieden werden. Nach Betätigung einer Taste ist oft eine weitere Eingabe erforderlich. Diese Eingabe ist abhängig vom Protokoll-Typ und kann sich im Inhalt oder der einzugebenden Nummer unterscheiden.

Jeder Befehl der Funktionstasten, einzeln oder in Kombination mit der <Shift>, <Ctrl> oder <Alt>-Taste, ist ein Tabelle 2 aufgelistet. Ebenfalls ersichtlich sind die im Online-Betrieb gesendeten Nachrichten des jeweiligen Protokolles.

Tabelle 2:
Eingabe-Daten über die Tastatur
Anhang, Seiten 25-27

3.1.12 LABMON Menü-Tasten

Menü-Tasten

Viele Tastenfunktionen der älteren Versionen sind umbenannt oder in artverwandten Gruppen zusammengelegt worden.
Die folgenden Tastenfunktionen gelten hauptsächlich für den Rockwell binary-Modus. Wird eine Tastenfunktion vom jeweils aktuellen Protokoll nicht unterstützt, erscheint eine Fehlermeldung.

Im folgenden werden die einzelnen Tastenfunktionen unter Beibehaltung der englischen Bezeichnung kurz beschrieben:

- Wechselt das Vorwahltasten-Menü (<Shift>, <Ctrl> oder <Alt>) — *Menu Change (M)*
- Schaltet das Menü-Feld aus, um dahinter liegende Daten zu sehen — *No Menu (N)*
- Spielt eine Log-Datei ab — *Replay (R)*
- Stoppt die Empfänger-Datenausgabe am Bildschirm (Fortsetzung durch erneutes Betätigen der P- oder Space-Taste) — *Pause (P)*
- Setzt die Filtrierungsparameter (siehe *3.1.5*) — *Filter (F)*
- Schaltet zwischen Normal- und Delta-Positionsmodus um — *Delta (D)*
- Extrahiert Daten beim abspielen einer Log-Datei in eine vorformatierte, durch Tabulatoren begrenzte Plotdatei (für Ausdruck) — *Extract Data (E)*
- Erhöht (+) oder vermindert (-) die Geschwindigkeit, mit welcher die Daten einer abgespielten Log-Datei (Datenerfassungs-Datei) verarbeitet werden — *Speed of Replay (+, -)*
- Navigiert im Einzel-Schritt-Modus (>=vor, <=zurück) durch die einzelnen Nachrichten einer abgespielten Log-Datei — *Single Step in Replay (<, >)*
- Beendet LABMON — *Quit (Q)*
- Baut den Bildschirm neu auf — *Clear Screen (C)*
- Bricht den aktuellen Befehl ab — *Exit Prompt without changes (Esc)*

- Sendet die im 24-Stunden-Format eingegebene UTC- oder GMT-Zeit zum Empfänger. Die eingegebene Zeit sollte nicht mehr als eine Stunde von diesen Zeiten abweichen, wenn kein Kaltstart möglich ist. Bei einem Warmstart müssen Zeit und Position nicht initialisiert werden — *Time Initialisation (F1)*
- Sendet die Länge, Breite, Geschwindigkeit, Titel und die Höhe des ungefähren Benutzer-Standortes. Längen- und Breitengrad müssen mit dem ausgewählten Datum-Modus übereinstimmen und in Dezimalgrad eingegeben werden. Westliche Länge und südliche Breite müssen als negative Werte eingegeben werden. Die anzugebende Höhe ist die Höhenlage in Meter über dem Datum-Ellipsoid. Das Standard-Datums-Format ist WGS-84 (World Geodetic System) — *Position and Velocity Initialisation (F2)*
- Sendet einen Wert als vorgeschlagene Benutzer-Höhenlage. Dieser Wert wird nur bei einer 2D-Navigation benötigt. — *Set Altitude (F3)*
- Sendet die verwendeten Datums-Definitions-Parameter — *Datum Definition (F4)*
- Sendet einen Built-In-Test (BIT)-Befehl. Navigation und Satelliten-Suche sind während des Tests unterbrochen. Der Empfänger ist nach dem BIT resetet — *Built-In-Test Command (F8)*
- Sendet einen Reset-Befehl und setzt sämtliche Empfänger-Parameter inkl. den Nachrichtenzähler zurück — *Reset Receiver (F11)*
- Setzt den Nachrichten- und Fehlermeldungszähler auf Null — *Reset Counters (F12)*

3.1.12.1 <Shift> + Funktionstaste

- Sendet die seriellen Port-Einstellungen (siehe *3.1.5*) — *GPS I/O Port Settings (<Shift>+F1)*
- Auswahl des vom Empfänger verwendeten Nachrichten-Protokolls (binary oder NMEA) — *GPS Message Protocol Type (<Shift>+F2)*
- Sendet eine Erfassungs-Kontroll-Nachricht zum Empfänger. Bei Verwendung des *Zodiac binary*-Protokolls muss die Nachricht von Zeit zu Zeit aufgefrischt werden, um die Ausgabedaten zu erhalten — *Message Log Control (<Shift>+F3)*

HINWEIS: Die Werte “???” (beim NMEA-Protokoll) sowie “65535" (beim *Zodiac binary*-Protokoll) deaktivieren sämtliche Nachrichten. Der Empfänger gibt keine Nachrichten mehr aus, ausser wenn eine “Frage-Nachricht” (Query Message) gesendet wird, die den Empfänger reseten oder gar nicht angenommen werden. Um die benötigten Daten zur Auswertung eines NMEA-Protokolles trotzdem zu bekommen, muss die Tastenkombination <Shift>+F6 betätigt werden.

- Sendet eine “Frage-Nachricht” und erhält eine One-Time-Nachricht vom Zodiac-Empfänger zurück — *Message Query (<Shift>+F4)*

• Ermöglicht die Eingabe von ASCII-Text für das Senden einer vom NMEA-Protokoll nicht unterstützten Nachricht. Die geschriebene Nachricht muss mit einem $-Zeichen beginnen und den NMEA-Konventionen entsprechen; verwenden Sie Null-Felder falls erforderlich. LABMON ermittelt die Checksumme bevor die Nachricht übertragen wird	*NMEA Generic Message (<Shift>+F5)*
• Sendet eine Reihe von NMEA-Log-Nachrichten, um dem Empfänger die Ausgabe eines Nachrichten-Satzes zu ermöglichen, welcher die benötigten Daten für die Leistungsberechnung enthält	*NMEA Typical Message Request (<Shift>+F6)*
• Reserve	*(<Shift>+F7)*
• Fordert unbearbeitete Almanach-Daten an (Almanach=Kalendertafel mit Angaben der Planetenbewegungen). LABMON speichert die erhaltenen Daten in der Datei ALMANAC.GPS	*Store Almanac Data (<Shift>+F8)*
• Bittet um die Ausgabe von Pseudoranges (=scheinbare Entfernung zu einem Satelliten)	*Request Pseudoranges (<Shift>+F9)*
• Bittet um die Ausgabe von bestehenden oder neuen Ephemeriden-Daten, sofern vorhanden (Ephemeriden=astronomische Tafeln mit Angaben der vorausberechneten Orte der Gestirne für zeitlich konstante Abstände)	*Request Ephemeris Data (<Shift>+F10)*
• Bittet um die Ausgabe von aktuellen oder neuen Ionosphären- und UTC Zeit-Daten, sofern vorhanden (Ionosphäre=eine Schicht der Atmosphäre, welche die Ausbreitung von elektromagn. Wellen frequenzabhängig beeinflusst)	*Request Iono and UTC Data (<Shift>+F11)*
• Laden von unbearbeiteten Almanach-Daten von einer binären Almanach-Datei in den Empfänger	*Load Almanac Data (<Shift>+F12)*

3.1.12.2 <Ctrl>+Funktionstaste

• Sendet gültige Navigations-Lösungs-Parameter für die 2D-Navigation	*Navigation Validity Criteria (<Ctrl>+F1)*
• Auswahl der verwendeten Anwendungs-Plattform	*Platform Type (<Ctrl>+F2)*
• Sendet Navigations-Konfiguration-Parameter (u. a. für Höhenlage, Position und Filtermessungen)	*Navigation Configuration (<Ctrl>+F3)*
• Sendet einen Kaltstart-Befehl. Auch wenn der Kaltstart ermöglicht wird, geht der Empfänger erst dann in den Kaltstart-Modus, nachdem er versucht hat die nach seiner Positions- und Zeitangabe sichtbaren Satelliten zu finden. Einmal im Kaltstart-Modus, sucht der Empfänger automatisch den ganzen Himmel nach sichtbaren Satelliten ab	*Cold Start (<Ctrl>+F5)*
• Sendet eine Datum-Nummer zum Zodiac-Empfänger (für Datum-Namen und Nummern siehe *Anhang* C und D)	*Datum Select (<Ctrl>+F6)*

- Auswahl des Antennen-Typs (passiv oder aktiv)
  HINWEIS: Dieser Befehl beeinflusst nur die Signal-Parameter. Für die korrekte Einstellung am Empfänger ist Abschnitt *3.2* zu konsultieren — *Antenna Type (<Ctrl>+F7)*

- Sendet den verwendeten Erhebungswinkel. Winkel die tiefer liegen als der Horizont müssen als negative Werte eingegeben werden. Der Standard-Wert beträgt +10 Grad [°] — *Antenna Elevation Type (<Ctrl>+F8)*
- Auswahl der verwendeten Satelliten. Durch Eingabe einer Satelliten-Nummer kann dieser aktiviert oder deaktiviert werden (Toggle-Betrieb). Die Eingabe “0" bewirkt, dass alle Satelliten ausgeschaltet werden. Deaktivierte Satelliten werden mit ”**XSV**“ in der Satelliten-Liste am Main-Display angezeigt. — *SV Selection (<Ctrl>+F9)*
- Sendet einen Differential GPS-Befehl — *DGPS Control (<Ctrl>+F10)*
- Sendet Power-Management-Parameter für die Leistungs-Erhaltung — *Power Management (<Ctrl>+F11)*

3.1.12.3 <Alt>+Funktionstaste

- Ändert die Einstellungen der seriellen Schnittstelle (siehe *3.1.5*). Nachdem die Einstellungen vorgenommen wurden, zeigt LABMON den gewählten COM-Port, die PC-Interrupt-Nummer und die Port-Adresse am unteren Bildschirmrand an — *LABMON I/O Port Settings (<Alt>+F1)*
- Auswahl des Protokoll-Typs (*Zodiac binary*, *Zodiac NMEA*, *NavCore binary* oder *NavCore NMEA*) — *LABMON Message Protocol Type (<Alt>+F2)*
- Setzt die Referenz-Position als Standard-Wert für Positions-Initialisierung (F2) und als Referenz-Wert für die Berechnung der Delta-Position im Delta-Mode — *Reference Position (<Alt>+F3)*
- Ändert die Hintergrundfarbe des Datenbereichs im Main-Display — *Main Display Background Color (<Alt>+F4)*
- Ändert die Farbe der Datenattribute im Datenbereich des Main-Displays — *Main Display Text Color (<Alt>+F5)*
- Ändert die Farbe der Daten im Datenbereich des Main-Displays — *Main Display Data Color (<Alt>+F6)*
- Ändert die Hintergrundfarbe des Menüs im Main-Display — *Menu Background Color (<Alt>+F7)*
- Ändert die Farbe der Funktionstasten-Bezeichnungen im Menübereich des Main-Displays — *Menu Function Key Color (<Alt>+F8)*
- Ändert die Farbe der Funktionstasten-Beschreibungen im Menübereich des Main-Displays — *Menu Description Color (<Alt>+F9)*

3.1.13 Features, Hinweise und Bemerkungen

Nachrichten-Satz

Die Familie der Zodiac-Empfänger stellt einen umfassenden Nachrichten-Satz zur Verfügung. In Form des OEM-Produktes beinhaltet dieser Satz die typischen und am häufigsten benötigten Daten. Wenn der Empfänger für spezielle Auswertungen verwendet wird, kann es jedoch leicht vorkommen, dass der Benutzer einen anderen Nachrichten-Satz wählt.

Wenn der Empfänger in zeitlich periodischen Abständen zu viele Ausgabe-Meldungen erzeugt, kann es vorkommen, dass der Empfänger während dem Erhalt einer neuen Meldung einen Checksummenfehler ausgibt.

Baud-Rate

Der Benutzer muss sicherstellen, dass die eingestellte Baud-Rate die Nummer und Grösse der angeforderten Nachrichten unterstützt.

Nachrichten-Erfassung

Der Benutzer sollte nicht benötigte Nachrichten ausschalten. Die Häufigkeit der Datenausgabe ist durch die Zeit bestimmt, die Aktualisierung der Ausgabedaten soll jedoch nach Wunsch erfolgen. Um eine einmalige, zu einem beliebigen Zeitpunkt gewünschte Nachricht zu erhalten, ist die “Frage-Nachricht” zu verwenden.
Um nur bei Änderung oder neu erhältlichen Daten eine Ausgabe zu erhalten, ist eine Log-Nachricht mit der Aktualisierungs-Option zu verwenden. Um in periodischen Abständen eine Ausgabe zu erhalten, ist eine Log-Nachricht mit der Zeitausgabe-Option zu verwenden.
Um sämtliche Nachrichten im NMEA-Mode auszuschalten, ist eine Log-Nachricht mit “???” als Nachrichten-Bezeichner, im *Zodiac binary*-Mode die Zahl “65535" (=FFFFh) zu verwenden.

Für Zodiac-Empfänger sind die folgenden Änderungen der Standard-Einstellungen empfohlen:

- Aktivieren der Nachricht 1012 für die Aktualisierungs-Ausgabe
- Aktivieren der Nachricht 1102 für eine Nachrichten-Ausgabe pro Sekunde
- Aktivieren der Nachricht 1005 für eine Nachrichten-Ausgabe pro Sekunde für DGPS-Berechnungen

Nachrichten-Zähler

Benützen Sie die Nachrichten-Zähler um die Nummer und Häufigkeit der ausgehenden Nachrichten anzuzeigen und den Überblick über die eingehenden und bereits gesendeten Nachrichten zu behalten. Wenn andere Daten durch die überfüllte Liste dieser Nachrichten blockiert werden, müssen die Nachrichten-Zähler zurückgesetzt werden (Reset).

Bildschirm säubern

Säubern Sie den Bildschirm nach einer Änderung der Nachrichten-Ausgabe-Konfiguration. Dies entfernt die nicht mehr aktuellen Daten vom Bildschirm.

Daten-Erfassung

Um erfasste Daten in Ruhe anzuschauen, benützen sie die Replay-Funktion. Damit können Sie interessante Zeitabschnitte oder bestimmte Ereignisse im Einzel-Schritt-Betrieb noch einmal überprüfen.

Daten-Reduzierung

Die Extract-Funktion schreibt Daten aus einer Log-Datei in eine eine vorformatierte, durch Tabulatoren begrenzte Plotdatei. Die Einstellung der Filter-Parameter bestimmt, welche Datenpunkte in diese Datei geschrieben werden. Die Auswertung von verschiedenen Filter-Einstellungen ermöglicht dem Benutzer die Ermittlung der wichtigsten Kriterien, um in der entsprechenden OEM-Anwendung die besten Ergebnisse zu erzielen.

DGPS-Operationen

Wann immer DGPS-Funktionen ausgeführt werden, wird empfohlen die LABMON Erfassungs-Funktion zu verwenden und die RTCM SC-104-Daten zusammenzuhalten, so dass diese auf Format und Vollständigkeit überprüft werden können (siehe Hilfsprogramm RTCMCHK auf der Diskette). Einige DGPS-Referenzstationen generieren Ausgangssignale, die nicht dem RTCM-Standard entsprechen. Rockwell GPS-Empfänger sind zwar so konzipiert, dass sie diese Signale tolerieren, doch in gewissen Fällen ist ein Unterbruch und somit Verlust von DGPS-Operationen nicht zu vermeiden. Diese Störungen können jedoch durch die oben erwähnte Prüfung der ankommenden, unbearbeiteten Daten weitgehend eliminiert werden.

Delta-Position

Der Delta-Positions-Modus ist nützlich um Änderungen relativ zu der gewählten Referenz-Position anzuzeigen. Die Referenz-Position, die PC-Zeit sowie der Zeitzonen-Offset werden als Standard-Eingaben für die Empfänger-Initialisation verwendet.

3.2 Development Kit Setup & Operation

JUPITER
Development
Kit
②

3.2.1 Einleitung

Einleitung

Dieser Abschnitt beschreibt die Konfiguration des Development Kit und des Empfängers sowie der Einsatz in der Entwicklungsumgebung.
Der GPS-Empfänger ist in einem Gehäuse untergebracht. Es besitzt Eingabe/Ausgabe-Anschlussklemmen, Status-LED's und DIP-Schalter zur Konfiguration wie in *Bild 3.2-1* ersichtlich.

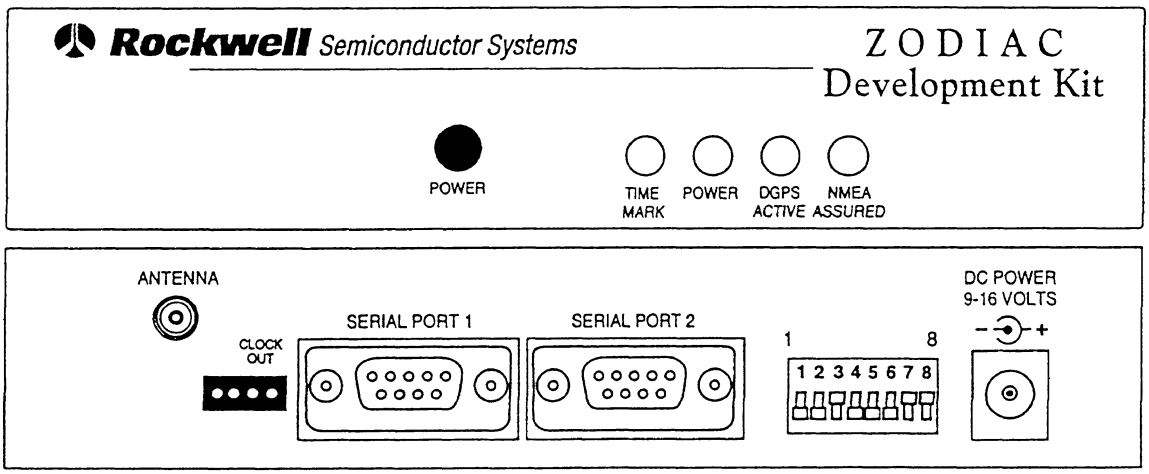


Bild 3.2-1: Vorder- (oben) und Rückansicht (unten) des Development Kit

3.2.2 Mitgelieferte Geräte

Mitgelieferte Geräte

Das Development Kit beinhaltet nach *Bild 3.2-2* die folgenden Komponenten:

- "Jupiter"-Empfänger, Spannungs-Versorgung, Spannungsregler und RS-232-Treiber im Gehäuse (1)
- Aktive Antenne mit männlichem SMA-Anschlussstecker (2)
- Zwei RS-232 Kabel mit weiblichen DB-9-Anschlussstecken an beiden Enden (3)
- 120V AC, 60 Hz/12 VDC-Netzteil (hier 230V AC, 50 Hz/12 VDC) (4)
- 12 VDC-Netzteil für Anschluss am Zigarettenanzünder im Auto (5)
- Zodiac GPS Receiver Family Designer's Guide (6)
- LABMON Operation Guide for the "Jupiter" GPS Receiver (7)
- Development Kit Setup and Operation Guide for the "Jupiter" GPS Receiver (dieser Abschnitt) (8)
- Zodiac LABMON-Software auf einer 3.5"-Diskette (9)

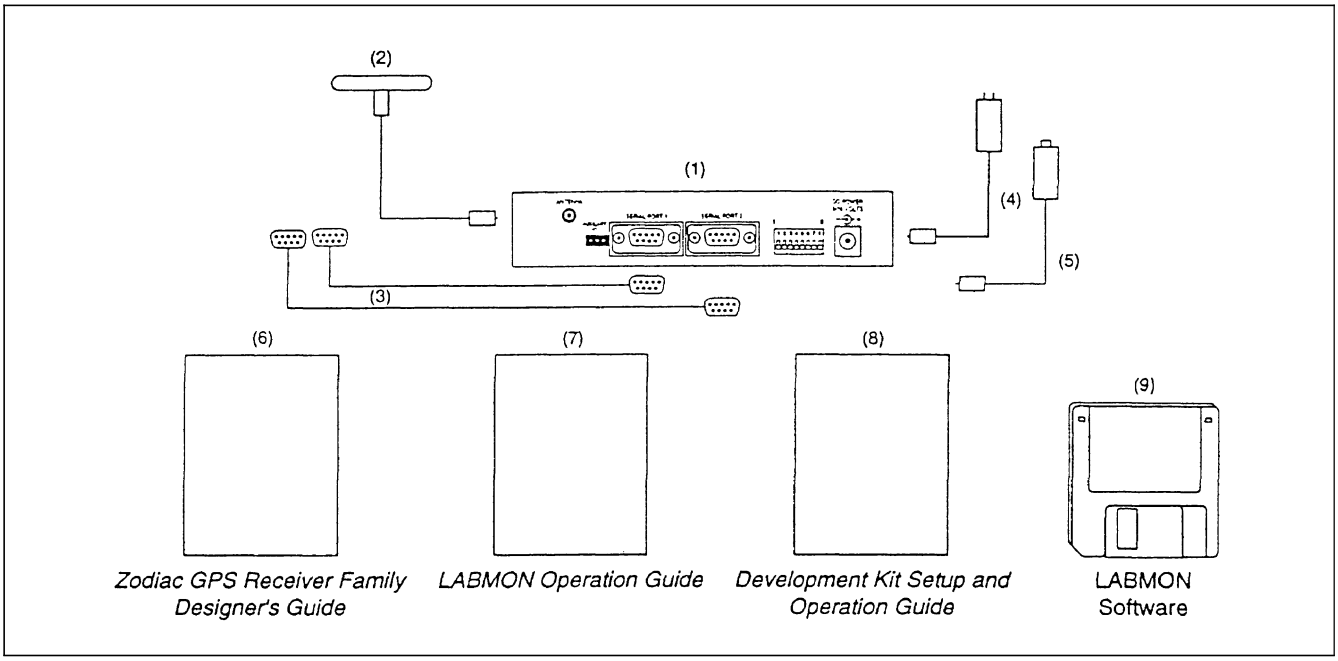


Bild 3.2-2: Development Kit-Komponenten

3.2.3 Benötigte Geräte

Benötigte Geräte

Folgende Mindestausstattung von zusätzlichen Geräten ist für den Betrieb des OEM-Moduls ebenfalls erforderlich:

- IBM-AT-kompatibler PC mit einer freien seriellen Schnittstelle, einem 3.5"-Diskettenlaufwerk, min. 640 KB RAM und DOS Version 3.0 oder grösser.
- Ein Laptop oder portabler PC ist empfehlenswert, da im Freien die Satelliten-Signale leichter zu empfangen sind.
- Farbbildschirm mit VGA oder EGA oder Monochrom-Monitor mit VGA.
- RTCM-Datenquelle (nur erforderlich für den Test der DGPS-Fähigkeit).
- Kabel für den Anschluss der RTCM-Datenquelle an der zweiten seriellen Schnittstelle und den Empfänger-Hilfsport für gleichzeitigen Betrieb der RTCM SC-104 Datensammlung und DGPS-Operationen. Empfohlen ist ein Kabel mit 3 Anschlüssen.

3.2.4 Beschreibung

Beschreibung

- Die besonderen Eigenschaften des Development Kit sind vor allem zwei RS-232-fähige serielle Schnittstellen, wählbare Vorspannungen für die aktive Antenne, konfigurierbare Ports, Nachrichten-Protokolle und einen internen, flexiblen “Keep-Alive”-Back-up Power-Mode für das SRAM und die Niederleistungsquelle.

- Ebenso verfügt das Kit über CMOS/RS-232-Pegelumsetzer für die Bereitstellung der Daten an den 9-poligen D-SUB-Steckern (fem). Der erste serielle Port kann für das Senden und Empfangen beider Nachrichten-Protokolle (binary und NMEA) verwendet werden. Der zweite, sogennante Hilfsport, ist nur für den Empfang von RTCM SC-104-DGPS-Korrekturdaten vorgesehen und kann keine Daten ausgeben.

Pegelumsetzer, Serielle Schnittstellen

- Eine Hilfsanschlussstecker am Kit gewährt dem OEM-Prozessor den Zugriff auf Impulse von 1Hz und 10 KHz für eine allfällige Synchronisation.

Hilfsanschlussstecker

- Eine Anzahl von Test-Pins auf dem Board verschafft Zugang zu den Ein- und Ausgabesignalen.

Test-Pins

- Mit Hilfe der vorhandenen Jumper-Blocks kann der Benutzer das Kit entsprechend konfigurieren.

Jumper-Blocks

- Dazu ist ein externer DIP -Schalter vorhanden, welcher den elektronischen Bus-Schalter kontrolliert.

DIP -Schalter

- Eine Spannungsreglerschaltung liefert die benötigte konstante DC-Spannung.

Spannungsregler

- Zur Spannungsversorgung kann entweder ein üblicher AC/DC-Netzadapter (+12 VDC) mit einem Ausgangstrom von ca. 500 mA oder ein Adapter zur Speisung des Gerätes vom Zigarettenanzünder im Auto verwendet werden. 4 LED's an der Frontplatte zeigen verschiedene Status-Informationen an. Zum reseten des Empfängers dient ein Taster, der unter der Abdeckung angebracht ist.

Spannungsversorgung, Status-LED's

- Zum Set gehören die ebenfalls mitgelieferten Kabel mit 9-poligen D-SUB-Steckern und eine aktive Antenne mit einem 8 Fuss (ca. 2.5m) langen RF-Kabel. Die Dämpfung des Kabels beträgt ca. 3dB.

RS-232 Kabel, Antenne

Die mitgelieferte Antenne muss auf +5 VDC eingestellt sein. Bei korrekter Einstellung der Schalter an der Rückseite des Kit's kann auch eine Antenne für +12 VDC verwendet werden.

Versichern Sie sich, dass die Schalter für die Speisespannung der Antenne schon vor dem Anschluss der Antenne richtig eingestellt sind. Eine für +5 VDC ausgelegete Antenne wird bei Anschluss an +12 VDC beschädigt !

- Mit Hilfe eines Drucktasters wird das "Jupiter"-Board EIN- und AUSgeschaltet. Ein weiterer Drucktaster ist für den System-Hardware-Reset des Empfängers vorgesehen und nur zugänglich, wenn man die Abdekkung entfernt.

Taster EIN, AUS

- Ein "Super-Kondensator" (Supercab) versorgt die SRAM- und RTC-Einheit bis zu einer Stunde nach Entfernen der Spannungsversorgung mit Strom. Diese Besonderheit stellt sicher, dass bei einem Warmstart die letzte Position, die aktuelle Zeit und die Ephemeriden-Daten der Satelliten aufrecht erhalten werden.

"Super-Kondensator"

Externe Back-up-Speisung

- Ebenso gibt es eine Einrichtung zur externen Back-up-Speisung für die RTC-Einheit. Wenn die Jumper und DIP-Schalter entsprechend gesetzt sind, wird die aktuelle Zeit auch bei entleertem Supercab aufrechterhalten. Dies ist von Vorteil bei einem Warmstart, wenn das System länger als eine Stunde vom Supercab betrieben wurde und der Inhalt des SRAM's nicht mehr gültig ist. Diese externe Back-up-Speisung erhält die RTC-Einheit bis die DIP-Schalter betätigt, der entspr. Jumper oder die +12 VDC-Qulle entfernt wird.

3.2.5 Konfiguration

Konfiguration

Wenn die Abdeckung infolge Konfiguration entfernt werden muss, ist darauf zu achten, dass durch elektrostatische Aufladungen keine Schaltungsteile beschädigt werden !

Das Nichtbeachten dieser Vorsichtsmassnahmen kann zu dauerhaften Beschädigungen der Einheit führen !

Das Kit muss vor dem Anschluss an den PC richtig konfiguriert werden.

Das "Jupiter"-Board besitzt eine Bank von 8 DIP-Schaltern, 3 Jumper-Blocks und einen einzelnen, zweistufigen Schalterblock. Die Position jeder Konfigurations-Einheit ist in *Bild 3.2-3* ersichtlich.

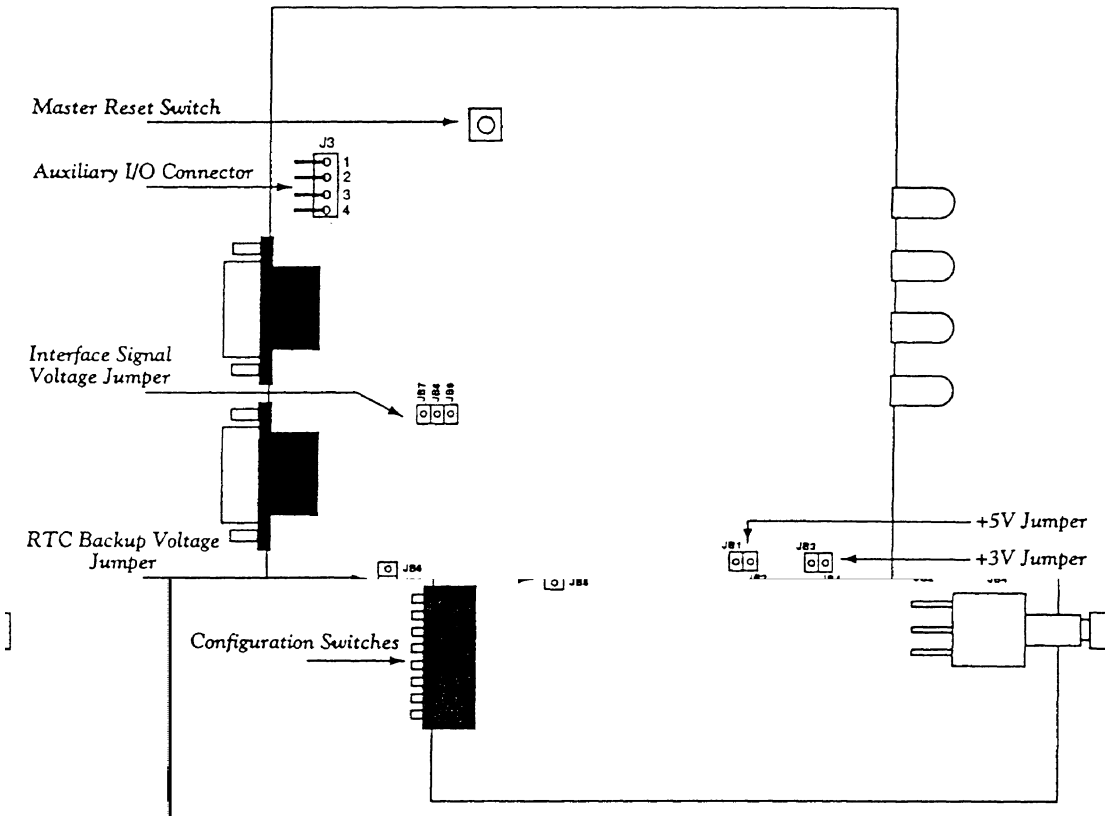


Bild 3.2-3: Positionen der Konfigurations-Einheiten

3.2.5.1 DIP-Schalter

DIP-Schalter

Die ursprünglichen Konfigurationseinstellungen befinden sich auf der Rückseite des Kit's und sind in *Bild 3.2-4* illustriert. Die Darstellung zeigt die normalen Standardpositionen.

Nr.	Schalter/Funktion	ON	OFF
1	Reserve	-	-
2	Haupt-Port-Kontrolle	NMEA	Rockwell binary oder NMEA, abhängig von S3
3	EEPROM aktiv/inaktiv	EEPROM inaktiv	EEPROM aktiv
4	Reserve	-	-
5	I/O-Kontrolle	aktiviert I/O-Signale	deaktiviert I/O-Signale
6	RTC-Kontrolle	RTC-Back-up-Speisung aktiv	RTC-Back-up-Speisung nicht vorhanden
7	Auwahl Spannung Vorverstärker	+5 VDC für aktive Antenne	+ 12VDC für aktive Antenne
8	Speisung Vorverstärker aktiv	Speisung Vorverstärker EIN für aktive Antenne	Speisung Vorverstärker AUS für passive Antenne

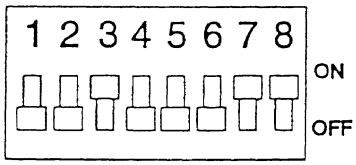


Bild 3.2-4: Development Kit Konfigurations-Schalter

Die mit dem Kit gelieferte aktive Antenne darf nur mit +5 VDC betrieben werden, bei einer passiven Antenne muss der Schalter auf 0 VDC eingestellt sein. Vor Betätigung der Schalter muss die Einheit von der Stromversorgung getrennt werden !
Die Einstellungen für eine aktive Antenne müssen 0, +5 oder +12 VDC sein, um mögliche Beschädigungen an der Einheit zu vermeiden !

Schalterstellungen und Funktion

Ist Reserve und muss auf OFF stehen *Schalter 1*

Mit Schalter 2 wird das Empfänger-Ausgabe-Protokoll eingestellt. *Schalter 2*

- ON = NMEA-0183-Protokoll mit den Hauptport-Einstellungen 4800 bps (bits per second), keine Parität, 8 Datenbits und 1 Stop-Bit.
  OFF: In dieser Stellung wird das Ausgabe-Protokoll und andere Parameter durch die Stellung von Schalter 3 wie folgt festgelegt:
- Schalter 2 = OFF und Schalter 3 = ON: Standard ROM-Einstellungen erzeugen Ausgabedaten im Rockwell binary-Format mit den Hauptport-Einstellungen 9600 bps, keine Parität, 8 Datenbits und 1 Stop-Bit.
  Schalter 2 = OFF und Schalter 3 = OFF: Das Ausgabe-Protokoll wird durch die gespeicherten Daten im SRAM oder EEPROM bestimmt. Dazu werden die Einstellungen der letzten Betriebs-Session übernommen. *Schalter 2 und Schalter 3*

Aktiviert oder deaktiviert das elektrisch lösch- sowie programmierbare Read only Memory (EEPROM). *Schalter 3*

- ON = deaktiviert das EEPROM und aktiviert die Standard-Einstellungen im ROM (Ausgabe-Protokoll, Hauptport-Einstellungen und Standard-Nachrichtensatz).
  OFF = aktiviert die benutzerdefinierten Einstellungen im EEPROM. Dies ist die normale Einstellung, um Daten wie Zeit-Parameter, letzte Position und Empfänger-Einstellungen aus dem EEPROM zu benützen.

- Wenn Schalter 3 auf ON steht (EEPROM = inaktiv), wird bei jedem Ein bzw. Ausschalten oder Reset eine neue Initialisierung (Time-To-First-Fix = TTFF) durchgeführt, da die Standard-Initialisierungs-Parameter aus dem ROM verwendet werden.
  HINWEIS: Für die Ausgabe der Reultate eines Built-In-Test (BIT) muss Schalter 3 in der OFF-Position stehen.

Ist Reserve und muss auf OFF stehen *Schalter 4*

Kontrolliert die EIN- und AUSGABE-Signale *Schalter 5*

- ON = aktiviert alle EIN- und AUSGABE-Signale
  OFF = deaktiviert alle EIN- und AUSGABE-Signale

Legt fest, ob eine RTC-Back-up-Speisung erwünscht ist. *Schalter 6*

- ON = aktiviert Back-up-Speisung solange, wie die +12 VDC anliegen
  OFF = deaktiviert Back-up-Speisung

Danach speist Der “Supercab” die RTC-Einheit (unabhängig von obiger Schalterstellung) solange, bis er völlig enleert ist.

Stellt die Speisespannung für eine aktive Antenne ein, wenn Schalter 8 = ON *Schalter 7*

- ON = +5 VDC (Bei Verwendung der mitgelieferten Antenne)
  OFF = +12 VDC

Auswahl zwischen einer aktiven oder einer passiven Antenne *Schalter 8*

- OFF = passive Antenne
  ON = aktive Antenne

3.2.5.2 Jumper-Blocks

Jumper-Blocks

Jumper JB5 und JB6

Wird der Jumper-Block mit den Pin-Bezeichnungen JB5 und JB6 in dieser Stellung belassen, versorgt die Back-up-Speisung die RTC-Einheit mit Strom, auch wenn die Speisespannung entfernt wird. Bei Entfernung des Jumpers wird die Back-up-Speisung daktiviert. Um in diesem Fall die RTC-Einheit mit einer externen Quelle zu speisen, muss diese an Pin JB6 angelegt werden (siehe auch Abschnitt *3.3*).

Jumper JB1 und JB2

Um den Empfänger über den Schnittstellen-Anschluss mit +5 VDC zu speisen, muss sich ein weiterer Jumper über den beiden Pins JB1 und JB2 befinden. Bei Speisung mit einer externen Quelle muss der Jumper entfernt und die Spannung am Pin JB2 angelegt werden (siehe auch Abschnitt *3.3*).

Jumper JB7, JB8 und JB9

Mit einem weiteren Jumper-Block (JB7, JB8 und JB9) kann entweder +3.3 oder +5 VDC für das Empfänger-I/O-Interface eingestellt werden. Wird das "Jupiter"-Board mit dem Development-Kit betrieben (+5 VDC), muss der Jumper über den Pins JB7 und JB8 liegen

Zusätzlich zu den beiden Ports ist ein 4-Pin-Hilfs-I/O-Anschlussstecker vorhanden, der folgende Signale liefert:

- **Pin 1:** K10. 10 KHz UTC Ausgangs-Referenz-Signal
- **Pin 2:** TM (Time Mark). 1 pulse per second (1 PPS = 1 Hz) CMOS-Ausgangs-Signal erzeugt von einem 74LS04.
- **Pin 3:** Nicht angeschlossen
- **Pin 4:** GROUND

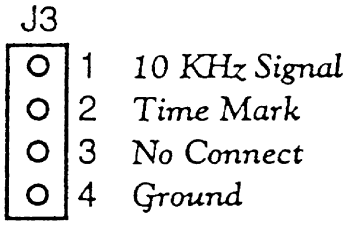


Bild 3.2-5: Hilfs-Anschlussstecker

Die beiden Referenz-Signale von 10 und 1 KHz können vom OEM-Prozessor zur Synchronisation verwendet werden.

Vor dem Einschalten der Speisespannung die Antennen-Schalter und Einstellungen überprüfen !
Aufgrund des kleineren Widerstandes einer passiven Antenne kann bei einer falschen Einstellung ein unzulässig hoher Strom entstehen, der den Empfänger, das Kit oder die Antenne beschädigt.

Das Nichtbeachten dieser Vorsichtsmassnahmen kann zu dauerhaften Beschädigungen der Einheit und/oder der Antenne führen !

Störeinflüsse

Die Antenne kann nur Signale von sichtbaren Satelliten Empfangen. Deshalb ist der Standort für optimalen Empfang so zu wählen, dass möglichst viel freier Himmel sichtbar ist.
Die Signale werden durch Häuser, Bäume Metal- und Ziegelkonstruktionen abgeschwächt oder sogar ganz verdeckt. Bei stationären Einrichtungen sollte man zudem beachten, dass die Signale von Metall oder beschichtetem Glas reflektiert werden und durch den längeren Weg das Signal verfälschen. Um bestmögliche Leistungen zu erzielen, sollte der Empfänger im Freien getestet werden (ein Laptop oder portabler PC ist empfehlenswert).

Test Equipment Setup

Test Equipment Setup
RTCM SC-104 Korrekturdaten

Wenn RTCM SC-104 Korrekturdaten zur gleichen Zeit erfasst und zum Empfänger geschickt werden sollen, ist ein spezielles Kabel mit 3 Anschlusssteckern nötig. Von der RTCM SC-104 Korrekturdatenquelle geht ein Kabel zum Empfänger wie auch zu einem zweiten seriellen Port am PC. In dieser Anordnung sind Daten nur zu empfangen, wenn LABMON im **Zusatz-Mode 2** gestartet wird (siehe *3.1.6*).

In der Regel macht es Sinn, RTCM-Korrekturdaten erst zu Erfassen und danach auszuwerten. Das bereits schon angesprochende Programm RTCMCHK ist ein nützliches TOOL, um die empfangen RTCM-Daten auf Vollständigkeit zu prüfen. DGPS-Daten können nur dann erfasst werden, wenn das entsprechende Kabel am Hilfsport angeschlossen ist.

Wie die einzelnen Komponenten miteinander verbunden werden zeigt Bild 3.2-6:

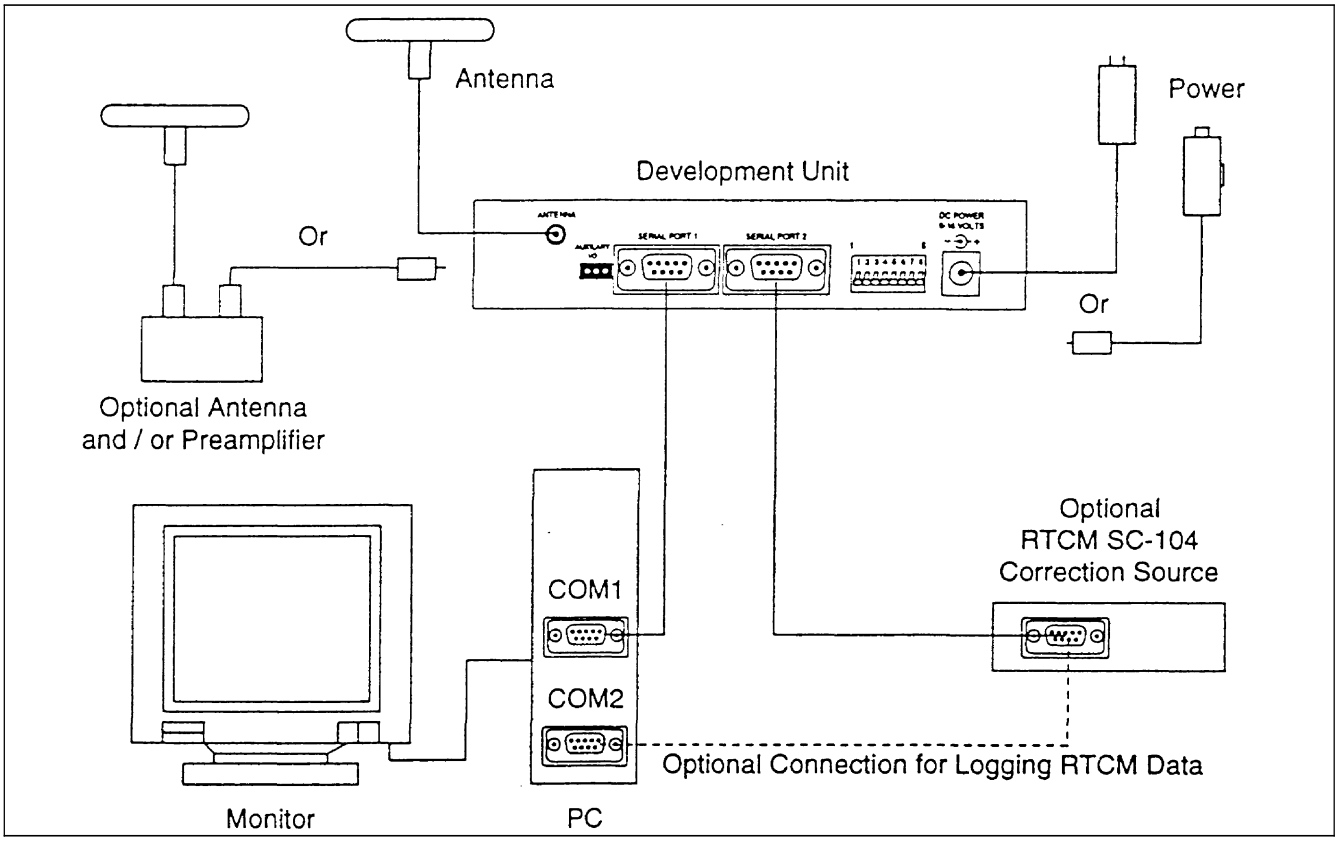


Bild 3.2-6: Development Kit Test Equipment Setup

Beachten Sie, dass das Development Kit an den Ports COM1-COM4 und die RTCM-Korrekturdatenquelle entweder direkt am Hilfsport des Empfängers, einem freien Port am PC oder an beiden angeschlossen werden kann.

Zustand-LED's

Sobald das Development Kit mit Spannung versorgt wird, ist der Empfänger betriebsbereit und zeigt dies entsprechend an:
Die POWER-LED leuchtet und die TIME MARK-LED blinkt mit einer Frequenz von 1 kHz.

Zudem leuchtet die NMEA ASSURED-LED, wenn für den Hauptport das NMEA-Protokoll gewählt wurde. Wenn Daten über den Hilfsport empfangen werden, leuchtet die DGPS ACTIVE-LED.

Um den Empfänger auszuschalten, muss erneut der Drucktaster betätigt werden. Danach versorgt der "Supercab" das SRAM sowie die RTC-Einheit noch ca. eine Stunde mit Strom.
Die POWER LED muss erloschen sein, bevor Änderungen an den Konfigurationseinstellungen vorgenommen werden. Zudem werden die Änderungen erst nach einem erneuten Start wirksam.

3.3 Zodiac-Family Designer's Guide

ZODIAC Receiver Family

③

3.3.1 Einleitung

Einleitung

Rockwells Zodiac-Familie der GPS-Empfänger sind Single-Board, 12-Kanal-parallel-Empfänger, welche u. a. auch für OEM-Applikationen entwickelt wurden. Sie suchen fortlaufend den Himmel nach sichtbaren Satelliten ab und stellen so kontinuierlich und fehlerfrei Navigationsdaten zur Verfügung.
Sie sind mit zwei hochintegrierten, von Rockwell designten IC's inklusive dem Zodiac Chipsatz bestückt: Dem "Gemini/Pisces" MonoPac™ und dem "Scorpio" Digital-Signal-Prozessor (DSP). Dank der Verwendung dieser Custom-Einheiten findet der Empfänger auf einem Print von 28 cm^2 platz.

3.3.2 Kurze Produkte-Übersicht

Produkte-Übersicht

Zum Betrieb benötigt der Empfänger eine konstante Gleichspannung (DC) und ein GPS-Signal von einer aktiven oder passiven Antenne. Um die Initialisierungszeiten nach einer kurzen Stromunterbrechung möglichst klein zu halten, werden das SRAM sowie die RTC-Einheit von einer Energie-Speichereinheit unterhalten. Die zwei unabhängigen seriellen Ports (RS-232) erlauben die Ausgabe von NMEA-0183 sowie Rockwell binary-Protokollen (Haupt-Port) und den Empfang von RTCM SC-104-Korrekturdaten (Hilfs-Port). *Bild 3.3-1* illustriert die interne Architektur.

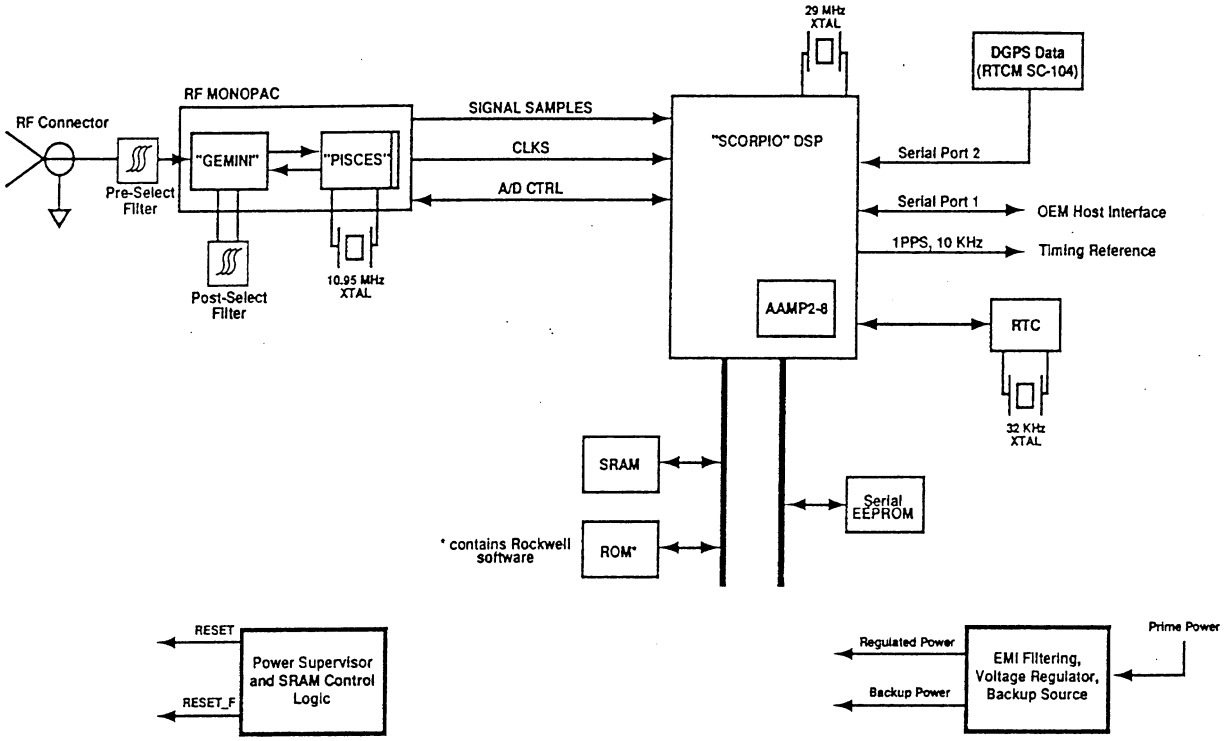


Bild 3.3-1: Interne Architektur des Zodiac-Empfängers

Bild 3.3-2 zeigt eine mögliche Integration des Empfängers in eine OEM-Applikation.

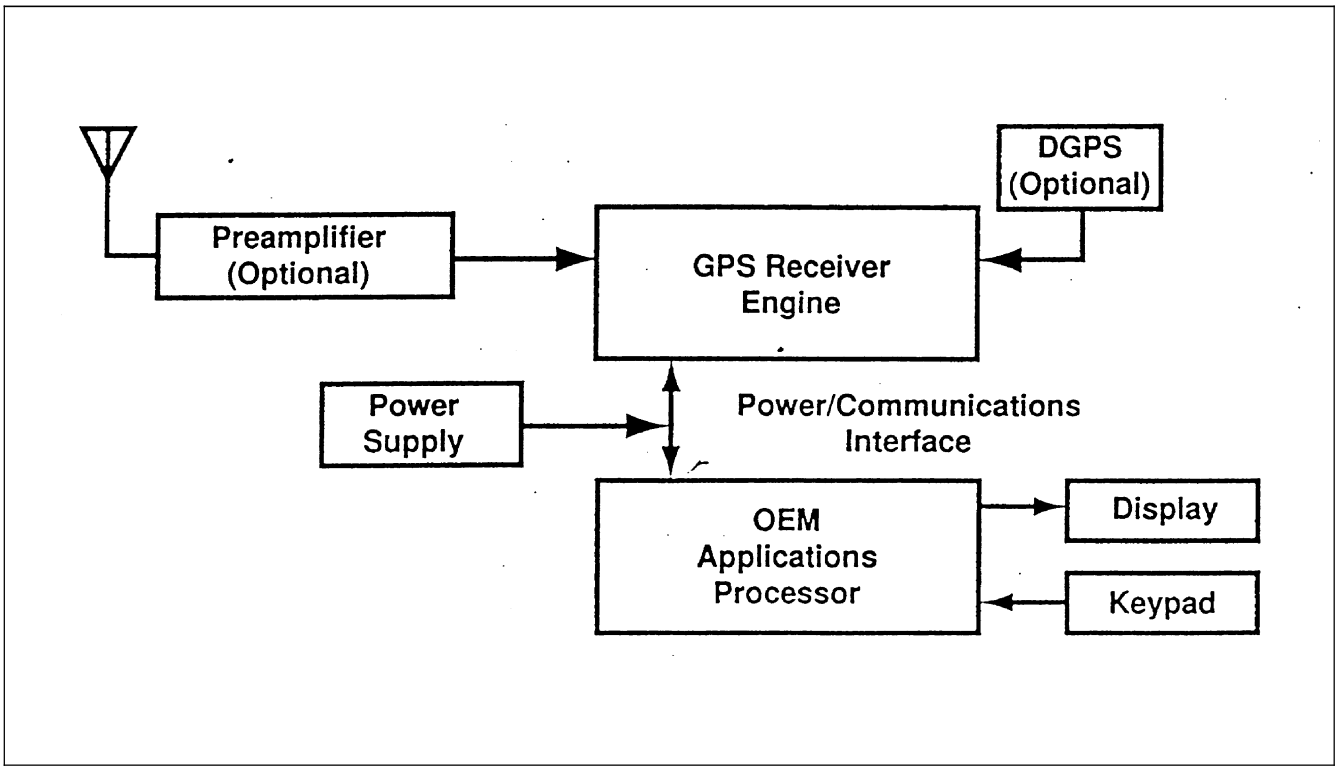


Bild 3.3-2: Integration des Zodiac-Empfängers in eine mögliche OEM-Applikation

3.3.2.1 Features

Features

Zusammenfassend und stellvertretend für sämtliche physikalischen und Betriebsunterstützende Eigenschaften dieser Empfänger-Familie sei die folgende Auflistung:

- OEM-Produkt-Einheiten werden bei der Applikations-Entwicklung vollumfänglich unterstützt
- Einer der kleinsten und kompaktesten Empfänger (70 x 40 x 12 mm)
- Zwölf parallel arbeitende Empfangs-Kanäle
- Power-Management zur Verlängerung der Batterie-Lebensdauer
- Unterstützt NMEA-0183 Datenprotokoll
- Direkteingang für parallel zur Positionsbestimmung empfangende RTCM SC-104 Korrekturdaten, welche sogleich in die Berechnungen miteinfliessen
- Unterstützt Geschwindigkeiten bis zu 950 m/s (= 260 km/h)

- Kann mit passiver oder aktiver Antenne betrieben werden
- Bestmögliche Positionsbestimmung mit dem Standard Positioning Service (SPS)
- Low power, +3V Standby-Betriebsmodus
- Steigerung der TTFF (=Time-To-First-Fix) nach dem Einschalten, weil aktuelle Daten von einer Energie-Speichereinheit über eine gewisse Zeit "am Leben" erhalten werden
- Verträgt harte Stösse und Vibrationen
- Automatischer Höhenlage-Halte-Modus beim Wechsel von Drei- zu Zweidimensionaler Navigation
- Automatischer Kaltstart-Prozess, wenn keine Eingaben durch den Benutzer erfolgen
- Maximale Betriebs- und Konfigurationsmöglichkeiten durch Benutzer-Befehle
- Kann externe Initialisierungsdaten annehmen
- Drei-Satelliten Navigations-Erfassung von Beginn weg
- Durch den Benutzer auswählbare Satelliten zur Positionsbestimmung
- Durch den Benutzer einstellbaren Erhebungswinkel
- Standard Mikrominiature Koaxial RF-jack-Buchse
- Standard 2x10-Pin Anschlussstecker

3.3.3 GPS-System-Beschreibung

GPS-System-Beschreibung

Das NAVSTAR GPS ist ein satellitengestütztes Echtzeit-Radio-Navigationssystem, entwickelt und betrieben durch das amerikanische Verteidigungsministerium **D**epartment **o**f **D**efense (DoD). Es ist ein absolut arbeitendes, passives System, das

- weltweit an jedem Ort
- zu jeder Zeit und
- bei jedem Wetter

verfügbar ist und die Benutzer kontinuierlich mit den erforderlichen Daten zur

- Positions-
- Geschwindigkeits- und
- Zeitbestimmung

versorgt.

Im Wesentlichen umfasst es die folgenden drei Hauptgruppen

- Die Satelliten als Sendernetz — *Space Segment*
- Die Boden- oder Kontrollstationen — *Control Segment*
- Den Empfänger — *User Segment*

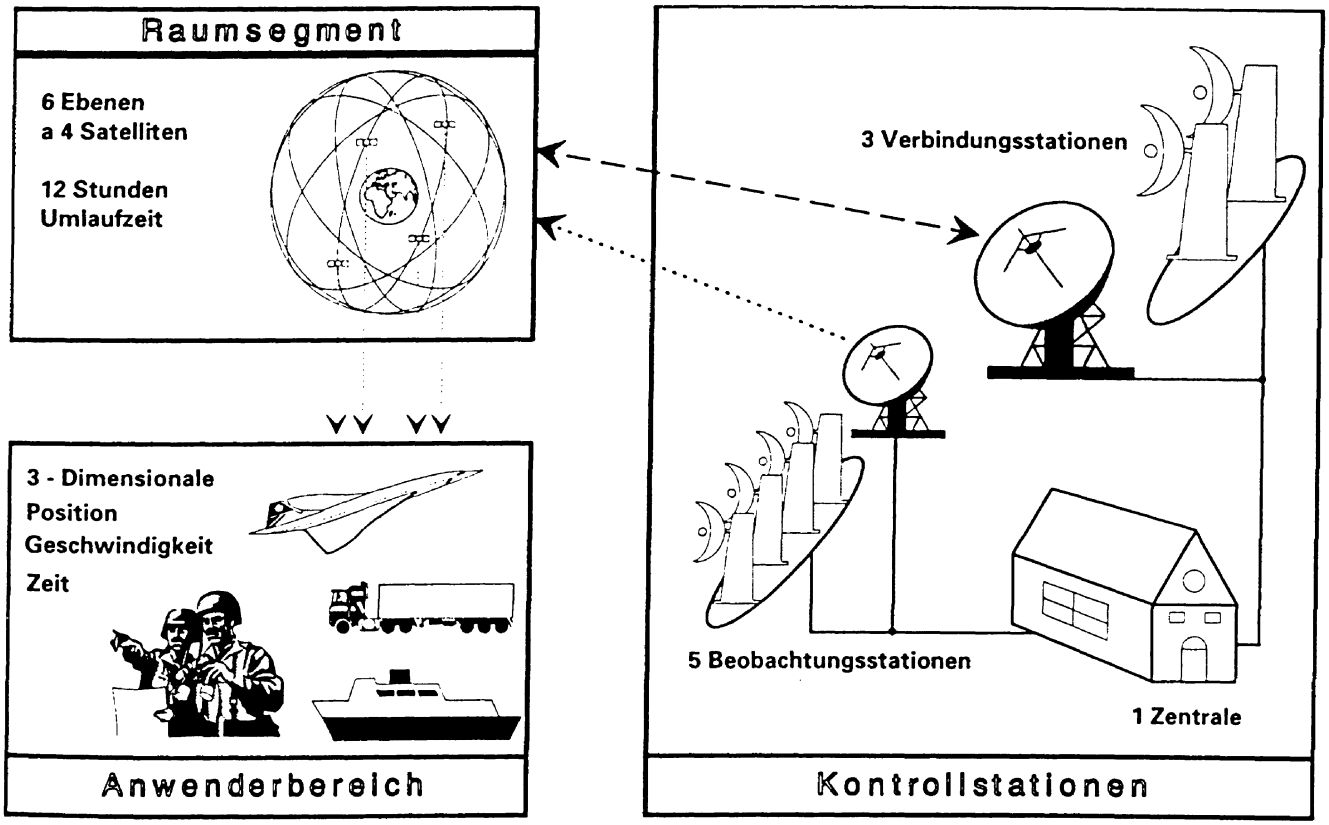


Bild 3.3-3: GPS-Systemüberblick

3.3.3.1 Space Segment (Raumsegment)

Space Segment

24 Satelliten (inkl. 3 Reserve) umkreisen in einer Höhe von 20'200 km auf 6 verschiedenen, zum Äquator um 55° geneigten Umlaufbahnen in genau 11 Stunden und 58 Minuten die Erde. Jeder Satellit führt bis zu 4 Atomuhren mit sich, welche regelmässig von den Kontrollstationen gestellt bzw,. synchronisiert werden. Aus der Frequenz der Atomuhren wird eine Grundfrequenz von 10,23 [MHz]abgeleitet. Aus diesem Wert ergeben sich durch Multiplikation bzw. Division alle anderen benötigten Frequenzen:

- Die Trägerfrequenzen L1 (1575.42 MHz) und L2 (1227.6 MHz)
- die Code-Frequenzen P (Precision) und C/A (Coarse Acquisition)

P-Code 10.23 MHz

- Die Frequenz des Datensignals, welches die Satelliten-Zeit, Uhren-Korrektur- und Ephemeridendaten, Almanachs und allgemeine Meldungen enthält, aus welchen der Anwender seine Position und die Zeitabweichung bestimmt

P-Code (PPS)
Genauigkeit ca. 5-30m

Der L1-Träger enthält alle Informationen, die nötig sind, um eine Position zu bestimmen. Auf dem L2-Träger sind der P- bzw. Y-Code zu finden, welcher nur für militärische Nutzer gedacht ist (**PPS**, **P**recise **P**ositioning **S**ervice) und bei Bedarf allgemeine Systemdaten.

C/A-Code mit SA (SPS)
Genauigkeit ca. 100m

Die Frequenz des C/A-Codes für zivile Nutzer beträgt mit 1.023 MHz nur ein Zehntel der Frequenz des P-Codes. Zusätzlich wird der C/A-Code noch mit dem S/A-Verfahren (Selective Availability) künstlich verschlechtert. Für die Auswertung dieses Codes spricht man auch von **SPS** (**S**tandard **P**ositioning **S**ervice).

3.3.3.2 Control Segment (Kontrollstationen)

Control Segment

Sämtliche Satelliten werden von einer Hauptstation in Colorado Springs und fünf über die gesamte Erde verteilten Kontrollstationen (Colorado Springs, Hawaii, Ascension, Diego Garcia und Kwajalein) kontrolliert. Diese empfangen die Signale aller sichtbaren Satelliten und berechnen daraus die jeweilige Entfernung zu einem Satelliten. Dieses Messergebnis und gegebenfalls andere wichtige Daten werden der Hauptstation zusammen mit meteorologischen Daten übermittelt. Das Kontrollzentrum berechnet daraus für jeden Satelliten die erforderlichen Parameter, die seine Umlaufbahn und sein Uhrenverhalten beschreiben. Diese Korrekturdaten werden über die Verbindungsstationen in die "Navigation Message" des Satelliten eingebunden. Die Orte der Kontrollstationen sind so gewählt, dass jeder Satellit diese mindestens einmal pro Tag so überquert, dass er vier Stationen gleichzeitig "in Sicht" hat.

3.3.3.3 User Segment (Anwenderbereich)

User Segment

Der Anwenderbereich beschreibt die Empfänger, die notwendig sind, um aus den Satellitensignalen die gewünschten Standort-, Navigations- oder Zeitdaten zu berechnen.

Empfängerkonzepte

Unterschiedliche Anwendergruppen und Aufgaben erfordern dazu unterschiedliche Empfängerkonzepte, welche grob in drei Gruppen eingeteilt werden können:

- Empfänger für Zivile Nutzung (eingeschränkte Genauigkeit)
- Empfänger für militärische Nutzung (hohe Genauigkeit)
- Geodäsie-Empfänger für die Auswertung der Trägerphase (sehr genau)

3.3.3.4 Systemfunktion und Positionsbestimmung

Systemfunktion

"Ein Empfänger für den zivilen Bereich ist auf die Träger-Frequenz L1 abgestimmt. Auf dieser Frequenz kommen die Informationen aller Satelliten an, wobei sämtliche Informationen einschliesslich der Codes mit der Systemzeit synchronisiert sind. Der Inhalt dieser Signale muss nun decodiert werden, um daraus alle notwendigen Daten für die Ortung und Navigation berechnen zu können.
Damit der Empfänger einen Satelliten identifizieren kann, muss er dessen Namen kennen. Diese Namen sind in der Form von sog. "Gold-Codes" (scheinbare Zufallsfolgen [**PRN**, **P**seudo **R**andom **N**oise] mit speziellen, nachrichtentechnisch positiven Eigenschaften) im Empfänger gespeichert. Zum Auffinden eines Satelliten wird nun eine Kopie des Codes des gesuchten Satelliten erzeugt.
Wenn ein Empfänger nicht auf einen gespeicherten Almanach zurückgreifen kann, wird der C/A-Code eines beliebigen Satelliten erzeugt, in der Hoffnung, dass der zugehörige Satellit im Sichtfeld des Empfängers liegt. Da die Phasenlage des ankommenden Signals nicht bekannt ist, muss dieser Code solange verschoben werden, bis ggf. Übereinstimmung festgestellt wird.
Die dazu benötigte Phasenverschiebung ist gleichzeitig ein Mass für die Signallaufzeit von dem entspr. Satelliten zum Empfänger." (aus Buch [7])

Positionsbestimmung

Da die Chiplänge des Codes bekannt ist, zählt man einfach die Anzahl der Verschiebungen um jeweils einen Chip - daraus folgt unmittelbar eine Laufzeitdifferenz bezogen auf das im Empfänger erzeugte Vergleichssignal.
Da nun für eine dreidimensionale Ortsbestimmung mindestens 4 Satelliten notwendig sind, erhält man durch Multiplikation der Signallaufzeit mit der Lichtgeschwindigkeit (s = c x t) vier Pseudoentfernungen (Pseude Ranges), welche alle um den gleichen Laufzeitfehler (= Zeitfehler mal Lichtgeschwindigkeit) von der tatsächlichen Position entfernt sind. Mit diesen Pseudoentfernungen kann nun die Auswertung der sog. Navigationsleichung beginnen.

Navigationsgleichung

4 unbekannte Grössen:
Position (X, Y, Z) und
die wahre Zeit (Systemzeit)

Diese besteht im Prinzip aus einem nichtlinearen Gleichungssystem mit 4 unabhängigen Gleichungen, welche aber im realen System nur durch Iteration und erheblichen Rechenaufwand lösbar ist, da die durch die Zeitmessung oder Störungen der Signale entstehenden Fehler als weitere Unbekannte in das Gleichungssystem miteinbezogen werden müssen.
Die Lösung des Gleichungssystems liefert die tatsächlichen Entfernungen und die Systemzeit. Aus den bekannten Koordinaten der Sender in bezug auf ein erdfestes Koordinatensystem (z.B. WGS-84) kann daraus wiederum die eigene Position in diesem Koordinatensystem abgeleitet und die Empfängeruhr auf die Systemzeit synchronisiert weren. Wenn nun eine der Unbekannten, z.B. die Höhe, bekannt ist, werden nur noch 3 sichtbare Satelliten zur Positionsbestimmung benötigt.

Geschwindigkeit

"Die Geschwindigkeit des Empfängers wird aufgrund der höheren Genauigkeit aus der Dopplerverschiebung der ankommenden Signale berechnet.

Dabei muss beachtet werden, dass sich auch ein auf der Erdoberfläche ruhender Empfänger durch die Erdrotation (in Abhängigkeit von der geographischen

Breite unterschiedlich schnell) gegenüber dem Satelliten bewegt. Beide Bewegungen verursachen Dopplerverschiebungen der Signale, die erst getrennt berechnet werden müssen. Die verbleibende Differenz zur gesamten Dopplerverschiebung der Signale, die im Empfänger gemessen wird, ist dann ein Mass für die Geschwindigkeit des Empfängers." (aus Buch [7])

Aus der bekannten Position und der Geschwindigkeit des Empfängers können dann alle relevanten Navigationsdaten abgeleitet werden.

3.3.3.5 Differential GPS (DGPS)

DGPS

Ist ein erweitertes Verfahren der Positionsbestimmung für Vermessungsaufgaben mit einer erhöhten geforderten Genauigkeit bis in den Millimeterbereich, welches auf einem einfachen Prinzip basiert:

Fehlerkorrektur

An einem bekannten und genau vermessenen Punkt befindet sich eine GPS-Referenz-Station, welche ihre eigene Position mittels vier Satelliten bestimmt. Durch Kenntnis der genauen Position, kann sie die Abweichung von der gemessenen Position bestimmen. Diese Abweichung (Fehlerkorrektur) wird dann entweder über Funk übertragen oder nach erfolgter Messung im nachhinein zur Korrektur verwendet und gilt für alle im Bereich von bis zu 200 km von der Referenz-Station entfernten GPS-Empfänger.

Fehlereinflüsse

Mit diesem Verfahren können die am häufigsten auftretenden Fehlereinflüsse wie

- Selective Availability (SA), (künstliche Verschlechterung)
- Ionosphären-Signalausbreitungsfehler
- Troposphären-Signalausbreitungsfehler
- Bahndatenfehler
- Uhrenfehler und
- geometrisch bedingte Fehler

weitgehend eliminiert werden.

3.3.4 HARDWARE Interface

HARDWARE Interface

Details zu der Schnittstelle des Zodiac GPS-Empfängers findet man im Anwendungs-Datenbuch. Eine Kopie dieses Dokumentes befindet sich im *Anhang* C dieses Abschnittes.

3.3.5 SOFTWARE Interface

SOFTWARE Interface

Details zu der Signalbeschreibung findet man im Rockwell-Dokument Zodiac Seriel Data Interface Specification oder im Abschnitt *3.4*.

3.3.6 BETRIEB

BETRIEB

Dieser Abschnitt beschreibt kurz und knapp die Betriebsfunktionen des Empfängers und gibt einen Überblick über die vom Empfänger unterstützten Navigationsfunktionen. Ebenso werden die Speichereinheiten, die Initialisierung sowie die verschiedenen Start-up-Modes vorgestellt.

3.3.6.1 Interne (onboard) Datenquellen

Interne Datenquellen

Static RAM (SRAM)

SRAM

Speichert sämtliche durch den Hersteller vorgegebene Variablen. Sobald Spannung anliegt, werden diese Daten zur Initialisierung bereitgestellt.
Satelliten-Ephemeriden, letzte Position und Frequenz-Standard-Parameter sind wichtige Daten um die Initialisierungszeit (TTFF) gering zu halten.

Real Time Clock (RTC)

RTC

Neben dem SRAM ebenfalls wichtig für die Initialisierung. Sobald Spannung anliegt, werden die Zeit-Daten zur Verfügung gestellt.

Electrically Erasable Programmable Read-Only Memory (EEPROM)

EEPROM

Speichert Daten über eine längere Zeit (Wochen). Nicht wie beim SRAM und der RTC, ist keine Spannung zur Erhaltung der Daten im "idle"-Status nötig. Wenn die Initialisierungsdaten vom SRAM nicht Verfügbar sind, stellt sie das EEPROM bereit (Satelliten-Almanach, letzte Position und Frequenz-Standard-Parameter).

Read-Only Memory (ROM)

ROM

Wird nur als Datenquelle benützt, wenn die Daten im SRAM oder EEPROM nicht gültig sind. Almanach-Daten sowie Frequenz-Standard-Parameter können mit begrenzter Genauigkeit benützt werden.

3.3.7 Initialisation

3.3.7.1 Definition

Initialisation

Definition

Die Initialisation ist definiert als der Datensatz und/oder Vorgänge, die unstetige Informationen beim Empfänger-Start bereitstellen und/oder ausführen. Die am häufigsten benützte Methode ist die Positions-, Geschwindigkeits- und Zeitinitialisation (PVT). Initialisierungsdaten sind erforderlich wenn die onboard-Daten ungültig sind. In diesem Fall werden Eingangs-Nachrichten durch den Benutzer vorbereitet und zum Empfänger gesendet. In der Regel kann sich der Empfänger auch selbst "booten", ohne gültige Daten, nur verlängert sich dadurch die Initialisierungszeit (TTFF).

3.3.7.2 Positions-, Geschwindigkeits- und Zeitdaten (PVT)

PVT (Position, Velocity, Time)

Wobei die Geschwindigkeit natürlich nur in dynamischen Systemen vorkommt. Fehlerfreie PVT- sowie gültige Almanach- und Ephemeridendaten sind erforderlich um die Liste aller momentan sichtbaren Satelliten zu erstellen und die TTFF möglichst gering zu halten.

3.3.7.3 Satelliten-Ephemeriden

Ephemeriden

Nicht wie PVT-Informationen, sind die Ephemeriden-Daten von jedem sichtbaren Satelliten erhältlich (min. 18s Erfassungszeit). Diese werden im SRAM gespeichert. Diese "over-the-air"-Verfügbarkeit hat den Vorteil, dass der Benutzer die Daten im Normalfall nicht bereitstellen muss.

3.3.7.4 Satelliten-Almanach

Almanach

Diese Daten sind ebenfalls von jedem sichtbaren Satelliten erhältlich (12.5 Minuten Erfassungszeit für den gesamten Datensatz) und werden im SRAM sowie im EEPROM abgelegt. Analog zu den Ephemeriden-Daten gilt dasselbe für die "over-the-air"-Verfügbarkeit.

3.3.7.5 UTC- und Ionosphären-Parameter

UTC und Ionosphäre

Für diese Informationen gilt dasselbe wie im obigen Abschnitt beschrieben.

3.3.8 Konfiguration

3.3.8.1 Definition

Konfiguration

Definition

Die Konfiguration ist definiert als der Datensatz und/oder Vorgänge welche Informationen zum Betrieb in einer bestimmten Umgebung und abgestimmt auf die Vorlieben des Anwenders bereitstellen und/oder ausführen. Häufige Beispiele sind Karten-Datum oder Satelliten-Erhebungswinkel. Die Konfiguration optimiert den Empfänger für den Betrieb in einer speziellen Situation. In der Regel bleiben die Parameter konstant bis sie erneut durch den Anwender verändert werden und werden im EEPROM gespeichert. *Tabelle 3.3-1* zeigt die Standard-Parameter, welche im folgenden kurz erläutert werden:

Parameter	Wert / Beschreibung
Datum	WGS-84
Mask angle	10 degrees (10 Winkel-Grad [°])
Cold start Control	Enabled (aktiviert)
Timeout	300 seconds
Platform lass	Automotive (Automobil)
Altitude measurement validity	Mark altitude solutions valid (bezeichnet Höhen-Lösungen als gültig)
Maximum Expected Horizontal Position Error (EHPE)	10 meters (*)
Maximum Expected Vertical Position Error (EVPE)	20 meters (*)
Criterion for the minimum number of satellites used for a solution	Zero (**)
DGPS validity	Not required for a valid solution (*) (für eine gültige Lösung nicht erforderlich)
Held altitude	Enabled (aktiviert)
Position Pinning	Enabled (aktiviert)
Ground Track Smoothing	Enabled (only when DGPS is not available) [nur wenn DGPS-Daten nicht verfügbar sind]
DGPS disable	"false" (i.e.,DGPS is active if corrections are available) [d.h., DGPS ist aktiv, wenn Korrekturen verfügbar sind]
Host port communication parameters (Rockwell binary)	9600-N-8-1
Host port communication parameters (NMEA)	4800-N-8-1
Auxiliary port communication parameters (RTCM input only)	9600-N-8-1

(*) = gültige Lösungskriterien
(**) = nach einem "vollständig ermittelten" oder erfolgreichen Übergang in die Navigation

Tabelle 3.3-1: Standard-Konfigurations-Daten

3.3.8.2 Geodetic Datums (Geodätisches Datum)

Geodätisches Datum

Es gibt zwei Konfigurationseinstellungen im Bezug auf des Datum: Datum-Auswahl und Datum-Definition. Die Datum-Auswahl bestimmt die Transformation in Bezug auf das für die Navigationsberechnungen verwendete Datumsformat. Über 189 versch. Datumsformate sind verfügbar. Die Datum-Definition erlaubt dem Anwender ein eigenes (max. 5) Datumsformat zu erstellen.

3.3.8.3 Satellite Selection (Satelliten-Auswahl)

Satelliten-Auswahl

Auch hier gibt es zwei Einstellungen: Den Erhebungswinkel und die spezifische Auswahl. Der Erhebungswinkel bestimmt den Einfallswinkel über dem Horizont (0°) und legt somit fest welche Satelliten dass aufgrund des DOP (Dilution of Precision) in die Navigationsgleichung miteinbezogen werden. Mit der spezifischen Auswahl kann ein sichtbarer Satellit explizit aus der Liste entfernt werden.

3.3.8.4 Differential GPS (DGPS) Control (DGPS-Kontrolle)

DGPS-Kontrolle

Hier sind folgende Einstellungen möglich: Wenn "DGPS disable" aktiviert ist, werden die Korrekturdaten bei der Positionsberechnung nicht berücksichtigt, auch wenn solche empfangen werden. Mit dem "DGPS timeout"-Parameter kann die maximale Zeit-Abweichung zwischen aktueller und DGPS-Korrektur-Zeit eingestellt werden. Wenn diese überschritten wird, werden die Korrekturdaten ebenfalls ignoriert.

3.3.8.5 Cold Start Control (Kaltstart-Kontrolle)

Kaltstart-Kontrolle

Bestimmt den Start-Modus (siehe entspr. Abschnitt weiter unten).

3.3.8.6 Solution Validity Criteria (gültige Lösungskriterien)

gültige Lösungskriterien

Diese Einstellung erlaubt dem Anwender die Vorgaben für eine "gültige" Navigationslösung selbst zu bestimmen. Diese wären:

- DGPS-Korrekturen
- Höhenlage
- Mindestanzahl der sichtbaren Satelliten
- Maximale horizontale und vertikale Positionsabweichung

3.3.8.7 Antenna Type Select (Antennen-Auswahl)

Antennen-Auswahl

Auswahl einer aktiven oder passiven Antenne. Kann je nach Erfassungsalgorithmus zur Optimierung beitragen.

3.3.8.8 User-Entered Altitude (Anwenderdef. Höhenlage)

Anwenderdef. Höhenlage

Wenn die eigene Höhenlage ganz genau bekannt ist, kann sie zur Verbesserung der Positionsgenauigkeit durch den Anwender eingegeben werden.

3.3.8.9 Vehicle Platform Select (Plattform-Klasse)

Plattform-Klasse

Legt fest, in welcher Umgebung der Empfänger installiert ist (siehe entspr. Abschnitt weiter unten).

3.3.8.10 Navigation Control (Navigations-Kontrolle)

Navigations-Kontrolle

Definiert weitere Einstellungen im Bezug auf die Navigationsberechnungen (siehe entspr. Abschnitt weiter unten).

3.3.8.11 NMEA Select (NMEA-Auswahl)

NMEA-Auswahl

Wenn aktiv, ist der Haupt-Port ausschliesslich für NMEA-Operationen konfiguriert (4800-8-N-1). Wenn inaktiv, werden die Haupt-Port-Operationen durch die vorhandenen Datenquellen (SRAM, EEPROM, etc.) konfiguriert. Mehr darüber im Abschnitt *3.4* .

3.3.8.12 ROM Defaults (ROM-Standard-Werte)

ROM-Standard-Werte

Wenn aktiv, werden sämtliche Konfigurations- und Initialisierungsdaten aus dem ROM verwendet. In diesem Fall ist der Haupt-Port für Rockwell binary-Operationen konfiguriert (9600-8-N-1). Wenn inaktiv (Normal-Zustand), erhält man die Konfigurations- und Initialisierungsdaten aus allen gültigen Datenquellen (SRAM, EEPROM, RTC, etc.).

3.3.9 Start-Up Modes (Start-Modus)

Start-Modus

Es existieren 4 Typen (Warm, Initialized, Cold und Frozen), basierend auf der Verfügbarkeit der benötigten Daten.

3.3.9.1 Warm Start

Warm Start

Dieser Zustand tritt ein, wenn alle Schlüsseldaten (PVT, Ephemeriden und Frequenz-Standard-Parameter) gültig und im SRAM vorhanden sind. Zwei übliche Methoden die zu diesem Zustand führen sind ein Software-Reset ("hot" start) oder die Rückkehr von einem "idle"-Power-Zyklus, nach welchem ein kontinuierlicher Empfang möglich war.

3.3.9.2 Initialized Start

Initialized Start

Dieser ist gleich dem Warmstart bis auf des Fehlen der Ephemeriden-Daten, dies bedeutet dass das SRAM noch nicht aktiv war oder die Daten nicht mehr gültig sind. Übliche Methoden die zu diesem Zustand führen sind PVT-Initialisierungen durch den Anwender oder kontinuierliche RTC-Operationen mit der letzten fehlerfreien Positionsangabe im EEPROM.

3.3.9.3 Cold Start

Cold Start

Dieser tritt auf wenn Position und/oder Zeit nicht bekannnt sind, meistens aufgrund einer unzuverlässigen Satellitenliste (sichtbare). Zur generierung dieser Listen werden Almanach-Daten benötigt.

3.3.9.4 Frozen Start

Frozen Start

Dieser tritt auf wenn überhaupt keine Datenquelle (SRAM, RTC, EEPROM) gültige Daten zur Verfügung stellt. So betrachtet gleicht es einem Recovery (Wiederherstellung) Mode, da mindestens das EEPROM immer gültige Daten enthalten sollte. Eine Einheit die über einige Monate nicht gebraucht wurde gelangt möglicherweise in diesen Zustand, weil die Daten im EEPROM abgelaufen oder nicht mehr komplett sind.

3.3.10 Satelliten-Management

3.3.10.1 Liste der sichtbaren Satelliten

Satelliten-Management

Liste der sichtbaren Satelliten

Ein Satellit wird als sichtbar erklärt wenn seine Höhe über dem Horizont bekannt ist, die Ephemeriden- und Almanach-Daten ihn als "gesund" erkennen und er nicht absichtlich aus der Liste der sichtbaren Satelliten entfernt wurde. Beachten Sie, dass die Messungen nur verwendet werden, wenn sich der Satellit über dem Erhebungswinkel befindet. Die Empfänger-Position und die aktuelle Zeit sind erforderlich um die Satellitenpositionen anhand der Umlaufbahndaten zu berechnen. Neue Satellitenlisten werden u.a. bei folgenden Ereignissen generiert: Ändern des Erhebungswinkels oder explizites an- oder abwählen eines Satelliten durch den Benutzer, runterladen eines neuen Satelliten-Almanachs oder Änderungen des Satelliten-Status. Bei der Initialisierung des Empfängers wird gleichzeitig zu Positions- und Zeit-Abgleichungen eine neue Liste der sichtbaren Satelliten erstellt (binary Message 1003).

Geometric Dilution of Precision GDOP (Verringerung der Genauigkeit)

GDOP

Ist ein Faktor der die geometrischen Verhältnisse der Satellitenkonstellation berücksichtigt. Er kann in der Ebene als das Problem verstanden werden, den Schnittpunkt von zwei sich flach schneidenden Geraden zu bestimmen. Er setzt sich aus dem PDOP (Faktor für die 3D-Position) und dem TDOP (Zeitfaktor: Empfängeruhrenabweichung mal Lichtgeschwindigkeit) zusammen. Der PDOP kann wiederum unterteilt werden und setzt sich aus dem HDOP (Positionsfaktor in der Ebene) und dem VDOP (Höhenfaktor) zusammen. Die geometrischen Verhältnisse der Satellitenkonstellationen ändern sich durch die Satellitenbewegung ständig und hängen deshalb von der Zeit und von dem Ort des Empfängers ab. Bei der günstigsten Satellitenkonstellation ist der Faktor GDOP am kleinsten. Bei mehr als vier sichtbaren Satelliten werden somit immer die vier mit dem kleinsten GDOP berücksichtigt.

3.3.10.2 Erfassungs-Modus

Erfassungs-Modus

Sequentielle Erfassung

Sequentiell

Ist die Erfassung eines Satelliten mit allen "non-tracking"- (nicht besetzten) Kanälen, z.B bei einem Kaltstart. Diese Erfassungsart wird immer beim ersten zu erfassenden Satelliten angewandt, danach wechselt der Empfänger automatisch auf die parallele Erfassung.

Parallele Erfassung

Parallel

Ist die Erfassung eines Satelliten mit einem "non-tracking"- (nicht besetzten) Kanal, z.B. nachdem der erste Satellit im Warmstart-Modus erfasst worden ist, bei welchem alle sichtbaren Satelliten einem Kanal zugewiesen und die Erfassung gleichzeitig erfolgte.

Gesamt-Suchprozess

Suchprozess

Bild 3.4 zeigt den Ablauf des Suchprozesses bei einer vorhandenen Liste der sichtbaren Satelliten. Ob sequentiell oder parallel gesucht wird ist abhängig von der Kanalverfügbarkeit und des erforderlichen Frequenz-Suchbereichs eines jeden Satelliten.

Zum besseren Verständis hier ein Auszug aus Buch [7]. "Wenn ein Empfänger nicht auf einen gespeicherten Almanach zurückgreifen kann, wird der C/A-Code eines beliebigen Satelliten erzeugt, in der Hoffnung, dass der zugehörige Satellit im Sichtfeld des Empfängers liegt. Da die Phasenlage des ankommenden Signals nicht bekannt ist, muss dieser solange verschoben werden, bis ggf. Übereinstimmung festgestellt wird. Das ankommende Signal ist aber aufgrund der Dopplerverschiebung verzerrt. Deshalb muss mit Schätzwerten eine zweidimensionale Suche nach Code und Frequenz durchgeführt werden. In diesem Fall ist es natürlich günstig, wenn ein Empfänger über mehrere Kanäle verfügt und gleichzeitig auf vier oder sogar acht Kanälen suchen kann."

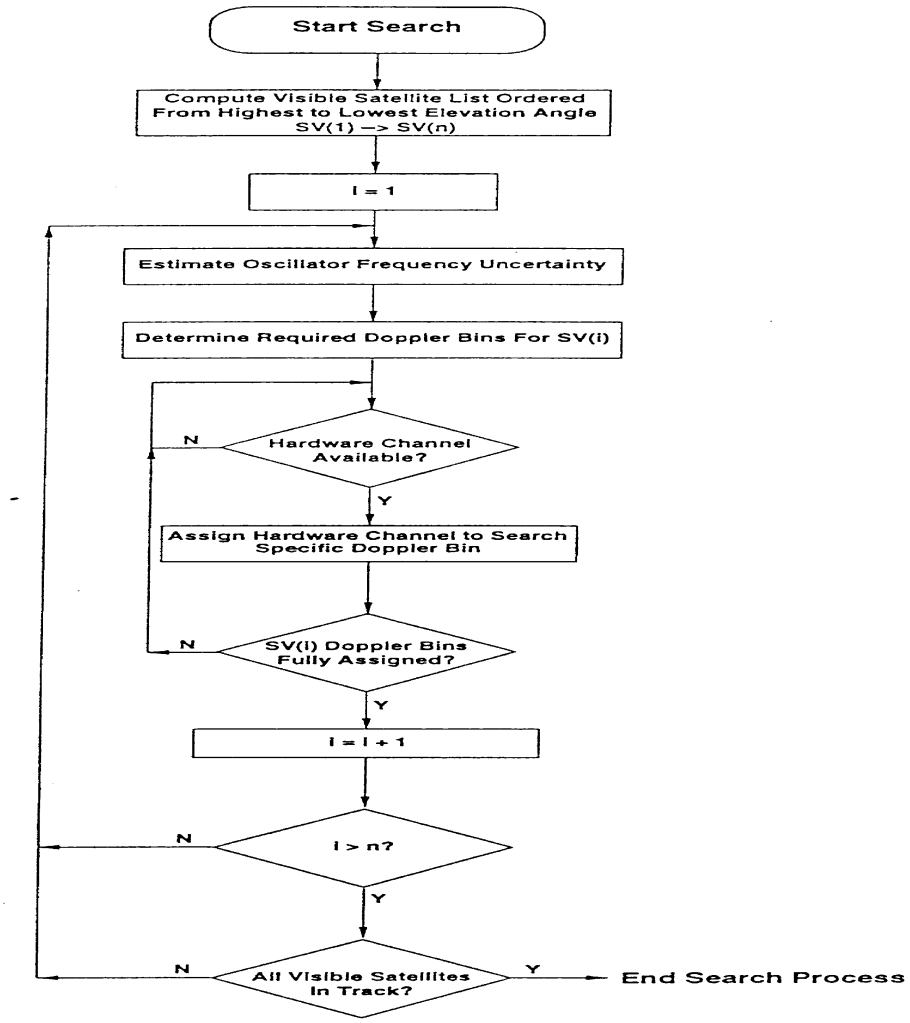


Bild 3.3-4: Zodiac-Such-Prozess

3.3.10.3 Daten-Sammlung

Daten-Sammlung

Ephemeriden

Ephemeriden

Bei fortlaufend Daten aussendenden, sichtbaren Satelliten dauert es zwischen 18 und 36 Sekunden um den gesamten Datensatz zu erfassen. Einmal Zusammengestellt, wird er benötigt um die richtige Satellitenposition, Geschwindigkeit und Beschleunigung (PVA, engl.) zu berechnen. Beachten Sie, dass dieser Datensatz aufgrund seiner zeitlich begrenzten Gültigkeit nicht im EEPROM gespeichert wird.

Almanach

Almanach

Bei fortlaufend Daten aussendenden, sichtbaren Satelliten dauert es mindestens 12.5 Minuten um den gesamten Datensatz zu erfassen. Die Almanach-Daten dienen primär der Bereitstellung des PVA-Status' , Bahn- und Uhrenparametern sowie der Satellitenkennung aller Satelliten. Beachten Sie, dass dieser Datensatz im EEPROM gespeichert wird, da die Almanach-Daten über den Zeitraum von einigen Wochen gültig sind.

UTC- und Ionosphärenkorrekturen

UTC- und Ionosphäre

Dieser Datensatz kann von einem beliebigen Satelliten erfasst werden, da er allgemeingültige Daten enthält. Bei fortlaufend Daten aussendenden, sichtbaren Satelliten dauert es mindestens 12.5 Minuten um den gesamten Datensatz zu aktualisieren. UTC-Korrekturdaten werden zur exakten Bestimmung der Zeitdifferenz zwischen der GPS- und der UTC-Zeit benötigt. Ionosphären-Daten dienen der Kompensation von Fehlern, die durch Störungen der Ionosphäre auf das Satellitensignal hervorgerufen werden. Beachten Sie, dass dieser Datensatz im EEPROM gespeichert wird, da er über den Zeitraum von einigen Wochen gültig ist.

3.3.11 Navigation

Navigation

Dieser Abschnitt beschreibt einige Funktionen der Navigations-Software des GPS-Empfängers.

Diese Software benützt Pseudo-Range, Trägerphase, Dopplermessungen der Satelliten und Höhenangabe (sofern vorhanden), um mit einem acht-fachen Kalmanfilter die Position, Geschwindigkeit und Uhrenfehler zu schätzen.

3.3.11.1 Geodätisches Datum

Geodätisches Datum: *[Beziehung zwischen einem lokalen (z.B. CH-1903) und einem globalen, geozentrischen System (z.B. WGS-84)], siehe [6]*

Geodätische Parameter werden in Eingabe- sowie Ausgabenachrichten verwendet. In der Ausgabenachricht 1000 (binary Message) sind die üblichen Koordinaten-Werte (Länge, Breite und Höhe) in Form von geodätischen Parametern ersichtlich. Ebenso sind in den NMEA-Nachrichten GGA und RCM sowie in der Eingabenachricht 1200 (binary Message) einige Parameter enthalten. Diese Werte sind immer auf das momentan aktive geodätische Datum bezogen, welches jeweils durch die folgenden Parameter definiert ist:

- Grosse Halbachse
- Abflachung des Referenz-Ellipsoid
- Verschiebungsparameter: Datumshiftkonstanten DX, DY und DZ

Der Anwender kann aus über 189 vordefinierten Datumsformaten auswählen. Die Auflistung aller verfügbaren Datumsformate ist im *Anhang* A zu Abschnitt *3.4* ersichtlich.

Auswahl des Geodätischen Datums

Auswahl

Das Standard-Datumsformat ist WGS-84 (Nr.0). Ein anderes Datum wird wie folgt eingestellt:

- Binary Message "Map Datum Select" (1211) mit dem gewünschten Datum-Code zum Empfänger senden (das angewählte Datum ist dann in der binary Output-Message 1000 und/oder 1001 ersichtlich)
- Binary Message "Geodetic Position and Velocity Initialisation" (1200) mit korrekter Angabe von Länge, Breite und Höhe zum Empfänger senden
- Um die Standard-Einstellung wieder herzustellen, erneut die Nachricht 1211 mit dem Datum-Code "0" senden
- Das jeweils aktive Datum ist im EEPROM gespeichert

HINWEIS: Das Datumsformat kann nur im binary-Modus geändert werden. Bei Betrieb im NMEA-Modus wird jeweils das vorgängig im binary-Modus eingestellte Datumsformat übernommen.

Anwenderdefinierte Datumsformate

Dem Anwender stehen die Datum-Codes von 300-304 zur Verfügung, um fünf eigene Datumsformate im EEPROM abzuspeichern. Bei nachträglicher Änderung wird das ältere jeweils überschrieben.

Anwenderdefiniert

3.3.11.2 Platform Class (Plattform-Klasse)

Damit ist die Umgebung im Bezug auf die Fortbewegungsgeschwindigkeit gemeint, in welcher der Empfänger installiert ist. Es kann aus sechs verschiedenen Klassen gewählt werden:

Plattform-Klasse

- Stationary (Stationär)
  Z.B. auf dem Dach eines Hochhauses

  Stationary (Stationär)

- Pedestrian (Fussgänger, langsam)
  Z.B. auf einer Wanderung oder Exkursion

  Pedestrian (Fussgänger, langsam)

- Marine
  Z.B. auf einem Schiff oder Boot, wo die vertikale Geschwindigkeit gleich Null ist

  Marine

- Marine-MSL (Mean Sea Level)
  Dasselbe wie Marine, ausser dass die geodätische Höhe in die Messungen miteinbezogen wird, um den erwarteten vertikalen Fehler (EVE) auszugleichen

  Marine-MSL (Mean Sea Level)

- Automotive (Aotomobile, mittel)
  Sämtliche Fahrzeuge, bei deren Fortbewegung die Höhe nicht unbedingt konstant bleibt (auch Motorräder und Bikes)

  Automotive (Aotomobile, mittel)

- Aircraft (Luftkraft, schnell)
  Für Flugzeuge und Helikopter, bei welchen auch die Höhenlage schnell ändert

  Aircraft (Luftkraft, schnell)

3.3.11.3 Navigation Cycle (Navigations-Zyklus)

Die Navigations-Software aktualisiert die ankommenden Daten einmal in der Sekunde.

Navigations-Zyklus

Zustands-Ausbreitung

Für Marine und Marine-MSL in 2 Dimensionen, für die restlichen 3 Klassen (ausgenommen Stationär, siehe oben) in 3 Dimensionen.

Zustands-Ausbreitung

Im stationären Fall, wenn die vertikale Geschwindigkeit Null ist, entspricht die Zustands-Ausbreitung der konstanten Höhe im aktiven Datumsformat.

Messungs-Prozess

Messungs-Prozess

Sobald vier Satelliten mit Ephemeridendaten und günstiger Konstellation über dem Erhebungswinkel sichtbar sind, beginnt der Kalman-Filter mit den Messungen. Dieser führt für jeden Satelliten zwei Messungen durch; die Trägerphase sowie die Dopplerverschiebung des Signals werden ausgeswertet.

Höhenlage-Prozess

Höhenlage-Prozess

Der Empfänger benützt die Angabe der Höhenlage unterstützend, wenn der erwartete vertikale Positionsfehler (EVPE) einen bestimmten Wert überschreitet und eine der folgenden Quellen verfügbar ist (Reihenfolge = Priorität):

- Eingabe durch den Anwender
- haltende Höhenlage
- Meereshöhe (nur Marine-MSL-Klasse)
- Rom-Höhenlage (nur bei Erfassung)

User-Entered Altitude (Eingabe durch den Anwender)

Eingabe durch den Anwender

Mit der Eingabe-Nachricht 1219 (binary Message) kann die Höhe mit sofortiger Wirkung, gültig bis sie Vom RAM oder vom EEPROM gelöscht wird, eingegeben werden. Diese Form wird vor allem bei gleichbleibender Höhe angewandt.

Held Altidude (haltende Höhenlage)

haltende Höhenlage

Diese ist bei gültiger Navigationslösung im Empfänger gespeichert. Die Benützung dieses Parameters ist in der Regel ein wichtiger Leistungsfaktor in einer Stadtumgebung mit Verkehrsverstopfung. In der Standard-Einstellung ist dieser Parameter aktiviert, kann aber mit der Konfigurations-Nachricht 1221 (binary Message) deaktiviert werden.

Position Pinning

Position Pinning

Ohne DGPS-Korrekturen unterliegen die Satelliten-Messungen Zeitbereichsschwankungen, welche Geschwindigkeitsfehler von einem halben bis zu zwei Metern zur Folge haben, d.h. der Empfänger erhält rechnerisch gesehen eine Geschwindigkeit obwohl er stationär ist. Dieses unfreiwillige "wandern" des Empfängers ist für viele Anwendungen (Routenplanung mit der Karte) nicht zu gebrauchen. Dieser Effekt kann nun durch die Software mittels "pinning" (festnageln, befestigen) annähernd ausgeglichen werden. Dieser Parameter ist im Normallfall auch aktiv, kann aber mit der Konfigurations-Nachricht 1221 (binary Message) deaktiviert werden. Die aktive Einstellung wird jeweils im EEPROM gespeichert und bleibt somit auch nach dem Ausschalten erhalten.

Ground Track Smoothing

Ground Track Smoothing

Ohne DGPS-Korrekturen werden die Satelliten-Messungen durch die Selective Availability (SA) künstlich verschlechtert, was zusätzliche Fehler im Bereich von ca. 30 Metern bedeutet. Wenn die sichtbaren Satelliten aufgrund der Erhebung und der Umlaufbahn wechseln, verschiebt sich die bis anhin geschätzte Position um 10-20, bei schlechter geometrischer Konstellation jedoch bis zu 100 Metern. Um diesen "Konstellations-Schalter-Effekt" augenblicklich zu kompensieren, hält der Empfänger einen geschätzten Wert einer geglätteten Plattform-Grundlinie sowie der Höhe fest. Diese Glättung ist möglich ohne Verlust einer dynamischen Gegenwirkung. Wie auch immer, diese Methode ist nicht erforderlich und bei aktiven DGPS-Korrekturdaten ohne Wirkung. Ohne DGPS, ist diese Funktion standardmässig aktiviert, kann aber mit der Konfigurations-Nachricht 1221 (binary Message) deaktiviert werden. Die aktive Einstellung wird jeweils im EEPROM gespeichert und bleibt somit auch nach dem Ausschalten erhalten.

3.3.11.4 Solution Validity (Gültigkeit der Lösung)

Gültigkeit der Lösung

Ob die Resultate einer Navigationsmessung gültig sind, kann durch den Anwender bestimmt werden, da je nach Anwendung gewisse Kriterien wichtiger sind als andere. Somit kann die Auswertung einer Navigationsmessung optimal auf die Anwendung abgestimmt werden. Die folgenden fünf Kriterien müssen dabei berücksichtigt werden:

Ist eine Höhenmessung nötig ?

- Dieser Parameter erlaubt dem Anwender die Navigationsresultate als ungültig zu erklären, sobald eine Höhenmessung erfolgt (Standard ist, dass Resultate mit Höhenmessungen gültig sind)

Sind DGPS-Korrekturdaten nötig ?

- Wenn ja, sind sämtliche Navigationslösungen ohne DGPS ungültig (Standard ist, dass keine DGPS-Korrekturdaten nötig sind)

Wie viele sichtbare Satelliten sind nötig ?

- Wenn z.B. 4 eingestellt wird, sind die Navigationsresultate mit nur 3 sichtbaren Satelliten ungültig (Standard-Wert ist hier Null)

Wie gross darf der maximale horizontale Positionsfehler (EHPE) sein ?

- Bei Überschreitung des eingegebenen Wertes sind die Navigationsresultate ungültig (Standard-Wert ist hier 10 Meter)

Wie gross darf der maximale vertikale Positionsfehler (EVPE) sein ?

- Bei Überschreitung des eingegebenen Wertes sind die Navigationsresultate ungültig (Standard-Wert ist hier 20 Meter)

Diese Einstellungen können nur in der "Solution validity Criteria"-Nachricht 1217 (binary Message) vorgenommen werden und sind bis zur nächsten Änderung im EEPROM gespeichert.

3.3.11.5 Mean Sea Level (Meereshöhe)

Meereshöhe

MSL ist ein sogenanntes Geoid. Ein Geoid ist ein von der tatsächlichen Erdgestalt abweichender thoeretischer Körper, dessen Oberfläche die Feldlinien der Schwerkraft überall im rechten Winkel schneidet. Die Höhe des Geoids über (+) oder unter (-) dem Referenz-Ellipsoid (WGS-84) wird "geoidal Separation" (Separation = Trennung) genannt. Die auf das Geoid bezogende Höhe (Altitude MSL) kann aus der Differenz zwischen gedätischer Höhe und der "geoidal Separation" berechnet werden.

3.3.11.6 Magnetic Variation (Magnetische Abweichung)

Magnetische Abweichung

Das vom Empfänger verwendete magnetische Abweichungsmodell ist das International Geomagnetic Reference Field (IGRF) 95. Informationen hierzu gibt es unter http://julius.ngdc.noaa.gov/seg/potfld/geomag.html .

3.3.12 Unterstützende Funktionen

3.3.12.1 Serielle Kommunikations-Schnittstellen

Unterstützende Funktionen

Serielle Schnittstellen

Der Empfänger besitzt zwei bidirektionale, serielle RS-232 Schnittstellen, die bereits in Abschnitt *3.2* und abermals in Abschnitt *3.4* erwähnt werden. Deshalb an dieser Stelle nur das wichtigste:

Host Port (Haupt-Port)

Haupt-Port

Ist standardmässig für das Rockwell binary Format konfiguriert (9600-N-8-1). Kann aber ebenfalls für das NMEA-Protokoll verwendet werden (4800-N-8-1). Die Ausgabe-Nachrichten können mit Hilfe der Erfassungs-Kontroll-Nachrichten in beiden Fällen konfiguriert werden. Die Port-Einstellungen werden im nichtflüchtigen EEPROM gespeichert.

Auxiliary Port (Hilfs-Port)

Hilfs-Port

Ist ausschliesslich für den Empfang von RTCM SC-104 Korrekturdaten vorgesehen (9600-N-8-1).

3.3.12.2 EEPROM Services

EEPROM-Dienste

Das EEPROM hat einen nichtflüchtigen Speicher und enthält System-Konfigurationsparameter, welche in der Regel über eine Zeit von einigen Wochen gültig sind. Die Konfigurations- und Betriebsdaten im EEPROM werden nur bei der System-Initialisierung gelesen, wenn der Inhalt des SRAMS ungültig ist. Dies wird mit einer Checksumme überprüft. Wenn auch die EEPROM-Daten ungültig sind, werden die Standard-Werte aus dem ROM zur Initialisierung verwendet. EEPROM-Datenblocks werden bei einer bedeutenden Änderung der Systemdaten aktualisiert (updated). Bei durch den Anwender beeinflussbaren Parametern genügt eine Änderung der Einstellungen um das EEPROM zu refreshen. Im Fall der langsam ändernden Daten (Position, Almanach, etc.) sind zusätzliche, fast erzwungene Eingaben nötig um die bestehenden Daten zu aktualisieren.

Die Variablen Parameter und im EEPROM erfasste Daten sind in der untenstehenden Tabelle aufgelistet:

Konfigurations-Daten	• Serial Port Configuration (beide Ports) (Port-Konfiguration)
	• Solution Validity Criteria (gültige Lösungskriterien)
	• Selected Datum (gewähltes Datum)
	• Platform Class (Plattform-Klasse)
	• Cold Start Control (Kalt-Start-Kontrolle)
	• Satellite Elevation Mask Angle (Satelliten-Erhebungswinkel)
	• Satellite Candidate List (Liste der betr. Satelliten)
	• Antenna Selection (Antennen-Auswahl)
	• Differential GPS Control (Differential-GPS-Kontrolle)
	• Default Serial Output Messages (Standard Ausgabe-Nachrichten)
	• User Defined Datums (Anwender-definiertes Datum)
	• Navigation Control (Navigation-Kontrolle)
Satellite Management Parameters	• UTC and Ionosphere Model Parameters (UTC- und Ionosphären-Parameter)
	• Frequency Standard Data (Standard Frequenz-Daten)
	• Almanac Data (Almanach-Daten)
Navigation Data	• Last Known Position (letzte bekannte Position)
	• User-Entered Altitude (durch den Anwender eingegebene Höhe)

Tabelle 3.3-2: Im EEPROM erfasste Parameter und Daten

3.3.12.3 RTC Services

RTC-Dienste

Die RTC-Einheit speichert die Zeit / das Datum und hält diese Daten aufrecht, wenn der Empfänger in einem "Idle"-Status ist (Speisung ext. oder durch den onboard "SuperCap"). Solange Betriebsspannung da ist, liefert die RTC-Einheit immer die richtigen und aktuellen Zeit/Datums-Daten. Die Daten werden nur bei der System-Initialisierung ausgelesen. Ein Abbild der Daten befindet sich ebenfalls im seperaten RTC-RAM, welches mit Hilfe einer Checksumme bestimmt, ob die geladenen Daten noch gültig sind. Ist dies einmal nicht der Fall, wird die zuletzt gespeicherte Zeit aus dem SRAM geladen. Wenn diese nicht vorhanden ist, wird die Zeit für ungültig erklärt und der Emfpäger muss neu initialisiert werden. Die Aktualisierung der Zeit/Datum erfolgt periodisch, wenn das System im Kalman-Filter-Modus ist.

3.3.12.4 Differential GPS (DGPS)

DGPS

DGPS-Funktionen werden zur Eliminierung der SA-Verschlechterung sowie anderen Fehlerquellen benötigt. Dazu müssen Satelliten-Korrekturdaten einer genau vermessenen Kontrollstation empfangen und in die Navigationsberechnungen miteinbezogen werden. Dabei werden vor allem die Zeit-, Ephemeriden- und durch die Atmosphäre bedingten Fehler nahezu kompensiert. Der Zodiac-Empfänger akzeptiert RTCM SC-104 Korrekturdaten, welche direkt über den Hilfsport eingelesen werden können. Auch über den Haupt-Port können Korrekturdaten empfangen werden, nur müssen sie entsprechend formattiert sein (binary Message 1351). Mehr dazu im Abschnitt *3.4* .

Das RTCM-Protokoll

RTCM-Protokoll

Der Empfäger akzeptiert 6 von 8 RTCM-Datensätzen, welche ohne jegliche Aufbereitung direkt eingelesen werden können. Das einzige was der Anwender zu tun hat, ist die Daten mit dem auf der Diskette vorhandenen Programm "RTCMCHK" auf Vollständigkeit zu prüfen. Dabei werden die 30-Bit Wörter bereits im Vorfeld verworfen, bei welchen die 6-Bit Parität nicht zutrifft, ebenfalls Wörter mit einzel-Bit Fehlern. Der Empfänger prüft die ankommenden Daten ebenfalls, decodiert und berechnet die benötigten Korrekturdaten. Dee nicht benötigten Nachrichten werden ebenfalls sogleich verworfen.

3.3.12.4.1 RTCM-Nachrichten-Typen

RTCM-Nachrichten-Typen

Der Empfänger als eine Untermenge der 64 RTCM SC-104-Nachrichten, welche in der Tabelle 4.2 der RTCM SC-104-Dokumentation, Version 2.1 Standard, zu finden sind. Obschon der Empfänger sämtliche Nachrichten akzeptiert und decodiert, werden nicht alle benötigt. Mehr Informationen dazu im "Jupiter" Global Positioning System (GPS) Receiver Datenbuch.

Nachrichtentyp 1

Typ 1

Beinhaltet Pseudorange und Pseudorange-Bereichs-Korrekturen für einen kompletten Satz von sichtbaren Satelliten und ist daher der meist verwendete Typ.

Nachrichtentyp 2

Typ 2

Beinhaltet Delta-Korrekturen und wird vor allem bei einem Ephemeriden-"timeout" benötigt. Dieser Typ wird vom Feldempfäger zusammen mit Typ 1 oder Typ 9 verwendet, wenn die Referenzstation ebenfalls den gleichen Ephemeriden-Satz verwendet.

Nachrichtentyp 9

Typ 3

Hat dasselbe Format wie Typ 1, enthält jedoch nur Korrekturdaten für die gerade sichtbaren Satelliten und wird mit einer höheren Geschwindigkeit übertragen.

Einverständnis mit RTCM SC-104-Forderungen

RTCM SC-104-Forderungen

Die Radio Technical Commission for Maritime Services (RTCM) besitzt ein spezielles Komitee mit der Bezeichnung 104 (SC-104), welches das Ziel hat, Standards für die Übertragung von DGPS-Korrekturdaten zu verfassen. Bis auf die Hardware sind die Empfänger der Zodiac-Familie kompatibel mit der aktuellen Version (2.1) dieses Standards. Der Empfänger ist jedoch nicht als “terminal equipment” konfiguriert, wie das unter Paragraph 5.3.4 im RTCM-Standard spezifiziert ist.

3.3.12.4.2 DGPS-Initialisation und Konfiguration

DGPS-Initialisation und Konfiguration

Sobald der Empfänger eingeschaltet wird, initialisiert dieser seine interne DGPS-Datenbank um anzuzeigen, dass keine gültigen DGPS-Daten vorliegen. Wenn die DGPS-Status-Nachricht 1005 (binary Message) angefordert wird, zeigt diese ebenfalls, dass keine Korrekturen ausgeführt wurden. Auch einige der Positions-Status-Nachrichten (binary Messages 1000 und 1001 sowie die NMEA-Message GGA) zeigen an, dass der Empfäger keine DGPS-Lösungen berechnet. Sobald jedoch genügend gültige RTCM-Daten vorliegen, werden sie auch verwendet, natürlich nur wenn die entsprechende Einrichtung vorhanden ist (RTCM-Datenquelle mit Hilfsport verbunden, Empfänger entsprechend konfiguriert). Die folgende Bedingungen müssen zudem ebenfalls alle erfüllt sein:

- Ephemeriden-Daten sind von mindestens 4 Satelliten erfasst
- DGPS-Korrekturdaten dieser 4 Satelliten sind ebenfalls erfasst und dürfen nicht älter als in der DGPS-Kontroll-Nachricht 1214 (binary Message) angegeben sein

- Die Ausgabe der Ephemeridendaten "Issue Of Data Ephemeris" (IODE) ist für den Empfäger dieselbe wie für die RTCM-Korrekturdaten
- Sämtliche der sichtbaren Satelliten sind "gesund"
- Der Anwender-Differential-Bereichs-Fehler "User Differential Range Error" (UDRE) der RTCM SC 104-Datenquelle ist gleich oder kleiner als 8 Meter für alle Satelliten
- Die RTCM SC 104-Datenquelle ist in gutem Zustand
- Der Anwender hat die DGPS-Operationen nicht explizit ausgeschaltet
- DGPS-Operationen deaktiviert

Dies ist mit der bereits erwähnten DGPS-Kontroll-Nachricht 1214 (binary Message) möglich. In diesem Fall wird der Empfäger weder DGPS-Informationen berechnen noch löschen. Vielmehr werden die ankommenden Daten fortlaufend aber stillschweigend berechnet und in der internen Datenbank abgespeichert, so dass beim nächsten Gebrauch sofort die aktuellen Daten zur Verfügung stehen.

DGPS Reset

DGPS Reset

Mit derselben Nachricht kann der laufende DGPS-Prozess auch "resetet" werden. Dies macht jedoch die aktuell gespeicherten Daten ungültig oder ersetzt sie durch Standard-Werte. Ebenfalls werden die DGPS-Operationen unterbrochen bis neue, vollständige RTCM- und Ephemeridendaten vorliegen. Weiter zu beachten ist, dass die Einstellungen "DGPS Disable" und "Correction Timeout" unberührt bleiben und auch nach dem Reset ihren Wert behalten. Ansonsten werden sämtliche Parameter auf ihren Standard-Wert zurückgesetzt !

DGPS-Status-Anforderung

DGPS-Status-Anforderung

Die Differential-GPS-Status-Nachricht 1005 (binary Message) enthält Statusinformationen über jede ausgeführte Korrektur. Wenn also DGPS-Lösungen nicht möglich sind, kann der Anwender mit Hilfe der Status-Meldungen herausfinden, warum diese nicht berechnet wurden.

3.3.12.5 Built-In-Test (BIT)

Built-In-Test (BIT)

Im Rockwell binary-Modus kann der Empfäger zu jeder Zeit einen BIT ausführen (binary Message 1300) und die Resultate in einer entsprechenden Nachricht ausgeben. Die Ausführung unterbricht sofort sämtliche anderen Operationen und führt zu einem System-Reset. Ausgabe-Nachrichten die gerade zu diesem Zeitpunkt hardware-mässig ausgeführt werden, werden ausgegeben, alle anderen müssen bis zur Beendigung des BIT warten. Dies bedeutet, dass die BIT-Resultate nicht unbedingt als erste Nachricht nach dem Test ausgegeben werden, sondern diejenigen, welche sich bereits in der Warteschlange befinden.

3.3.12.5.1 Interpretation der BIT-Resultate

BIT-Resultate

Die fehlerhaften Wörter, welche in der Nachricht 1100 (binary Message) definiert sind, sind Null wenn die zugehörige Einheit in Ordnung und nicht Null, wenn ein Fehler vorliegt.

ROM-Fehler

ROM-Fehler

Das ROM-Fehler-Wort in der Nachricht 1100 zeigt das Resultat eines Checksummen-Tests. Ein fehlerhafter Status bedeutet, dass der ROM-Chip womöglich defekt ist.

RAM-Fehler

RAM-Fehler

Das RAM-Fehler -Wort in der Nachricht 1100 zeigt das Resultat eines Adress-Linien-Prüf-Tests. Ein fehlerhafter Status bedeutet, dass der/die RAM-Chip(s) womöglich defekt ist/sind.

EEPROM-Fehler

EEPROM-Fehler

Der Empfänger merkt sich die Parameter, welche ins EEPROM geschrieben werden. Wenn diese nun nicht mehr komplett herausgelesen werden können oder auf einen versuchten Zugriff nicht geantwortet wird, gilt dies als Fehler. In diesem Fall ist das EEPROM womöglich defekt.

DSP-Fehler

DSP-Fehler

Das DSP-Fehler-Wort in der Nachricht 1100 zeigt die Resultate der DSP-Tests. Ein fehlerhafter Status bedeutet, dass der DSP-Chip womöglich defekt ist.

RTC-Fehler

RTC-Fehler

Das RTC-Fehler-Wort in der Nachricht 1100 zeigt die Resultate eines "time rollover"-Tests. Ein fehlerhafter Status bedeutet, dass die RTC-Einheit womöglich defekt ist.

Haupt- und Hilfsport-Fehler und Byte-Zähler

Haupt- und Hilfsport-Fehler und Byte-Zähler

Wie für das EEPROM gibt es keine expliziten Test für die beiden Ports. Jedenfalls ist die Anzahl sämtlicher durch beide Ports empfangenen Bytes in der Nachricht 1100 (binary Message) erfasst. Dies gibt einen Überblick über die Zuverlässigkeit der Schnittstellen. Wird nun die Anzahl der fehlerhaften Bytes zu gross, ist entweder die Schnittstelle defekt oder nicht richtig konfiguriert.

3.4 Zodiac Serial Data Interface

ZODIAC Serial Data Interface
④

3.4.1 Zodiac Datentypen und Nachrichtenformate

Dieser Abschnitt beschreibt die Datenformate der beiden Nachrichten-Typen, welche der Zodiac-GPS-Empfänger an der seriellen Schnittstelle ausgibt.
Der Inhalt und die Struktur der binären Nachrichten (nachfolgend *"binary message"* genannt) kann im folgenden, und der Inhalt sowie die Struktur der NMEA-Nachrichten (*"NMEA message"*) im Abschnitt 3.4.2 nachgesehen werden.

3.4.1.1 Binäres Nachrichten-Format und Wort-Struktur

Binary Message Format

Binary Message Format

Dieses ist nahezu identisch mit dem bei den Vorgänger-Empfänger-Modellen (NacCore/Micro Tracker) verwendeten Protokoll. Einziger Unterschied ist, dass sämtliche floating-point-Werte nun durch fixed-point integer (Ganzzahlen) mit ex- oder implizitem Skalierungsfaktor represäntiert werden.
Jede binary message besteht aus einem Header- und einem Datenteil, wobei jeder Teil seine eigene Checksumme hat. Jede Nachricht besteht mindestens aus einem Header, doch nicht jede Nachricht hat auch einen Datenteil. Die Erkennung der Nachricht erfolgt durch den Header.

Tabelle 3.4-1 zeigt die verwendeten Datentypen:

Typ	Abkürzung	Wörter (16 Bit)	Bits	Max. Bereich
Bit	Bit	-	0 bis 15	0 bis 1
Character	C	-	8	ASCII 0 bis 255
Integer	I	1	16	-32'768 bis +32'768
Double Integer	DI	2	32	-2'147'483'648 bis +2'147'483'648
Triple Integer	TI	3	48	-140'737'488'355'328 bis +140'737'488'355'328
Unsigned Integer	UI	1	16	0 bis 65'535
Unsigned Double Integer	UDI	2	32	0 bis 4'294'967'295
Unsigned Triple Integer	UTI	3	48	0 bis 281'474'976'710'656

Tabelle 3.4-1: Binary Messages Datentypen

Wort-Struktur

Wort-Struktur

Ein Integer ist mit einer Länge von 16 Bits definiert. Hat der Ausdruck eine feste Anzahl Stellen, ist der Wert des “least significant bit” (LSB) des Integers gegeben. Um einen fixed-point-Wert in eine floating-point-Variable zu konvertieren, muss die Integer-Repräsentation (floated) mit der entspr. Auflösung multipliziert werden. Bei der float-Konvertion muss der Wert und die Auflösung des Ausdrucks besonders berücksichtigt werden, um sicherzustellen , dass der gewählte float-Typ eine ausreichende Mantissen-Länge besitzt, damit die Genauigkeit des Wertes erhalten bleibt. Ausdrücke mit 3 Wörtern erfordern skalierte Teile der Variable separat, bevor sie in die floating-point-Form umgerechnet werden.

Zusammengesetzte Wörter können unabhängige Definitionen für jedes Bit-Feld haben. Flag-Bits sind entweder Null (false) oder Eins (true). Sämtliche Bits welche innerhalb der Bit-Beschreibung als Reserve bezeichnet sind haben undefinierte Werte bei der Ausgabe und müssen deshalb Null gesetzt werden.

3.4.1.2 Binary Message Header

Header-Teil

Dieser ist seit dem NavCore V-Format modifiziert worden um sich an die Erfassungs-Anfrage-Nachrichten (message logging requests) anzupassen. Das neue Format zeigt *Bild 3.4-1*.

1000 001 1111 1111	Wort 1
Nachrichten ID	Wort 2
Datenwort-Zähler	Wort 3
DCLE QRAN 00XX XXXX	Wort 4
Header-Checksumme	Wort 5

Bild 3.4-1: Binary Message Header Format

Message Header Word 1

Header-Wort 1

Ist das synchron-Wort mit DEL (255d), besetzt die ersten 8 Bits und SOH (129d), besetzt die zweiten 8 Bits

Message Header Word 2

Header-Wort 2

Beinhaltet die numerische Nachrichten ID (Identitäts-Nummer)

Message Header Word 3

Header-Wort 3

Beinhaltet die Anzahl der Worte im Datenteil der Nachricht. Ist die Anzahl = 0, besteht die Nachricht nur aus dem Header

Message Header Word 4

Header-Wort 4

Das Flag-Wort ist zur umfassenden Kontrolle der ACK/NAK-Anfragen sowie zur Ausführung der Erfassungs-Anfrage-Nachrichten modifiziert worden. Das Prinzip ist vergleichbar mit den älteren Versionen, ausser das der Anwender jetzt die Möglichkeit hat, unabhängig von der Anfrage entweder nur ACK oder NAK, oder aber beide abzufragen. Durch setzen des Anfrage-Bits (request, R) und des Erkennungs-Bits (acknowledge, A) oder des negativen Erkennungs-Bits (negative acknowledge, N) oder aber beiden kann das richtige Erkennungs-Verhalten eingestellt werden.
Mit dieser Näherung kann der Anwender die Anfragen (requests) nur im NAK -Modus deklarieren, wenn beim Auftreten eines Problemes ohne Verlust des benötigten overheads fortlaufend NAK's ausgeführt werden.

Die ersten 6 Bits dieses Wortes können als zusätzlichen Eingabe-Identifier benützt werden. Dieser wird nicht explizit vom Empfänger ausgeführt, er kommt zurück, an derselben Stelle als Teil des Headers in ACK/NAK-Antworten. Diese Eigenschaft gewährt dem Anwender die eindeutige Unterscheidung, welche Eingabe-Nachricht eine Erkennung aufweist, wenn mannigfaltigen Eingabe-Nachrichten mit den gleichen nachrichten-ID's wärend der Abfrage-Zeit ausgeführt werden.

Die Bits Anschluss (connect, C) und kein Anschluss (disconnect, D) werden zum aktivieren und deaktivieren der Ausgabe-Nachrichten gebraucht und können entweder unabhängig oder in Verbindung mit den Erfassungs-Anfrage-bits genützt werden. Eine nur Header-Datei, mit einer Nachrichten-ID und dem Bit [C] gesetzt, aktiviert die spezifische Nachricht mit den existierenden Zeit-Charakteristiken. Genau gleich, aber mit gesetztem Bit [D], deaktiviert die spezifische Nachricht. Werden beide Bits gesetzt (C und D), so wird die Nachricht ignoriert. Beachten Sie, dass deaktivieren oder aktivieren einer Nachricht keinen Einfluss auf die Zeit-Charakteristiken (Trigger, Intervall oder Offset) hat. Bei gesetztem Bit [C] werden zuerst die Zeit-Charakteristiken aktualisiert und dann die Nachricht aktiviert. In ählicher Weise wird bei einer kombinierten Erfassungs- und deaktivierten Anfrage die Nachricht nach dem Setzen der Zeit-Charakteristiken deaktiviert. Um sämtliche Nachrichten abzuschalten, muss die Nachrichten-ID auf FFFFh gesetzt und das Bit [D] gesetzt werden.

Das Frage-(query) Bit [Q] bewirkt die Ausgabe der durch die Nachrichten-ID spezifizierte Nachrichticht irgendwann beim nächsten Ausgabezyklus. Standard Erfassungs-Anfrage-Nachrichten werden akzeptiert wenn das Erfassungs-(log) Bit L gesetzt und die erforderlichen Parameter im Datenteil der Nachricht vorhanden sind.

Das ausgedehntere Erfassungs- (log) Bit E kann für Nachrichten benützt werden, die weitere Definitionen der Erfassungs-Anfrage erfordern, entsprechend der Nachrichten-abhängigen Daten. Z. B. eine Ephemeriden-Nachrichten-Anfrage, die zur Ausgabe des Ephemeriden-Datensatzes ein spezifisches Satelliten-PRN (Pseudo Random Noise) Signal benötigt.

Message Header Word 5

Header-Wort 5

Ist die Datenchecksumme. Diese berechnet sich aus der Summe (modulo 216) aller sich im Header befindlichen Wörter (inkl. DEL und SOH) und anschliessender Komplement-Bildung der Summe.

If SUM = -32'768, Header Checksumme = SUM; else Header Checksumme = -SUM

Die Berechnung der Header-Checksumme lässt sich mathematisch wie folgt ausdrücken:

$$SUM = Mod2^{16} \sum_{i=1}^{4} Word(i)$$

Wobei gilt:

- Einwertige Negation berechnet sich aus den beiden Komplements eines 16-Bit Datenwortes
- Mod 216 betrifft die ersten 16 Bits eines arithmetischen Prozesses. Somit werden die Überlauf- (Carry) Bits ab Bit-Position 16 ignoriert
- Die Summe ist die algebraische, binäre Summe der mit dem subscript "i" gekennzeichneten Wörter
- Der Summenwert -32'768 muss als spezieller Fall behandelt werden, da dieser nicht negiert werden kann

Log Request Messages

Erfassen von Anfrage-Nachrichten

Bild 3.4-2 zeigt das Format des Datenteils einer Standard- Anfrage-Nachricht. Die Bereiche für die Wörter 6, 7 und 8 sind die folgenden:

Trigger

- 0 = on time (fest), 1 = on update (bei aktualisierung)

Intervall

- 0 bis 65'535 Sekunden (wenn = 0, wird eine Anfrage ausgelöst, sofern das query Bit [Q] im Header-Wort 4 gesetzt ist)

Offset

- 0 bis 60 Sekunden (bei 0 erfolgt die Ausgabe relativ zur aktuellen Zeit, bei einem Offset von 60 relativ zu der nächstfolgenden Minute [Null Sekunden in der nächsten Minute])

Trigger (on time, on update)	Wort 6
Intervall (sec)	Wort 7
Offset (sec)	Wort 8
Daten-Checksumme	Wort 9

Bild 3.4-2: Standard Log Request Message Format (Daten-Bereich)

Erweiterte Anfrage-Nachricht

Wenn das Trigger-Feld auf “on time” gesetzt ist, erfolgt die Ausgabe mit der nächsten Offset-Sekunde in der Minute und wird jede Intervall-Sekunde danach repetiert. Bei “on update” wird die spezifizierte Nachricht nur ausgegeben, wenn die Daten aktualisiert werden (z.B. wenn ein Satelliten-Almanach geladen wird).
Bild 3.4-3 zeigt das Format des Datenteils einer erweiterten Anfrage-Nachricht. Die ausgedehnte Erfassungs-Fähigkeit ist ***TBD***.

Trigger (on time, on update)	Wort 6
Intervall (sec)	Wort 7
Offset (sec)	Wort 8
Nachrichten-abhängige Daten	Wort 9
...	Wort 9+n
Daten-Checksumme	Wort 10+n

Bild 3.4-3: Extended Log Request Message Format (Daten-Bereich)

Einige Nachrichten unterstützen beide Formate. Eine Standard-Ephemeriden-Anfrage zum Beispiel, kann viele verschiedene Sätze von Ephemeriden-Daten ausgeben, einen für jeden Satelliten, den der Empfänger ins Auge gefasst hat.

3.4.1.3 Binary Message Data

Daten-Teil

Der Daten-Teil kann, sofern er existiert, in der Länge variabel sein, wie durch den Datenwort-Zähler im Header-Teil angegeben. Die Daten-Checksumme folgt nach den Daten und wird durch den Datenwort-Zähler nicht berücksichtigt.

Die Daten-Checksumme ist ein 16-Bit Wort und wird bei jeder Nachricht, welche Daten enthält, als letztes Wort übertragen (siehe Bild *4-2* oder *4-3*).

Wenn der Datenwort-Zähler = 0 ist, existiert auch die Daten-Checksumme nicht. Diese berechnet sich aus der Summe (modulo 216) aller sich im Daten-Teil befindlichen Wörter und anschliessender Komplement-Bildung der Summe.

If SUM = -32’768, Daten-Checksumme = SUM; else Daten-Checksumme = -SUM

Die Berechnung der Header-Checksumme lässt sich mathematisch wie folgt ausdrücken:

$$SUM = Mod2^{16} \sum_{i=6}^{5+n} Word(i)$$

Wobei die gleichen Bedingungen gelten wie bei der Header-Checksumme.

Sämtliche Datenelemente welche als Reserve bezeichnet sind, müssen für Eingabe-Nachrichten auf Null gesetzt werden und sind undefiniert für Ausgabe-Nachrichten. Wenn nichts anderes angegeben, ist die Auflösung für jeden numerischen Daten-Ausdruck eine Integer-Einheit.

3.4.1.4 NMEA-Nachrichten-Format und Satz-Struktur

NMEA-Nachrichten sind Ausgaben gestützt auf die Standard Q (Query) oder Hersteller-spezifischen ILOG (Log Control) Nachrichten wie nachfolgend beschrieben. Das Timing der Nachrichten-Ausgabe ist synchronisiert mit dem Time Mark-(TM, 1kHz) Ereignis.

NMEA-Ausgabe-Nachrichten

NMEA-Ausgabe-Nachrichten

Die folgenden unterstützten NMEA-Ausgabe-Nachrichten sind kompatibel zum NMEA-0183 Standard, Version 2.01.

- GGA: GPS Fix Data (feste Daten) *GGA*
- GSA: GPS DOP and Active Satellites (DOP und aktive Satelliten) *GSA*
- GSV: GPS Satellites in View (sichtbare Satelliten) *GSV*
- RMC: Recommended Minimum Specific GPS Data (empfohlene Mindest-Daten zur Pos.-Bestimmung) *RMC*

Zudem existieren die folgenden Rockwell-eigenen Ausgabe-Nachrichten:

- BIT: Rockwell Proprietary Built-In Test (BIT) Results (BIT-Resultate) *BIT*
- RID: Rockwell Proprietary Receiver ID (Empfänger-Identitäts-Nummer) *RID*
- ZCH: Rockwell Proprietary Zodiac Channel Status (Zodiac-Kanal-Status) *ZCH*

Diese geschützten Nachrichten sind konform mit dem weiter unten beschriebenen Nachrichten-Format.

NMEA-Eingabe-Nachrichten

NMEA-Eingabe-Nachrichten

Der Zodiac-Empfänger unterstützt die folgenden Hersteller- spezifischen Eingabe-Nachrichten:

- IBIT: Rockwell Proprietary Built-In Test (BIT) Command (BIT-Befehl) *IBIT*
- ILOG: Rockwell Proprietary Log Control (Erfassungs-Kontrolle) *ILOG*
- INIT: Rockwell Proprietary Receiver Initialisation (Empfänger-Initialisation) *INIT*
- IPRO: Rockwell Proprietary Protocoll (Protokoll) *IPRO*

Die INIT-Nachricht führt die Initialisierung aus und die IPRO-Nachricht wird zum Wechseln des Nachrichten-Protokolls benützt. Das erste Zeichen des Nachrichten-Satzes ist “P”, gefolgt von einem 3-Zeichen-Mnemonik-Code für Rockwell International (RWI) gemäss Anhang III des NMEA-0183-Standards.

Das NMEA-Nachrichten-Format

Das NMEA-0183-Protokoll ist ein ASCII-Format. Jeder Datensatz (oder Block) beginnt mit dem ASCII-Zeichen “$” (24H) und endet mit ASCII <CR><LF> (0DH und 0AH) [Carriage Return und Line Feed]. Der gültige Zeichensatz besteht aus allen druckbaren ASCII-Zeichen (20H bis 7EH), ausgenommen von den reservierten Zeichen, welche in *Tabelle 3.4-2* aufgeführt sind.

Zeichen	Hex-Wert	Dezimal-Wert	Beschreibung
<CR>	0D	13	Carriage Return (Satzende-Delimiter)
<LF>	0A	10	Line Feed (Satzende-Delimiter)
$	24	36	Satzanfang-Delimiter
*	2A	42	Checksummenfeld-Delimiter
,	2C	44	Datenfeld-Delimiter
!	21	33	Reserviert
\	5C	923	Reserviert
^	5E	94	Reserviert
-	7E	126	Reserviert

Tabelle 3.4-2: Reservierte Zeichen für NMEA-Protokolle

Jede NMEA-Nachricht, oder -Satz, besteht aus mehreren Daten-Feldern, welche durch einen sog. Delimiter (in diesem Fall ein Komma “,”) getrennt sind. Die Datenfelder enthalten einen String aus gültigen Zeichen, können aber auch leer sein. Gültige Zeichen müssen mit den Angaben in Tabelle I-3 (siehe *Anhang,* [gelb]) übereinstimmen.

Die maximale Zeichen-Anzahl pro Satz beträgt 82, bestehend aus maximal 79 Zeichen zwischen dem Start-Delimiter “$” und den abschliessenden Zeichen <CR> und <LF>.

Dass die Anzahl der Datenfelder von Satz zu Satz varieren kann, ist es für den Programmierer wichtig zu wissen. Um Felder zu lokalisieren, ist es somit besser die Kommas zu zählen, als sich auf die Anzahl der total empfangenen Zeichen zu verlassen. Sollten die Daten, aus welchen Gründen auch immer, einmal nicht verfügbar sein, wird der Satz (Block) trotzdem gesendet. In diesem Fall folgt das nächste Komma sofort auf das vorhergehende.

Anerkannte NMEA-0183 Sätze

Ein anerkannter NMEA-0183-Satz beinhaltet die folgenden Elemente in der Reihenfolge von oben nach unten:

“$”	Startzeichen (24H)	*“$”*
<Adress-Feld>	Satz-Identifier und -Formatierer	*<Adress-Feld>*
[“,”<Daten-Feld>]	kein oder mehrere Datenfelder	*[“,”<Daten-Feld>]*
[“,”<Daten-Feld>]	das letzte Datenfeld	*[“,”<Daten-Feld>]*
[“*”<Checksummen-Feld>]	optionale Checksumme	*[“*”<Checksummen-Feld>]*
<CR><LF>	Endzeichen (0DH, 0AH)	*<CR><LF>*

HINWEIS: Da der Zodiac-Empfänger eine GPS-Einheit ist, ist der Satz-Identifier immer “GP” !

Hersteller-spezifische Sätze

Diese erlauben dem Anwender Daten zu übertragen, die nicht in den Bereich von anerkannten NMEA-Sätze fallen. Ein Hersteller-spezifischer NMEA-Satz beinhaltet die folgenden Elemente in der Reihenfolge von oben nach unten:

“$”	Startzeichen (24H)	*“$”*
“P”	Hersteller-spezifische Satz-Identitäts-Nummer	*“P”*
<aaa>	OEM- (Anwender) Mnemonik-Code	*<aaa>*
[<gültige Daten, OEM-Daten>]		*[<gültige Daten, OEM-Daten>]*
[“*”<Checksummen-Feld>]	optionale Checksumme	*[“*”<Checksummen-Feld>]*
<CR><LF>	Endzeichen (0DH, 0AH)	*<CR><LF>*

Checksumme

Diese ist die 8-Bit exklusiv-oder-(XOR) Verknüpfung (keine Start- oder Stop-Bits) von allen Zeichen eines Satzes, inklusive den Datenfeld-Delimitern (,), jedoch ohne das Startzeichen ($) und den optionalen Checksummen-Delimiter (*). Der hexadezimale Wert der ersten vier (LSB) und der letzten vier (MSB) Bits des Resultates wird zwecks Übertragung zu zwei ASCII-Zeichen konvertiert (0-9,A-F). Das most significant Zeichen wird zuerst übertragen.

3.4.2 Zodiac binary Daten-Nachrichten

Dieser Abschnitt beschreibt die binären Daten-Nachrichten des Zodiac-GPS-Empfängers. Sämtliche der Ein- und Ausgabe-Nachrichten sind in *Tabelle 3.4-3* mit den zugehörigen Nachrichten-Identitäts-Nummern (ID) aufgelistet. Power-up Standard-Nachrichten sind ebenfalls gekennzeichnet.

Binary Messages werden über das Serielle Interface (RS-232, Haupt-Port) empfangen und gesendet. Folgende Einstellungen sind erforderlich:

[9600 bps] [8 Daten-Bits] [keine Parität (N)] [1 Stop-Bit]

HINWEIS: Die folgende Beschreibung der einzelnen Ein- und Ausgabe-Nachrichten beschränkt sich aus Zeitgründen auf einige einführende Informationen. Die angegebenen Querverweise führen jedoch direkt zu den im entspr. *Anhang* [gelb] aufgelisteten Original-Tabellen. Aus diesem Grund werden auch die Nachrichten-Überschriften in englischer Sprache belassen.

Output Message Name	Message ID	Input Message Name	Message ID
Geodetic Position Status Output (*)	1000	Map Datum Select	1211
ECEF Position Status Output	1001	Satellite Elevation Mask Control	1212
Channel Summary (*)	1002	Satellite Candidate Select	1213
Visible Satellites	1003	Differential GPS Control	1214
Differential GPS Status	1005	Cold Start Control	1216
Channel Measurement	1007	Solution Validity Criteria	1217
Receiver ID (**)	1011	Antenna Type Select	1218
User Settings Output	1012	User-Entered Altitude Input	1219
Built-In Test (BIT) Results	1100	Nav Configuration	1221
Measurement Time Mark	1102	Perform Built-In Test Command	1300
Seriel Port Communication Parameters In Use	1130	Restart Command	1303
EEPROM Update	1135	Seriel Port Communication Parameters	1330
EEPROM Status	1136	Message Protocol Control	1331
Geodetic Position and Velocity Initialisation	1200	Raw DGPS RTCM SC-104 Data	1351

(*) = aktiviert bei beim Einschalten (Standard)
(**) = einmal beim Einschalten / Reset

Tabelle 3.4-3: Zodiac binary Daten-Nachrichten

3.4.2.1 Ausgabe-Nachrichten Beschreibung

Geodetic Position Status Output (Message 1000)

Gibt die vom Empfänger geschätzten Werte der Position, Geschwindigkeit, Kurs, Steigung und das Kartendatum aus. Ein Lösungs-Status gibt aufgrund der eingestellten Lösungskriterin an, ob die Lösung gültig ist. Ebenso wird die Anzahl der für die Lösung benötigten Messungen angezeigt. Das Polar-Navigations-Flag zeigt an, ob die Näherungslösung näher beim Nord- oder Südpol des Längengrades liegt. Wenn das Flag true ist, sind die Längen- und Kursangaben ungültig und nicht aktualisiert. Anwender in der Nähe der Pole sollten die ECEF Position Status Message benützen.

Message 1000
Tabelle II-2, Anhang 3.8.4

ECEF Position Status Message (Message 1001)

Gibt die vom Empfänger geschätzte ECEF- (Earth Centered, Earth Fixed) Position und Geschwindigkeit sowie das Kartendatum aus. Ein Lösungs-Status gibt aufgrund der eingestellten Lösungskriterin an, ob die Lösung gültig ist. Ebenso wird die Anzahl der für die Lösung benötigten Messungen angezeigt.

Message 1001
Tabelle II-3, Anhang 3.8.4

Channel Summary (Message 1002)

Beinhaltet eine Kurzfassung der Satelliten-Bereichsmessungen und Signalträger-Informationen für jeden Kanal.

Message 1002
Tabelle II-4, Anhang 3.8.4

Visible Satellites (Message 1003)

Gibt die Liste der sichtbaren Satelliten und deren korrespondierende Höhenlage sowie den Azimut aus. Ebenso sind die bestmöglichen DOP-Werte, berechnet aus den geometrischen Verhältnissen der Satelliten, ersichtlich.

Message 1003
Tabelle II-5, Anhang 3.8.4

Differential GPS Status (Message 1005)

Beinhaltet DGPS-Status-Informationen vom letzten Satz der durch den Empfänger ausgeführten DGPS-Korrekturen.

Message 1005
Tabelle II-6, Anhang 3.8.4

Channel Measurement (Message 1007)

Beinhaltet Messungen und vereinte Daten für jeden der 12 Kanäle des Empfängers.

Message 1007
Tabelle II-7, Anhang 3.8.4

Receiver ID (Message 1011)

Diese Nachricht wird nach dem Start automatisch ausgegeben, sobald der Empfänger die Initialisation beendet hat. Sie gibt an, wann der Empfänger für serielle Eingaben bereit ist. Manuelle Anfragen für diese Nachricht sind ebenfalls möglich. Sie besteht aus fünf 20-Byte (zwei Zeichen pro Wort), Null-padded (mit Null belegten) ASCII-Datenfeldern.

Message 1011
Tabelle II-8, Anhang 3.8.4

User Settings Output (Message 1012)

Verfügt über eine Kurzfassung (Zusammenzug) von vielen Benutzer-definierten Einstellungen, welche entweder zu Standard-Werten werden oder zu Werten, die der Benutzer in Eingabe-Nachrichten bestimmt.

Message 1012
Tabelle II-9, Anhang 3.8.4

Built-In Test (BIT) Results (Message 1100)

Verfügt über detaillierte Test-Resultate des seit dem letzten Start ausgeführten BIT. Diese Nachricht wird nach einem beendeten BIT-Befehl automatisch ausgegeben, kann aber bei Bedarf auch manuell angefordert werden. Ein Fehler liegt vor, wenn der Einheiten-Fehler-Status NICHT NULL ist.

Message 1100
Tabelle II-10, Anhang 3.8.4

Measurement Time Mark (Message 1102)

Diese Nachricht verfügt über “rohe” Messungs- und vereinte andere Daten.

Message 1102
Tabelle II-11, Anhang 3.8.4

Seriel Port Communication Parameters In Use (Message 1130)

Beinhaltet die Kommunikationsparameter für die beiden Seriellen Schnittstellen (Haupt- und Hilfsport) des Empfängers.

Message 1130
Tabelle II-12, Anhang 3.8.4

EEPROM Update (Message 1135)

Verfügt über dynamische Status-Meldungen der EEPROM-Beschreibung. Sie beinhaltet die Datenblock-ID für den letzten Datensatz, der ins EEPROM geschrieben wurde. Diese Nachricht ist sehr nützlich, wenn sie für die “Ausgabe bei Aktualisierung” konfiguriert wird (Standard), da sie Informationen über alle gespeicherten Konfigurations-Änderungen enthält.

Message 1135
Tabelle II-13, Anhang 3.8.4

EEPROM Status (Message 1136)

Verfügt über Fehler- und Speicherstatus-Informationen des EEPROM’s. Die gesetzten Bits in den fehlerhaften Wörtern repräsentieren Schreibfehler während des Versuches, den entsprechenden Datenblock zu aktualisieren. Die gesetzten Bits in den Status-Wörtern zeigen an, dass diese Datenblocks im EEPROM am wenigsten aktualisiert wurden.

Message 1136
Tabelle II-14, Anhang 3.8.4

3.4.2.2 Eingabe-Nachrichten Beschreibung

Geodetic Position and Velocity Initialisation (Message 1200)

Diese Nachricht erlaubt dem Benutzer die Initialisation des Empfängers mit der spezifischen geodätischen Position, Geschwindigkeit, Kurs über Grund und Steigung. Der Kurs kann entweder “true” (geografisch) oder magnetisch sein, wie durch das “Magnetic Course”-Feld angezeigt. Die GPS/UTC-Zeit repräsentiert die Zeit, zu welcher die Lösung berechnet wurde. Zudem wird sie zur Ausbreitung (Übertragung) der Lösung auf die aktuelle Zeit benutzt.

Message 1200
Tabelle II-15, Anhang 3.8.4

Map Datum Select (Message 1211)

Erlaubt dem Benutzer ein Datum des Empfängers zur Umwandlung der Positions-Lösung zu wählen.

Message 1211
Tabelle II-16, Anhang 3.8.4

Satellite Elevation Mask Control (Message 1212)

Erlaubt dem Benutzer die Eingabe des Erhebungswinkels, welcher die durch den Empfänger sichtbaren Satelliten bestimmt. Der Erhebungswinkel wird im EEPROM gespeichert.

Message 1212
Tabelle II-17, Anhang 3.8.4

Satellite Candidate Select (Message 1213)

Erlaubt dem Benutzer die Zusammenstellung einer Liste von Satelliten, die durch den Empfänger bei der Auswahl berücksichtigt werden.

Message 1213
Tabelle II-18, Anhang 3.8.4

Differential GPS Control (Message 1214)

Erlaubt dem Benutzer das Verhalten des Empfängers bezüglich seiner “differential”-Fähigkeiten zu kontrollieren. Diese Nachrichten-Parameter werden im EEPROM gespeichert.

Message 1214
Tabelle II-19, Anhang 3.8.4

Cold Start Control (Message 1216)

Erlaubt dem Benutzer den Kalt-Start-Erfassungs-Modus des Empfängers zu deaktivieren. Im Normal-Betrieb ist dieser Modus aktiviert. Trotzdem, in manchen Situationen (z.B. Tiefgaragen, Häuser, Bürogebäuden) ist mit deaktiviertem Kalt-Start eine schnellere Erfassung möglich. Dieser Parameter (disable) wird im EEPROM gespeichert.

Message 1216
Tabelle II-20, Anhang 3.8.4

Solution Validity Criteria (Message 1217)

Der Empfänger gibt immer die beste Positionslösung aus, die möglich ist, abhängig von der Anzahl und Qualität der gültigen Messungen. Diese Nachricht gewährt dem Benutzer die Kriterien für den Positions-Gültigkeits-Status selbst zu definieren, welcher in der Positions-Ausgabe-Nachricht angegeben ist. Der Status ist “ungültig”, wenn eine der Kriterien nicht erfüllt ist. Diese Nachrichten-Parameter werden im EEPROM gespeichert.

Message 1217
Tabelle II-21, Anhang 3.8.4

Antenna Type Select (Message 1218)

Erlaubt dem Benutzer die Auswahl des Antennen-Typs. Eine aktive Antenne wird die Amplitude des Empfänger-Signal-Levels erhöhen, um Seitenband-Korrelationen zu reduzieren. Eine passive Antenne macht den Empfänger empfindlicher für tiefe Signal-Level und beschützt ihn vor “searching hot signals”. Dieser Parameter wird im EEPROM gespeichert.

Message 1218
Tabelle II-22, Anhang 3.8.4

User-Entered Altitude Input (Message 1219)

Erlaubt dem Benutzer die Eingabe einer Höhe für die 2D-Navigation. Wenn das “Force Use”-Feld nicht gesetzt ist, kann der Empfänger die Eingabe ignorieren, wenn er denkt er habe einen besseren Schätzwert.

Message 1219
Tabelle II-23, Anhang 3.8.4

Beim Setzen des “Clear”-Feldes wird der letzte Höhenlage-Wert gelöscht. Setzen des “MSL Select”-Feldes erlaubt die Eingabe der Meereshöhe. Eine Standard-Abweichung kann in Bezug auf die Unsicherheit der angegebenen Höhe explizit angegeben werden. Der Empfänger wird diese Unsicherheit in die Höhenmessungen miteinbeziehen. Als spezieller Fall kann der Wert NULL eingegeben werden, wenn die Qualität der Höhe nicht bekannt ist.

Nav Configuration (Message 1221)

Message 1221
Tabelle II-24, Anhang 3.8.4

Erlaubt dem Benutzer die Kontrolle über diverse Eigenschaften des Navigations-Prozesses. Das “Held Altitude Disable”-Bit kontrolliert die Benutzung von gespeicherter GPS-basierender Höhe um den Empfänger zu unterstützen (helfen), wenn die vertikale Geometrie an Wert verliert (sich verschlechtert). Das “Ground Track Smoothing”-Bit kontrolliert die Benutzung der geschätzten Satelliten-Bereichsabweichung, um die Positionsverschiebung resultierend durch die SA und die Konstellationen-Wechsel zu minimieren. Das “Position Pinning”-Bit kontrolliert die Benutzung eines horizontalen Geschwindigkeits-Tests um die durch den Empfänger ermittelte Position zu “befestigen” (to pin) und die Abweichung durch die SA bei stehender Position zu eliminieren. “Ground Track Smoothing” und “Position Pinning” werden NICHT benützt, wenn DGPS-Korrekturen verwendet werden.

Perform Built-In Test Command (Message 1300)

Message 1300
Tabelle II-25, Anhang 3.8.4

Diese Nachricht beauftragt den Empfänger zu einer sofortigen Ausführung des Built-In Test’s (BIT). Die Resultate des BIT’s sind in der Nachricht 1100 ersichtlich. Beachten Sie, dass diese Nachricht keine Daten enthält.

Restart Command (Message 1303)

Message 1303
Tabelle II-26, Anhang 3.8.4

Befiehlt einen vollen Neu-Start zu jeder Zeit.

Seriel Port Communication Parameters (Message 1330)

Message 1330
Tabelle II-27, Anhang 3.8.4

Erlaubt dem Benutzer die Eingabe der Parameter für die beiden Seriellen Schnittstellen (Haupt- und Hilfsport) des Empfängers.

Message Protocol Control (Message 1331)

Message 1331
Tabelle II-28, Anhang 3.8.4

Erlaubt dem Benutzer die Auswahl des Nachrichten-Format-Protokolles, das zur Kommunikation von Informationen zum und vom Empfänger über den Host-Port benützt werden soll. Zur Auswahl stehen das Rockwell binary und das NMEA-0183-Protokoll. Diese Parameter werden im EEPROM gespeichert.

Raw DGPS RTCM SC-104 Data (Message 1351)

Message 1351
Tabelle II-29, Anhang 3.8.4

Beinhaltet DGPS RTCM SC-104-Korrekturdaten. Sie ist abwärtskompatibel zum dem älteren MicroTracker GPS-Empfänger und kann anstelle der Hilfsport-Daten verwendet werden.

3.4.3 Zodiac NMEA-Daten-Nachrichten

Dieser Abschnitt beschreibt die NMEA-Daten-Nachrichten des Zodiac-GPS-Empfängers. Sämtliche der Ein- und Ausgabe-Nachrichten sind in *Tabelle 3.4-4* mit den zugehörigen Nachrichten-Identitäts-Nummern (ID) aufgelistet. Power-up Standard-Nachrichten sind ebenfalls gekennzeichnet.

NMEA Messages werden über das Serielle Interface (RS-232, Haupt-Port) empfangen und gesendet. Folgende Einstellungen sind erforderlich:

[4800 bps] [8 Daten-Bits] [keine Parität (N)] [1 Stop-Bit]

HINWEIS: Die folgende Beschreibung der einzelnen Ein- und Ausgabe-Nachrichten beschränkt sich aus Zeitgründen auf einige einführende Informationen. Die angegebenen Querverweise führen jedoch direkt zu den im entspr. *Anhang* [gelb] aufgelisteten Original-Tabellen. Aus diesem Grund werden auch die Nachrichten-Überschriften in englischer Sprache belassen.

Output Message Name	Message ID
Rockwell Proprietary Built-In Test	BIT
GPS Fix Data (*)	GGA
GPS DOP and Active Satellites (*)	GSA
GPS Satellites in View (*)	GSV
Recommended Minimum Specific GPS Data (*)	RMC
Rockwell Proprietary Receiver ID	RID
Rockwell Proprietary Zodiac Channel Status (*)	ZCH
Input Message Name	**Message ID**
Rockwell Proprietary Built-In Test (BIT) Command	IBIT
Rockwell Proprietary Log Control Message	ILOG
Rockwell Proprietary Receiver Initialisation	INIT
Rockwell Proprietary Protocol Message	IPRO

(*) = Standard-Nachricht beim Einschalten

Tabelle 3.4-4: Zodiac NMEA-Daten-Nachrichten

3.4.3.1 Ausgabe-Nachrichten Beschreibung

Rockwell Proprietary Built-In Test (BIT) Results (BIT)

Diese Hersteller-spezifische Nachricht enthält detaillierte BIT-Resultate. Ein Fehler liegt vor, wenn der Einheiten-Fehler-Status NICHT NULL ist.

BIT Message
Tabelle III-2, Anhang 3.8.4

GPS Fix Data (GGA)

Beinhaltet Zeit-, Positions- und Fixdaten für den Zodiac GPS-Empfänger. Wenn eine Navigationslösung allen Gültigkeits-Kriterien entspricht (siehe binary message 1217), wird diese Nachricht automatisch generiert. Andererseits, wenn nur eine der Kriterien für die Lösung ungültig ist, wird die Nachricht nicht generiert.

GGA Message
Tabelle III-3, Anhang 3.8.4

GPS DOP and Active Satellites (GSA)

Beinhaltet den Zodiac-Empfänger-Betriebs-Modus, für die Navigation benützte Satelliten und DOP-Werte.

GSA Message
Tabelle III-4, Anhang 3.8.4

GPS Satellites in View (GSV)

Beinhaltet die Anzahl der sichtbaren Satelliten, PRN-Nummern, Höhe, Azimut und Signal zu Rausch-Verhältnis- (SNR)-Werte. Jede Übertragung identifiziert bis zu max. vier Satelliten; zusätzliche Satellitendaten erfordern eine zweite oder dritte Nachricht. Die totale Anzahl und die jeweilige Nummer der übertragenen Nachrichten wird in den ersten zwei Feldern angezeigt.

GSV Message
Tabelle III-5, Anhang 3.8.4

Recommended Minimum Specific GPS Data (RMC)

Beinhaltet Zeit-, Datums-, Positions-, Kurs- und Geschwindigkeitsdaten. Die Felder dieser Nachricht enthalten auch Daten, wenn der Empfänger nicht navigiert. Dies erlaubt Benutzer-initialisierte, gespeicherte oder Standard-Werte anzuzeigen, bevor eine Lösung erhältlich ist.

RMC Message
Tabelle III-6, Anhang 3.8.4

Rockwell Proprietary Receiver ID (RID)

Diese Nachricht wird nach dem Start automatisch ausgegeben, sobald der Empfänger die Initialisation beendet hat. Sie gibt an, wann der Empfänger für serielle Eingaben bereit ist. Manuelle Anfragen für diese Nachricht sind ebenfalls möglich.

RID Message
Tabelle III-7, Anhang 3.8.4

Rockwell Proprietary Zodiac Channel Status (ZCHG)

Diese Nachricht ergänzt die GSV-Nachricht mit einer Satelliten-zu-Kanal-Abbildung und einer Status-Anzeige für jeden Kanal.

ZCH Message
Tabelle III-8, Anhang 3.8.4

3.4.3.2 Eingabe-Nachrichten Beschreibung

Rockwell Proprietary Built-In Test (BIT) Command Message (IBIT)

Diese Hersteller-spezifische Nachricht instruiert den Empfänger zu einer sofortigen Ausführung des BIT. Die Resultate sind in der Rockwell Proprietary Built-In Test (BIT) Results-Nachricht ersichtlich. Die Datenfelder sind reserviert und müssen NULL sein.

IBIT Message
Tabelle III-9, Anhang 3.8.4

Rockwell Proprietary Log Control Message (ILOG)

Kontrolliert die Ausgabe der NMEA-Nachrichten des Zodiac GPS-Empfängers.

ILOG Message
Tabelle III-10, Anhang 3.8.4

Rockwell Proprietary Receiver Initialisation Message (INIT)

Befiehlt dem Zodiac-Empfänger die Ausführung eines Resets, die Modifizierung des Betriebs-Modus oder die Re-Initialisierung mit eigenen, spezifischen Parametern.

INIT Message
Tabelle III-11, Anhang 3.8.4

Rockwell Proprietary Protocol Message (IPRO)

Erlaubt dem Benutzer die Auswahl des Nachrichten-Format-Protokolles, das zur Kommunikation von Informationen zum und vom Empfänger über den Host-Port benützt werden soll. Zur Auswahl stehen das Rockwell binary und das NMEA-0183-Protokoll. Diese Parameter werden im EEPROM gespeichert.

IPRO Message
Tabelle III-12, Anhang 3.8.4

3.5 VISYS® GPS ADAPTOR

VISYS GPS Adaptor

⑤

3.5.1 Serial and Power Adaptor Card for Rockwell Jupiter™ Board

3.5.1.1 Allgemeines

Allgemeines

Die GPS-Adapter-Karte ist entwickelt worden, um (basierend auf dem Zodiac Chipsatz) eine einfache Schnittstelle zu Rockwell's erstem12-Kanal Empfänger-Board JUPITER™ zur Verfügung zu stellen.

Die Karte besteht aus einer Spannungsversorgung, Seriellen Schnittstellen und Anschlüssen um einen einfachen Zugriff auf die Features des Jupiter™-Boards zu gewährleisten.

Die Speisung hat einen grossen Eingangsspannungs-Bereich von 7-18V DC und generiert die +5V DC-Speisespannung für das Jupiter™-Board. Ein Batterie-Halter für den Einsatz einer Lang-Zeit-Speisung der UTC-Uhr existiert bei fehlender Hauptspannung ebenfalls. Um eine aktive Antenne zu betreiben, ist aus Sicherheitsgründen auch ein Sicherungshalter vorhanden.

Zudem verfügt die Adapter-Karte über zwei RS-232-fähige serielle Schnittstellen, wovon eine als Haupt-Port (EIN- und AUSgabe) und die andere als Hilfsport für den Empfang von DGPS-Korrekturdaten verwendet werden kann. Für eine einfache Betriebskontrolle wird das Time Mark-(TM) Signal mittels einer blinkenden LED angezeigt. Schalter für den Zugriff auf verschiedene Kommunikations-Setups sind auf Wunsch erhältlich.

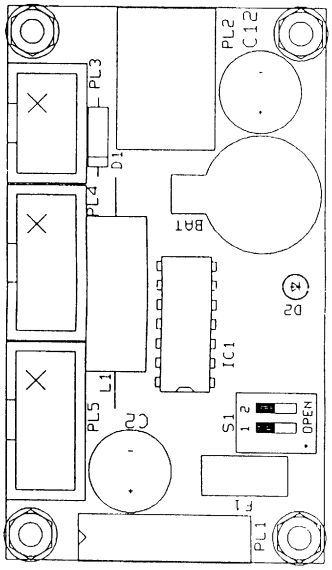


Bild 3.5-1: GPS-Adaptor-Card

3.5.1.2 Eigenschaften

Eigenschaften

Die folgenden Angaben sind auf das Wesentliche beschränkt. Für weiterführende Informationen sei auf den entspr. *Anhang* verwiesen.

Elektrisch

Spannungsversorgung	Normal 12V DC, (7-18V), max. 150 mA	*Elektrisch*

Physisch

Länge x Breite x Höhe	70mm x 40 mm x 20 mm	*Physisch*

Anschlüsse

PL1	Interface zu Jupiter™-Board	*Anschlüsse*
PL2	Spannungsversorgungs-Eingang DIN45323	
PL3	Verschiedene Signale	
PL4	DGPS-Eingang	
PL5	Haupt-Port	

3.5.1.3 Interface zu Jupiter™-Board

Diese Verbindung versorgt die Adapter-Karte mit den vom Jupiter™-Board empfangenen und ausgewerteten Daten.

Pin	Bezeichnung	Beschreibung
1	PREAMP	PREAMP POWER INPUT / Vorverstärker Power-Anschluss
2	PWRIN_5	PRIMARY +5VDC POWER INPUT / Hauptspeisungs-Anschluss +5V DC
3	VBRTC	BACKUP BATTERY (RTC) / Back-Up-Batterie für die RTC-Einheit
4	PWRIN_3	RESERVED (NOT CONNECTED) / Reserve (nicht angeschlossen)
5	M_RST	MASTER RESET INPUT (ACTIVE LOW) / Master-Reset Eingang (low=aktiv)
6	GPIO1	RESERVED (NOT CONNECTED) / Reserve (nicht angeschlossen)
7	GPIO2	NMEA PROTOCOL SELECT / NMEA-Protokoll Auswahl (LOW = aktiv)
8	GPIO3	ROM DEFAULT SELECT / Standard ROM-Daten Auswahl (0=ROM / 1=SRAM/EEPROM)
9	GPIO4	RESERVED (NOT CONNECTED) / Reserve (nicht angeschlossen)
10	GND	GROUND
11	SDO1	SERIAL DATA OUTPUT PORT #1 / serielle Datenausgabe Port 1
12	SDI1	SERIAL DATA INPUT PORT #1 / serielle Dateneingabe Port 1
13	GND	GROUND
14	SDO2	SERIAL DATA OUTPUT PORT #2 / serielle Datenausgabe Port 2
15	SDI2	SERIAL DATA INPUT PORT #2 / serielle Dateneingabe Port 2
16	GND	GROUND
17	N/C	RESERVED (NOT CONNECTED) / Reserve (nicht angeschlossen)
18	GND	GROUND
19	TMARK	1PPS TIME MARK OUTPUT / synchronisiertes Zeit-Marken-Signal von 1Hz
20	RESERVED	RESERVED (NOT CONNECTED) / Reserve (nicht angeschlossen)

Tabelle 3.5-1: Pinbelegung “Jupiter”-Empfänger Standard 2x10 Pin OEM-Schnittstellen-Anschluss

Für eine ausführlichere Interface-Signalbeschreibung sei auf das spezifische *Zodiac GPS Receiver Data Sheet* verwiesen, welches sich im Anhang zu Abschnitt *3.3* [blau] befindet.

3.6 Abkürzungen & Glossar

Die folgende Auflistung ist eine Auswahl von spezifischen Bezeichnungen im Zusammenhang mit GPS und aus Buch [7] entnommen:

Ursprüngliche Bezeichnung astronomischer Ephemeriden bzw. kalenderartiger Tafeln mit Angaben der Planetenbewegungen mit beigefügten astrologischen und sonstigen Notizen.	***Almanach***
Amerikanischer, nahezu weltweit gültiger Standard für die digitale Zeichenübermittlung ganz allgemein für Geräte, die Informationen austauschen, wie z.B. Fernschreiber, Rechner, u.a.	***ASCII***
Eine zur Zeitmessung höchster Genauigkeit verwendete Hochfrequenzeinrichtung, bei der die bekannte Resonanzfrequenz bestimmter Atom- oder Molekülschwingungen oder -übergänge als Vergleichsmass dient.	***Atomuhr***
Bezogen auf die Horizontalebene des Beobachters in der Navigation der Winkel zwischen geographisch Nord und einer Linie zwischen Beobachter und angepeiltem Objekt, wobei von Norden aus im Uhrzeigersinn gemessen wird.	***Azimut***
Allgemein gebräuchlicher Begriff für den asymmetrischen Fehlerbereich eines Signales, d.h. das Signal ist mit einem systematischen Fehler behaftet.	***Bias***
Winkelangabe zur Bezeichnung eines Standortes auf der Erde bezogen auf die Nord-Süd-Richtung; die angegebene Grösse bezeichnet den Winkel zwischen Äquator-Ebene und Standort.	***Breite, geographische (engl. latitude)***
GPS Code geringerer Auflösung, der für die zivile Nutzung vorgesehen ist.	***C/A-Code*** **Coarse / Acquisition**
Bezugssystem für die Vermessung der Erde; als geodätisches Datum bezeichnet man die Beziehung zwischen einem lokalen Ellipsoiden und einem globalen Referenzsystem.	***Datum, geodätisches***
Winkelabstand eines Himmelskörpers vom Himmelsäquator	***Deklination***
Verfahren zur Genauigkeitssteigerung von GPS-Daten; im Rahmen des DGPS werden über vermessene Sender am Boden Korrekturdaten ermittelt und über Sendestationen an den Nutzer weitergegeben.	***DGPS***
U.S. **D**epartment **o**f **D**efense (amerikanisches Verteidigungsministerium)	***DoD***
Dilution **o**f **P**recision (Fehleranteil, der bei der Positionsbestimmung mit GPS aufgrund der Satellitenkonstellation entsteht).	***DOP***
Eart **C**entered, **E**art **F**ixed (Koordinatensystem, welches auf den Mittelpunkt der Erde bezogen ist).	***ECEF***
Astronomische Tafeln oder nautische Jahrbücher, in denen für zeitlich konstante Abstände die vorausberechneten Orte der Gestirne an der Himmelskugel veröffentlicht werden.	***Ephemeriden***
Im Zusammenhang mit GPS die Voraussage über die aktuelle Satellitenposition, die dem Nutzer im Rahmen der Datenübermittlung gesendet wird.	***Ephemeris***

Geodäsie
Wissenschaft und Technik der Bestimmung von Form und Grösse der Erde; grundlegende Bedeutung kommt dabei auch der Bestimmung des Schwerefeldes der Erde zu.

Geoid
Bezeichnung für den von der wahren Erdgestalt abweichenden theoretischen Körper, der diejenige Niveaufläche des Erdschwerefeldes beschreibt, die sich den Ozeanen am besten anpasst und die als Bezugsfläche für Höhenangaben gewählt wird.

Geoidundulation
Senkrechter Abstand zwischen dem Erdellipsoid und dem ⇒ Geoid, der 100m kaum übersteigt.

GLONASS
Global **Na**vigation **S**atellite **S**ystem (russisches Satelliten-Navigations-System mit vergleichbaren Eigenschaften zu GPS).

GMT
Greanwich **M**ean **T**ime (die westeuropäische Zeit, die für den Meridian von Greenwich gilt).

GPS
Satelliten-Navigationssystem in der Verantwortung des amerikanischen Verteidigungsministeriums.

Inklination
Winkel zwischen der Bahnebene eines Planeten oder Satelliten und der Ebene der Ekliptik oder auch der Winkel zwischen einer Satellitenbahn und der Äquatorebene des betreffenden Planeten.

Länge, geographische
Die geographische Länge ist der Winkel zwischen dem ⇒ Längenkreis, auf dem sich ein Punkt befindet, und dem durch Greenwich verlaufenden Nullmeridian.

Längenkreis
Ein durch die Erdpole verlaufender Grosskreis auf der Erdkugel; ein halber Längenkreis ist ein ⇒ Meridian.

Meridian
In der Geographie versteht man unter Meridian die Hälfte eines Grosskreises, der durch die beiden Erdpole verläuft und der den Äquator senkrecht schneidet; der Nullmeridian geht durch die Sternwarte Greenwich; er ist Ausgangsmeridian für die geographische ⇒ Länge; in der Astronomie ist der Meridian der grösste Kreis am Himmel, der durch Zenit und Pol geht und in dessen Ebene der Ort des irdischen Beobachters liegt.

MEZ
Mittel**e**uropäische **Z**eit
Bei dem Übergang von der Ortszeit zu Zeitzonen wurde die MEZ eingeführt, bei der innerhalb einer Zeitzone einheitlich die mittlere Sonnenzeit eines vereinbarten Längengrades gilt; MEZ entspricht ⇒ UTC plus eine Stunde.

MESZ
Mittel**e**uropäische **S**ommer **Z**eit
Verschiebung der MEZ um eine Stunde, um den Tagesablauf besser den hellen Zeiten des Tages anzupassen; MESZ entspricht ⇒ UTC plus zwei Stunden.

NAVSTAR
Navigation **S**atellite **T**iming **A**nd **R**anging (der Begriff gehört zur Gesamtbezeichnung des amerik. Satelliten-Navigationssystems **NAVSTAR GPS**).

NMEA
National **M**arine **E**lectronics **A**ssociation
Genormter Schnittstellenstandard vorwiegend für den Datenaustausch von elektronischen Geräten in der Seefahrt.

Precise **P**ositioning **S**ervice — ***PPS***
Bezeichnung für den militärischen Teil des Satelliten-Navigationssystems GPS; durch eine höhere Frequenz der Satellitencodes und durch die Nutzung einer zweiten Frequenz wird eine höhere Genauigkeit als im Standard-Modus ⇒ SPS erreicht, der für zivile Nutzer vorgesehen ist. Der im Rahmen des PPS gesendete P-Code wurde mit der Erklärung der vorläufigen Erprobungsphase verschlüsselt und steht seither nur noch autorisierten Anwendern zur Verfügung, die über die entsprechenden Dekodiereinrichtungen verfügen.

Pseudo **R**ange — ***PR***
Scheinbare Entfernung zu einem Satelliten; bei Pseudo-Entfernungsmessungen sind die tatsächlichen Entfernungen zu einem Satelliten mit systematischen, im allgemeinen aber für alle Kanäle mit gleichen Abweichungen behaftet, die im Laufe der Auswertung korrigiert werden. Die Abweichungen beruhen in erster Linie darauf, dass Satellitenuhr und Empfängeruhr nicht synchronisiert sind.

Radio **T**echnical **C**ommission for **M**aritime Services (amerikanische Kommission für die Erarbeitung von Kommunikationsstandards für die Marine. — ***RTCM***

Standard **P**ositioning **S**ervice — ***SPS***
Bezeichnung für den zivilen Teil des Satelliten-Navigationssystems GPS; durch eine niedrigere Frequenz der Satellitencodes und durch die Nutzung nur einer Frequenz kann der zivile Nutzer nur eine geringere Genauigkeit erreichen, als der militärische Nutzer des ⇒ PPS; für SPS-Nutzer ist zusätzlich eine künstliche Verschlechterung (⇒ SA) eingeschaltet.

Space **V**ehicle (Kurzzeichen für Satellit) — ***SV***

Time **T**o **F**irst **F**ix (Zeitraum, den ein Satelliten-Navigationsempfänger vom Einschalten bis zur Darstellung der ersten Positionsangabe braucht. — ***TTFF***

Universal **T**ime — ***UT***
Die mittlere Sonnenzeit für den Nullmeridian (Greenwich), gezählt von Mitternacht an; die Weltzeit (UT) beruht auf der astronomischen Zeiteinteilung.

Universal **T**ime **C**oordinated — ***UTC***
Koordinierte Weltzeit auf der Grundlage einer Atomzeitskala, die durch Einfügen von Schaltsekunden in annähernde Übereinstimmung mit der aus der Erddrehung hergeleiteten astronomischen Weltzeit (⇒ UT) gehalten wird.

World **G**eodetic **S**ystem von 19**84** — ***WGS 84***
Militärische Version eines globalen Geodätischen Referenzsystems und Grundlage für GPS; Nachfolger des WGS 60.

Der Punkt am Himmelsgewölbe genau senkrecht über dem Beobachter. — ***Zenit***

3.7 Allg. Hinweise und Bemerkungen

Feststellungen

Bei der Inbetriebnahme des Empfängers (Serial and Power Adaptor Card) mit der LABMON-Software habe ich folgendes festgestellt:

- Bei Betrieb unter Windows 3.11 arbeitete die Software nur unter DOS. Abgesehen davon, dass sich die Maus verabschiedete, konnte ich das Programm wohl aus einer DOS-BOX starten, jedoch keine Daten empfangen.
- Im Gegensatz dazu arbeitete die Software unter WIN95 einwandfrei (vorausgesetzt man konfiguriert zuerst die entspr. Schnittstelle in der Systemsteuerung ⇒ gilt auch für Windows 3.xx).
- Weiter stellte ich fest, dass beim ersten Start des Programmes von den beiden Konfigurationsdateien LABMON.INI bzw. LABMON.CFG tatsächlich automatisch eine “Sicherheitskopie” erzeugt wurde, welche im Grundverzeichnis (Root) der Festplatte (Windows 3.11), bzw. im Verzeichnis C:\Windows (WIN95) abgelegt werden.

3.7.1 Probleme

Probleme

Einstellen des Protokolles

- Das schnelle Umschalten zwischen den beiden Protokollen Zodiac binary und NMEA-0183 war nicht so einfach wie im Manual angepriesen. Abgesehen davon, dass ich den Betriebsmodus mit den beiden Schaltern S1/1 bzw. S1/2 vorgängig hardwaremässig einstellen musste, hatte auch die Software Mühe, die Konfigurationseinstellungen so schnell zu ändern bzw. überhaupt anzunehmen. Auch wenn ich die benötigten Parameter wie COM-Einstellungen (<Alt>+F1), Protokoll (<Alt>+F2), Zeit (F1) und Position (F2) erneut setzte, endete dies mit der Fehlermeldung:
  MESSAGE BUFFER OVERFLOWS X, CHECK MESSAGE PROTOCOL!
  Abhilfe schaffte da nur zeitintensives, fortwährendes probieren und herumpröbeln...

Ausführen eines BIT

- Ebenfalls mühsam gestaltete sich die Ausführung dieses Built-In-Test’s (BIT), da ich es nie fertigbrachte, die Resultate auf dem Bildschirm darzustellen. Zudem kommt noch, dass der Empfänger bei jeder Ausführung eines BIT’s resetet wird und somit wieder neu initialisiert werden muss.

3.7.2 Behebung

Behebung

Einstellen des Protokolles

- Die Einstellung des gewünschten Protokolles muss (wie bereits erwähnt) hardwaremässig mit dem sich auf der GPS-Adapter-Karte befindlichen Schalter S1 (siehe *3.5.1*) vorgenommen werden. Dies funktioniert auch einwandfrei, sofern die Bezeichnungen auf dem Schalter (S1) mit denen in den mitgelieferten Unterlagen übereinstimmen ...

- Bei der von mir verwendeten GPS-Adapter-Karte war dies leider nicht der Fall, was mich einige Zeit kostete. Damit dieser "Fehler" nicht wieder passiert, habe ich im folgenden die benötigten Einstellungen für den Empfang eines NMEA-0183-Protokolles im Bezug auf die Unterlagen noch einmal dargestellt.

Die Bezeichnungen des Schalters S1 waren vertauscht ...

Einstellungen für den Empfang eines NMEA-0183-Protokolles

- **RS232-Parameter:**
  [4800 bps] [8 Daten-Bits] [keine Parität (N)] [1 Stop-Bit]
- **Schalterstellung S1:**
  S1/1 ≡ Pin J1-7 ⇒ muss auf Pos. "CLOSED" stehen (LOW aktiv, = 0V)
  S1/2 ≡ Pin J1-8 ⇒ kann auf Pos. "CLOSED" oder "OPEN" stehen
- wobei für S1/2 gilt:
  "CLOSED" = Datenausgabe vom ROM (LOW, = 0V)
  "OPEN" = Datenausgabe vom SRAM oder EEPROM (HIGH, = 5V)

Diese Einstellungen können auch durch Messen der Spannung über dem entsprechenden Schalter überprüft werden.

- Die Geschichte mit dem Schalter erklärt auch, warum der Built-In-Test (BIT) nie richtig funktionierte. Mit den oben angegebenen Einstellungen machte ich mich sofort daran, meine Vermutung bestätigen zu lassen und erhielt durch Betätigung der Taste F8 die folgende Bildschirm-Ausgabe:

Ausführen eines BIT

```
        ZODIAC BUILT-IN TEST RESULTS

FUNCTIONAL TEST                        FAILURES

READ-ONLY MEMORY (ROM)                   0000
RANDOM ACCESS MEMORY (RAM)               0000
NON-VOLATILE STORAGE (EEPROM)            0000
DUAL PORT RAM (DPRAM)                    0000
DIGITAL SIGNAL PROCESSOR (DSP)           0000
REAL-TIME CLOCK (RTC)                    0000
SERIAL PORT 1                               0
SERIAL PORT 2                               0

SERIAL PORT 1 RECEIVE COUNT                56
SERIAL PORT 2 RECEIVE COUNT                 0

          SOFTWARE VERSION 1.80
```

Mit den richtigen Einstellungen funktioniert auch das !

Die Nullen bedeuten, dass **kein** *Fehler vorliegt.*

Wer sich einen schnellen Überblick über die technischen Daten und möglichen Konfigurations-Einstellungen den Zodiac GPS-Empfängers verschaffen möchte, konsultiert am besten das spezifische *Zodiac GPS Receiver Data Sheet*, welches sich im *Anhang* zum Abschnitt *3.3* [blau] befindet. Dort sind auch die oben beschriebenen Einstellungen noch einmal ausführlich erklärt und tabellarisch dargestellt.

3.8 Anhang zu ...

3.8.1 LABMON Operation Guide for the “Jupiter” GPS Receiver [rosa]

3.8.2 Development Kit Setup and Operation Guide for the “Jupiter” GPS Receiver [lachs]

3.8.3 Zodiac GPS Receiver Family Designer’s Guide [blau]

3.8.4 Zodiac Seriel Data Interface Specification [gelb]

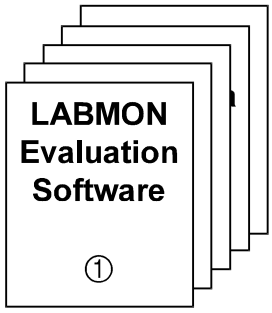
3.8.5 VISYS® GPS ADAPTOR [grün]

4. Konzept

Vor der Ausarbeitung eines Konzeptes befasste ich mich mit Literatur, die ähnliche Problemstellungen beschreiben. Zudem erhielt ich durch die Aufarbeitung der Manuals bereits einen ersten Überblick über den Umfang der auf mich zukommenden Arbeit, was ich im Zeitplan entsprechend berücksichtigte.

4.1 Handbuch zu Jupiter-GPS-Receiver

Um den riesigen Umfang der technischen Beschreibungen in eine geignetere Form zu bringen, wollte ich alle 4 Manuals in einem Register unterbringen.

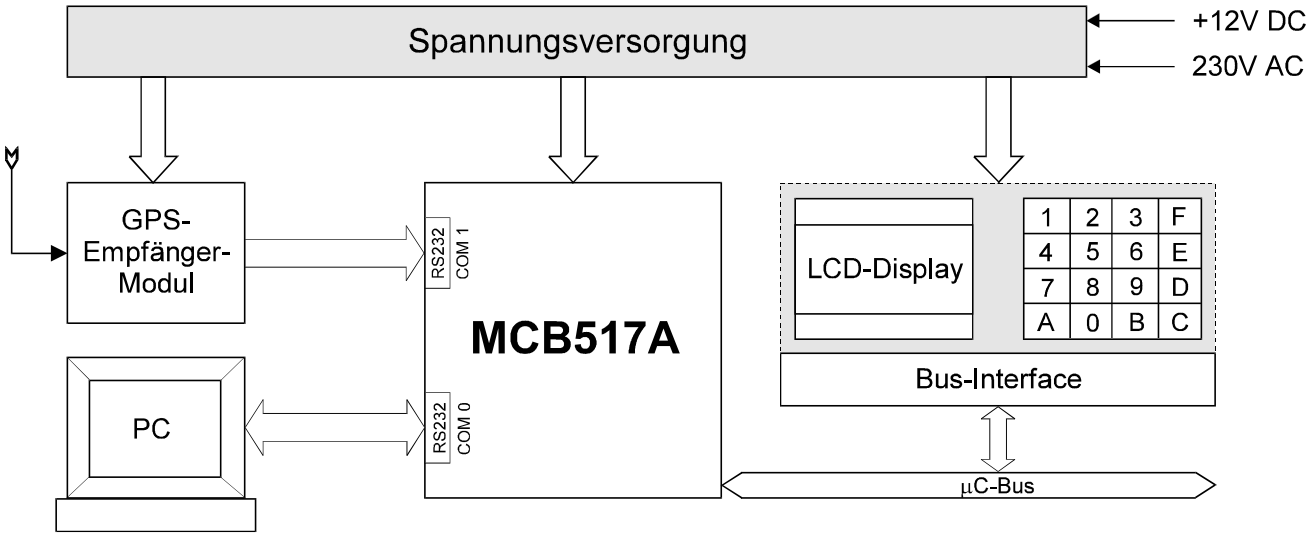


Die dadurch fehlende Unterteilung in die 4 einzelnen Kapitel musste ich mit einem sinnvollen Layout wieder wettmachen. Dies erreichte ich mit einem zweispaltigen Seitenaufbau, wobei die sich in der rechten Spalte befindlichen Untertitel dem schnellen Auffinden von Stichwörtern beim durchblättern dienen.

Damit dieses Kapitel als eigenständiges Dokument verwendet werden kann, müssen die notwendigen Zusatz-Blätter aus dem Anhang der Originaldokumentation ebenfalls beigelegt werden.

4.2 Blockschaltbild

Das folgende Blockschaltbild verdeutlicht das Zusammenspiel der verschiedenen Komponenten und gab mir erste Anhaltspunkte für die Lösung:



Ein- / Ausgabe-Geräte extern	Verarbeitung SOFTWARE	Ein- / Ausgabe-Einheit intern HARDWARE

Gemäss dem Blockschaltbild kann eine Unterteilung in drei Bereiche vorgenommen werden:

HARDWARE

- Die grau hinterlegten Komponenten werden selbst aufgebaut und bilden den Hardware-Teil.

HARDWARE

SOFTWARE

- Der Mikrocontroller bildet die zentrale Einheit, der die Daten verarbeitet und die Kommunikation steuert.

SOFTWARE

Ein- / Ausgabe-Geräte

- Das GPS-Empfänger-Modul kann sowohl Daten senden als auch empfangen, wobei in dieser Arbeit nur Daten über die serielle Schnittstelle COM1 empfangen werden sollen. Der PC wird zum herunterladen des Intel-Hex-Files und zum Starten des Programmes benötigt.

Ein- / Ausgabe-Geräte

- Zusätzlich kann mit einem geeigneten Terminalprogramm das vom GPS-Empfänger ausgegebene Protokoll visualisiert werden. Dazu ist das über die erwähnte Schnittstelle des Prototypen-Boards empfangene Protokoll direkt an den zweiten COM-Port 0 durchzuschlaufen und auszugeben. Als Terminalprogramm kann in diesem Modus auch die in *Kapitel 3* beschriebene LABMON-Software dienen, womit die elementarsten Funktionen wie z.B. die Initialisierung des Empfängers oder das Einstellen der aktuellen Position / Zeit direkt und ohne Umstecken von Verbindungskabeln realisiert werden kann.

Terminalprogramm

LABMON-Evaluation-Software

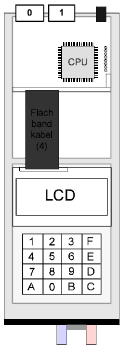
Das EVA-Prinzip bestätigt sich auch hier. Die Ein- und Ausgabe erfolgt durch die Ein- / Ausgabe-Geräte, die Verarbeitung wird softwaremässig durch den Mikrocontroller vorgenommen. So betrachtet ist klar ersichtlich, dass die Hardware nur “Mittel zum Zweck” ist, jedoch einen grossen Teil der Zeit in Anspruch nehmen wird.

“Mittel zum Zweck"

4.3 Hardware

4.3.1 Aufbau und Realisierung

Mein Ziel beim Aufbau war, die einzelnen Komponenten so anzuorden, dass sie durch ein paar Handgriffe wieder voneinder getrennt werden können. Dies erleichtert die Nachbearbeitung sowie die Austauschbarkeit von Bauteilen und ist in Form eines Prototyps sinvoll, da noch keine Anforderungen im Bezug auf Platzbedarf und Industrietauglichkeit erfüllt werden müssen.
Zudem wissend, dass auf dem Markt kompakte GPS-Empfänger mit Anzeige und serieller Schnittstelle für einige hundert Schweizer Franken erhältich sind, hätte dieses Vorhaben im Keim erstickt.



Somit konzipierte ich eine Einrichtung, welche in die folgenden “Teile” zerlegt werden kann:

4.3.1.1 Grund- und Frontplatte

Grund- und Frontplatte

Der Hardware-Aufbau erfolgt auf einer PVC-Grundplatte, an welche stirnseitig eine Bedieneinheit in Form eines Alu-Winkels montiert ist.

Bedieneinheit

An dieser befinden sich ebenfalls die Anschlüsse für die Spannungsversorgung und die Sicherungen.

4.3.1.2 Laborkarten 1 und 2

Sämtliche elektronischen Bauteile werden auf einer Laborkarte im Europa-Format angeordnet. Da die beiden Transformatoren sowie das Display bzw. die Tastatur relatv viel Platz beanspruchen, werden auf Wunsch des Dozenten 2 Printplatten übereinander angeordet.

Laborkarte 1
Spannungsversorgung

Die gesamte Spannungsversorgung befindet sich auf der unteren Karte und wird mittels einer geeigneten Steckverbindung (Flachbandkabel) mit der Bedieneinheit verbunden.

Laborkarte 2
Logikschaltung und interne Ein- / Ausgabe-Einheit

Das Display wird mitsamt der Tastatur und der Logikschaltung auf der oberen Printplatte montiert. Auch hier ist eine sinvolle Verbindung zur darunterliegenden Speisung erforderlich. Zudem wird von dieser Karte aus die Verbindung zum Prototypen-Board realisiert.

4.3.1.3 Prototypen-Board MCB-517AC

Prototypen-Board

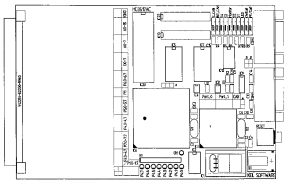
Da auf das Prototypen-Board hardwaremässig nicht eingegriffen werden soll, werden die Verbindungen mit Flachbandkabeln und geeigneten Printsteckern realisiert. Das Board wird mittels Führungsschienen auf der Grundplatte befestigt, um den schnellen Austausch zu gewährleisten.

Stand-Alone-System

An dieser Stelle noch eine Bemerkung, die ich aufgrund der Gedanken zu einem Stand-Alone-System festhalten möchte:

EEPROM

Im Falle dieser Anforderung wäre der Einsatz eines selbst gebrannten EEPROMS gegeben und natürlich auch sinvoll, da man zur Inbetriebnahme der Einrichtung nicht jedesmal den Laptop benötigen würde.



Andererseits gehört das verwendete Prototypen-Board nach wie vor der Schule und kann in meinem Fall direkt und ohne weitere Eingriffe für andere Zwecke verwendet werden, was beim Einsatz eines EEPROMS ohne Entfernung desselben nicht mehr möglich ist.

4.4 Software

In der Konzeptphase konnte ich zu der Software noch nicht viel beisteuern, da das Schreiben eines Programmes immer eine individuelle Sache ist und ich mich zuerst in die Diplom-Arbeit von D. Läubli [9] vertiefte, da dieser eine ähnliche Problemstellung bereits mit Erfolg lösen konnte. Ich merkte jedoch bald, dass ich eine ganz andere Vorstellung davon hatte, wie mein Programm aussehen sollte - doch nicht nur das: Aufgrund der gänzlich anderen Hardware-Voraussetzungen wurde mir klar, dass mein Programm vor allem einem gerecht werden muss:

```
LCD_TXT:    CLR     A
            MOVC    A,@A+DPTR
            JZ      LCD_ENDE
            CALL    LCD_WRITE
            INC     DPL
            JMP     LCD_TXT
LCD_ENDE:   RET
TXT_ENDE:   CALL    LCD_OFF_TXT
            JMP     LCD_ENDE
```

- Die Software sollte mit Hilfe von Subroutinen möglichst modular gestaltet und gut leserlich sein, damit für weitere Experimente mit der Hardware-Plattform die wesentlichsten Funktionen übernommen werden können.
  Klar war auch, dass ich zu diesem Zweck mit Interrupts arbeiten würde.

Die Software sollte möglichst modular realisiert werden.

Anhand des Programmes von D. Läubli erahnte ich in etwa den Umfang, den mein Programm haben würde. Dies erhärtete die oben genannte Forderung zwar, hielt mich jedoch nicht davon ab, die benötigten Subroutinen zu übernehmen. Nach einigem hin- und her entschied ich mich, beide Teile zu berücksichtigen und kam zu folgendem Ansatz:

- Von der Hardware abhängige (Ansteuerung mittels Adress-/Datenbus!) Funktionen wie etwa die Kommunikation mit dem LCD-Controller sollen nach obiger Vorgabe möglichst in einzelne Subroutinen “verpackt” werden
- Die Routine zur Filterung und Abspeicherung der NMEA-Daten sowie die in der Hochsprache C implementierten Berechnungsroutinen für Sommer-/Winterzeit und die Schweizer-Landeskoordinaten werden übernommen
- Ebenso wird die Idee des Display-Abbildes und die Speicherung von Daten in einem explizit freizugebenden Speicherbereich (XRAM) in leicht veränderter Form ebenfalls weiterverwendet, da sich vor allem die C-Berechnungsroutinen auf diese Vorgaben stützen

4.4.1 Struktur und Programmablauf

Aufgrund dieser Überlegungen nahm ich mir vor, mein Programm in etwa der folgenden Form zu realisieren:

Struktur

Mit Struktur meine ich den bildlichen Aufbau eines Programmes, nach welchem sich soweit möglich auch die Anordnung des Source-Codes auf dem Papier gestaltet. Dies erleichtert die Lesbarkeit und trägt zum besseren Verständnis des Programmes bei.
In der Regel ist damit die Reihenfolge
[Programm-Kopf] - [Definitionen] - [Initialisierung] - [Hauptprogramm] - [Subroutinen] - [Interrupt-Routinen] - [Ende] gemeint.

“Struktur” eines Programmes

Programmablauf

Dieser beschreibt den zeitlichen Ablauf, also die Reihenfolge der Tätigkeiten, welche das Programm ausführt und wie es auf eintretende Ereignisse reagiert. Die hierzu besser geeignete graphische Darstellung in Form von Flussdiagrammen oder Struktogrammen wird erst im Kapitel *Software* anhand des definitiven Programmes vorgenommen.

"verbale" Beschreibung des Programmablaufs

An dieser Stelle eine grobe verbale Beschreibung des Programmablaufes, wie ich ihn mir zu diesem Zeitpunkt vorstellen konnte:

- BEGIN
- Empfangen des NMEA-Protokolles
- Filterung der Daten bis zum Erkennen des Datensatzes $GPRMC
- Einlesen und abspeichern der benötigten Daten (Rohdaten)
- Wenn Daten gültig, die gespeicherten Rohdaten entsprechend aufbereiten (Anzeige-Form, Berechnungen) und erneut speichern
- Die gewünschten Daten zur Anzeige im Display-Abbild-Bereich bereitstellen
- Daten auf LCD-Display ausgeben
- END

4.4.2 Einschränkung

Einschränkung

Was ich aus der Übersetzung der englischen Manuals bereits wusste, war, dass ich den von D. Läubli benutzten NMEA-Datensatz mit dem Identifier $PGRMZ nicht zur Verfügung habe, da dieser ein GARMIN-spezifischer Datensatz ist. Dieser Datansatz enthält die Höhenangabe in feet.
Natürlich liefert das GPS-Empfänger-Modul ebenfalls Höhenangaben, jedoch in einem anderen Datensatz und zudem bezogen auf die Meereshöhe ALT_MSL (siehe Kapitel *3, GGA Message*). Die geodätische Höhe müsste nun aus der Differenz des WGS-84-Referenzellipsoids und des MSL-Geoids berechnet werden. Dies wird jedoch aus zeitlichen Gründen nicht realisiert.

Anstelle der Höhe wird das aktuelle Datum auf dem Display angezeigt.

4.4.3 Funktion und Anzeige

Funktionen des Programmes

Funktionen und Anzeige

Die folgenden Punkte beschreiben wie sich das Programm verhalten soll:

- Das Programm soll nach dem Start ohne weitere Eingriffe die Längen- und Breitengrade, die UTC-Zeit und das Datum auf die Anzeige übertragen.
- Wenn keine Verbindung zum GPS-Empfänger vorhanden oder die Daten ungültig sind, wird dies mit einer entsprechenden Meldung angezeigt.

Da mir 16 Tasten zur Verfügung stehen, kann ich diesen beliebige Funktionen zuweisen.

Tasten-Funktionen

- Nicht implementierte Tasten zeigen dies bei Betätigung mit einer Fehlermeldung an.

Fehlermeldungen

- Da der GPS-Empfänger dauernd Daten an die Schnittstelle des Prototypen-Boards sendet und nur durch Trennen der Stromversorgung ausgeschaltet werden kann, muss mit einer Taste die Schnittstelle gesperrt bzw. die Annahme des Interrupts verweigert werden können.

Sperren des Interrupts COM1

- Die verschiedenen Anzeigeformen der Positionsdaten sind per Tastendruck abrufbar und werden so lange angezeigt, bis die nächste Taste betätigt wird.

Anzeigen der Informationen

Die Visualisierung des NMEA-Protokolls bedingt nach dem Starten des Programmes mittels MON51 oder TS51 das öffnen eines Terminalprogrammes oder der LABMON-Evaluation-Software am PC.

- Diese Visualisierungsvorgang, also das “Durchschlaufen” der Daten von COM1 zu COM0, soll ebenfalls per Tastendruck gestartet werden können.

Visualisierung NMEA-Protokoll

Anzeige

Die aufbereiteten Daten müssen entsprechend der betätigten Taste auf dem Display dargestellt werden. Da ich die bereits angesprochene Idee des “Display-Abbildes” (Ablage der anzuzeigenden Daten in einem genau definierten Speicherbereich) übernehme, ist auch die Darstellungsform definiert:

Anzeige 1:
Geographischen Koordinaten, Datum und UTC-Zeit

Taste 1

```
L/B-KOORDINATEN:
47°23.17N  08°10.02E
DATUM:     28:07:98
TIME UTC:  15:52:03
```

Anzeige 2 und 3:
Schweizer-Landeskoordinaten, Datum und MESZ / MEZ

Taste 2 und 3

```
X/Y-KOORDINATEN:
X=716107m  Y=233458m
DATUM:     28:07:98
XZEIT CH:  17:52:03
```

Bei Betätigung der Taste 2 soll die MESZ, bei Taste 3 die MEZ dargestellt werden. An der Stelle von X steht bei der Zeitangabe ein **S** für Sommer- (UTC +2 Stunden) und ein **W** für die Winterzeit (UTC + 1 Stunde).

Abkürzungen siehe Kapitel3, Abschnitt 3.6

4.5 Entwicklungsvorgang

Die Realisierung meiner Diplomarbeit erfolgte aufgrund des Konzeptes in einzelnen Schritten, wobei ich mit dem Studium der englischen Manuals sowie anderer Literatur begann, danach die erforderliche Hardware erstellte und zuletzt die Verknüpfung mittels der Software realisierte.

4.5.1 Entwicklungsumgebung

uVision-51.lnk

dScope-51.lnk

Diese bestand zu Beginn aus einer Experimentierplatte, der Prototypenbaugruppe MCB517A sowie der zugehörigen Entwicklungssoftware und (nicht zu vergessen) aus einem PC.
Der auf der “KEIL Demonstration-Tools CD-ROM” enthaltene DScope-51 - Debugger (Windows-DEMO-Version) erlaubt zu diesem Zweck im Debugg-Modus den Eingriff auf die SFR sowie die Interrupt-Quellen und kann so das Verhalten des Programmes simulieren. Die komfortable Benutzeroberfläche ermöglicht zudem die gleichzeitige Darstellung sämtlicher Register oder einzelner Bits während der Laufzeit des Programmes. Ebenso kann das Verhalten einzelner Programm-Module ohne vorhandene Hardware relativ bequem ausgetestet werden.
Somit war ich in der Lage, Funktions- und Ablauffehler des Programmes frühzeitig und zum Teil unabhängig von der Hardware zu erkennen und zu verbessern.

Der Einsatz des Windows-DScope-51-Debuggers ermöglichte mir, das Verhalten gewisser Programm-Module bereits im voraus zu Testen.

Mit dem Testaufbau der Adressdecodierung auf der Experimentierplatte und einer “provisorischen” Verbindung zum Prototypen-Board hatte ich die Möglichkeit, den Einfluss der Hardware auf die Software zu betrachten und auszuwerten. Zu diesem Zweck musste ich mir jedoch ab einem gewissen Software-Stand die Vollversion der Entwicklungssoftware besorgen, welche mir in der DOS-Version von der Firma W. Fehlmann AG zur Verfügung gestellt wurde. Damit konnte ich meine Programme unter realen Bedingungen im Hardware-Debug-Modus Zeile für Zeile testen und analysieren.
Dies jedoch erst, nachdem ich einige Zeit damit verbrachte hatte, die Unterschiede zur Windows-Version “kennenzulernen” und ich mir Batch-Dateien für die Kompilierung und das Linken der Source-Files erstellt hatte.

Spätestens mit der Einbindung der C-Routinen versagte die DEMO-Version.

TS 51.pif

MON 51.pif

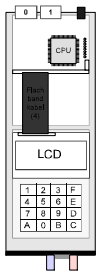
Den ungefähren Entwicklungsablauf möchte ich hier wiedergeben:

- Erhalten der Aufgabenstellung — *Woche 17 / 98*
  Vertiefung in das Thema mittels Literatur — *ca. 15 Std.*
  Erstellen Layout und Gliederung der Dokumentation
  Erste Gedanken zum Konzept
- Detaillierung der Aufgabenstellung — *Woche 18-19 / 98*
  Studium und Übersetzung der Manuals — *ca. 22 Std.*
  Sammeln von Informationen

• Studium und Übersetzung der Manuals Einarbeitung in die Entwicklungsumgebung (LABMON Software und GPS-Empfänger-Modul)	*Woche 20-22 / 98* *ca. 80 Std.*
• Weitere Detaillierung der Aufgabenstellung Korrektur und Abschluss der Übersetzugen Fertigstellen des Konzeptes Entwicklung HARDWARE Definitiver Aufbau und Test Spannungsversorgung (Karte 1) Zeichnen von Schemas	*Woche 23-24 / 98* *ca. 60 Std.*
• Testaufbau der Logikschaltung (Adressdecodierung) Versuche und Fehlschläge Definitiver Aufbau und Test Adressdecodierung (Karte2) Zeichnen von Schemas Schreiben von Subroutinen zur Ansteuerung von Display und Tastatur Erste Gedanken zum Software-Konzept	*Woche 25-27 / 98* *ca. 80 Std.*
• Schreiben der Subroutinen zur Ausgabe von Text am Display Probleme mit dem GPS-Empfänger Modul und deren Lösungen Überarbeitung der LCD-Subroutinen Überarbeitung des SOFTWARE-Konzeptes	*Woche 28 / 98* *ca. 50 Std.*
• Subroutine zur Filterung der NMEA-Daten Einlesen und abspeichern des benötigten Datensatzes Anzeige der Rohdaten in L/B-Koordinaten mit Zeit UTC	*Woche 29 / 98* *ca. 20 Std.*
• Umstellen der Entwicklungssoftware von WIN auf DOS Modifizierung und Einbinden der C-Routinen Beheben der Randprobleme betr. C-Compiler sowie Linker	*Woche 30 / 98* *ca. 40 Std.*
• Schreiben der Routine zum NMEA-Protokoll-Transfer “Fertigstellen” der Software (bis auf C-Routinen) Nachführen der Dokumentation, zeichnen von Graphiken	*Woche 31 / 98* *ca. 60 Std.*
• Korrekturen und Verbesserungen Kopieren und zusammenstellen der Unterlagen Schreiben der Zusammenfassung Erstellen der Unterlagen für die Taxation	*Woche 32 / 98* *ca. 60 Std.* **Total** **ca. 460 Std.**
• Abgabe der Arbeit	*Woche* **33**

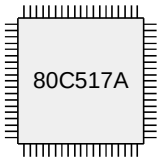
Das Schreiben der Dokumentation erfolgte wann immer möglich parallel zu den einzelnen Arbeitsschritten und ist in den angegeben Stunden enthalten.

5. Hardware



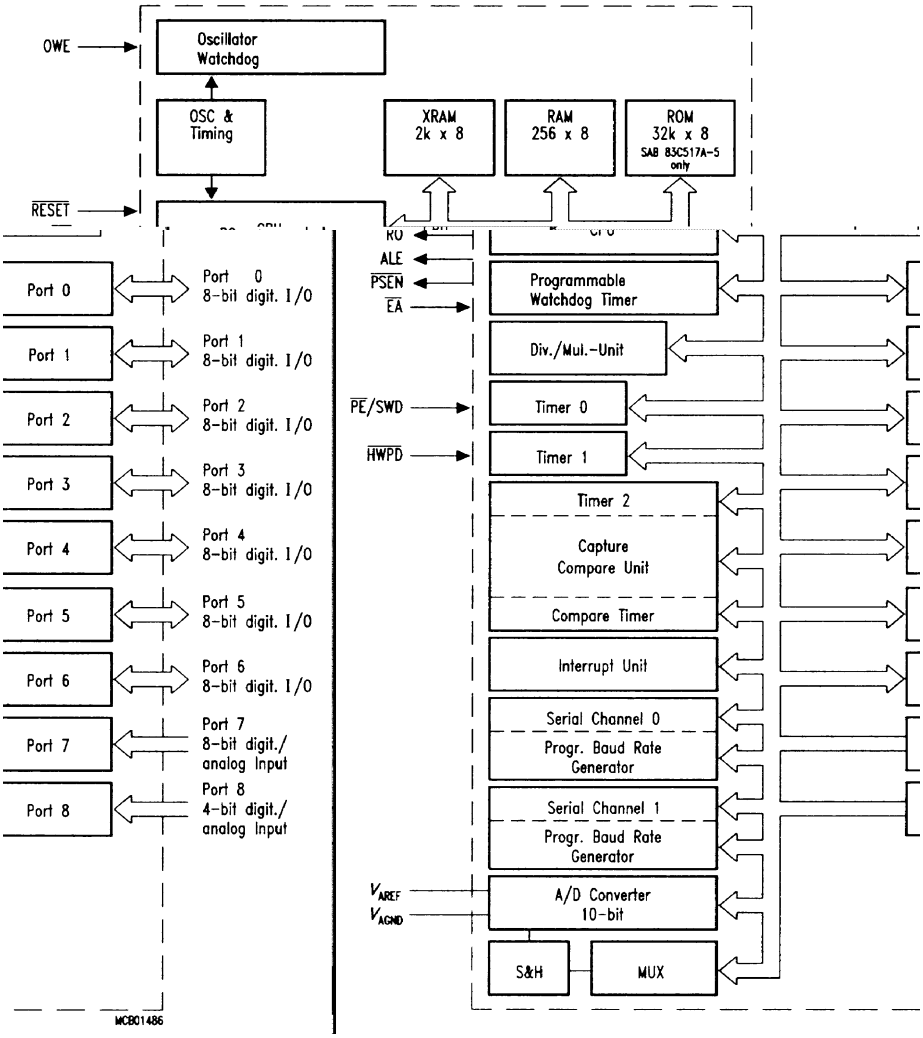
In diesem Kapitel werden die verwendeten HW-Komponenten sowie ihre wichtigsten Eigenschaften kurz vorgestellt und die Ansteuerung der Peripherie-Geräte mit Hilfe der Adressdecodierschaltung erklärt.

5.1 Mikrocontroller 80C517A



Die folgende Kurz-Beschreibung des 80C517A-Mikrocontrollers beschränkt sich auf die wesentlichen Informationen, welche ich während der Entwicklungszeit oft benötigte.
Für weiterführende Informationen sei an dieser Stelle auf das dem *Anhang* beigelegte Datenblatt oder auf das Kapitel *Literaturverzeichnis* verwiesen.

5.1.1 Blockschaltbild



5.1-1 Blockschaltbild des SAB 80C517A

Anhand des Blockschaltbildes wird verständlich, warum der 80C517A in zahlreichen Anwendungen eingesetzt wird.
Die Vielzahl der auf dem Chip integrierten Funktionen ermöglicht den Einsatz ohne grossen Aufwand an externer Hardware.

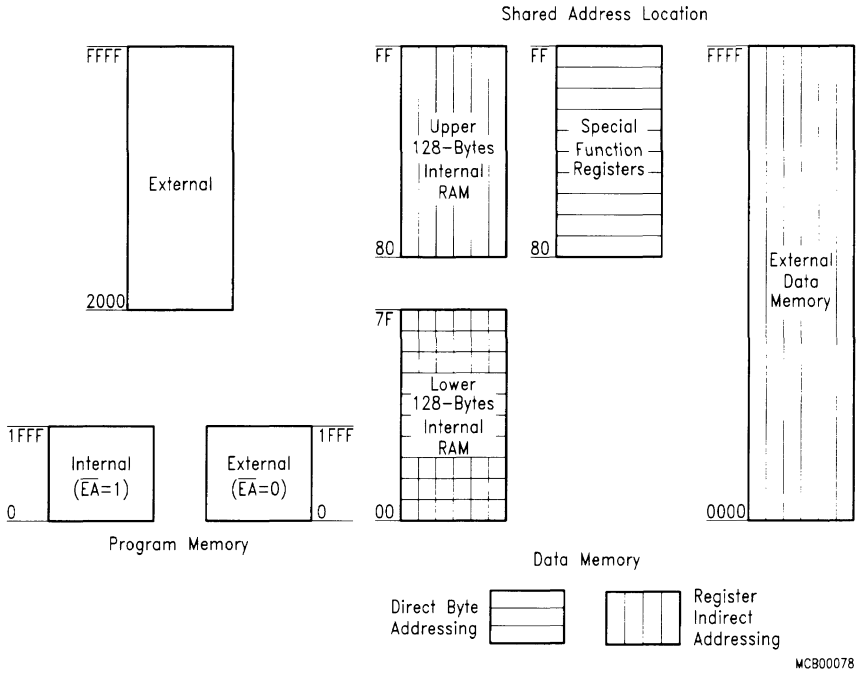
5.1.2 Speicherkonzept

5.1.2.1 Einleitung

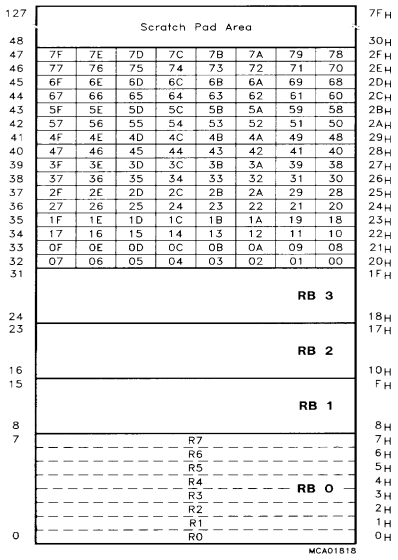
Grundsätzlich ist die Speicherarchitektur der 8051-Familie eine “Mischung” aus Harvard- und Von Neumann-Architektur. Der 80C517A kann neben seinem internen Daten- und Programmspeicher zusätzlich mit externem Daten- und Programmspeicher von je 64 kByte ausgerüstet werden (Harvard).
Der externe Daten- und Programmspeicher kann auch eine physikalische Einheit bilden, ist dann allerdings maximal 64 kByte gross (Von Neumann). Diese Architektur ist für Entwicklungssysteme wie das MCB517A-Prototypen-Board unumgänglich, da nur so neu erstellte Programme im laufenden Betrieb editiert werden können.

Alle MCS-51 Bausteine haben einen getrennten Adressbereich für Programm- und Datenspeicher von jeweils 64 kByte Grösse.

5.1.2.2 Speicherorganisation



5.1-2 Memory Map



5.1-3 Lower RAM

Interner Speicher

Auf dem Prozessor-Chip integriert ist ein 256 Byte grosser Datenspeicher, ein 128 Byte grosser Speicherbereich zur Steuerung der internen Mikrocontroller-Funktionen (Special Function Register SFR) und ein Programmspeicher von 8 kByte, der aber nur genützt werden kann, wenn ihn der Hersteller mit Inhalt gefüllt hat.

Programmspeicher

Wesentlich interessanter ist der nur 256 Byte grosse, aber sehr effizient einsetzbare Datenspeicher, welcher in drei Bereiche unterteilt werden kann:

Datenspeicher

Adressierugsart:

- Lower-RAM (Adressbereich 00H-7FH, 128 Byte) — `direkt oder indirekt`
- Upper-RAM (Adressbereich 80H-FFH, 128 Byte) — `indirekt  MOV A,@R1`
- SFR-RAM (Adressbereich 80H-FFH, 128 Byte) — `direkt  MOV A,R1`

Da Upper-RAM und SFR-RAM sich den gleichen Adressbereich teilen, erfolgt der Zugriff durch unterschiedliche Adressierungsarten.

Lower-RAM

Weiter kann der Lower-RAM-Bereich wiederum in drei Adressbereiche unterteilt werden. Er enthält

- 4 Registerbänke mit jeweils acht 8-Bit-Registern (00H-1FH)
- 16 bitadressierbare Bytes (20H-2FH) und
- 80 frei verfügbare Bytes (30H-7FH)

Upper-RAM

Das Upper-RAM kann vollständig zur Zwischenspeicherung von Daten genutzt werden.

SFR-RAM

Adressenmässig parallel zum Upper-RAM liegt ein ebenfalls 128 Byte grosser Speicherbereich (SFR-RAM), in dem sich die 81 Special Function Register des Controllers befinden; unter anderem das Programmstatuswort (PSW), der Akkumulator (ACC), der Stack Pointer (SP) und sämtliche Register, die zur Steuerung und Kontrolle der Peripherie dienen.
Als Programmierer ist es von Vorteil zu wissen , dass alle SFR-RAM-Adressen, die auf 0 oder 8 enden, Bit- **und** Byte-adressierbar sind.

Externer Speicher

externer Speicher

Wie bereits erwähnt, ist in Harvard-Architektur ein externer Speicherbereich von je 64 kByte Daten- sowie Programmspeicher adressierbar.

5.1.2.3 Zugriff auf den externen Speicherbereich

externer Programmspeicher

- Zugriffe auf den externen Programmspeicher (MOVC) verwenden grundsätzlich 16-Bit-Adressen und finden immer dann statt, wenn das Signal $\overline{\text{EA}}$# aktiviert (=0) ist, oder die Adresse oberhalb des verfügbaren internen Code-Bereichs von 0-1FFFH (8 kB, durch Hersteller programmierbar) liegt. Zudem wird das Steuersignal PSEN# (Program Store Enable) als Lesesignal aktiviert.

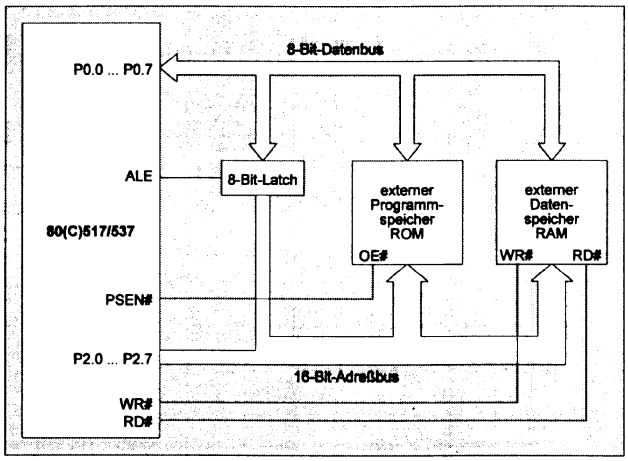
`MOVC  A,@A+DPTR`

externer Datenspeicher

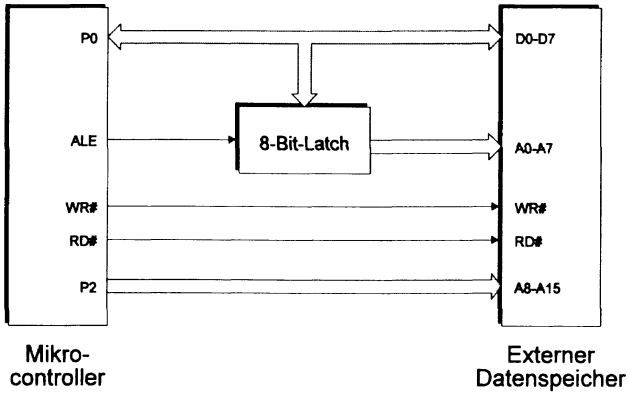
- Zugriffe auf den externen Datenspeicher sind nur mit dem MOVX-Befehl möglich und verwenden 8- oder 16-Bit-Adressen, wobei mit den Steuerleitungen $\overline{\text{RD}}$# (Read) bzw. $\overline{\text{WR}}$# (Write) zwischen EIN- oder AUSgabe unterschieden wird.

`MOVX  @DPTR,A`

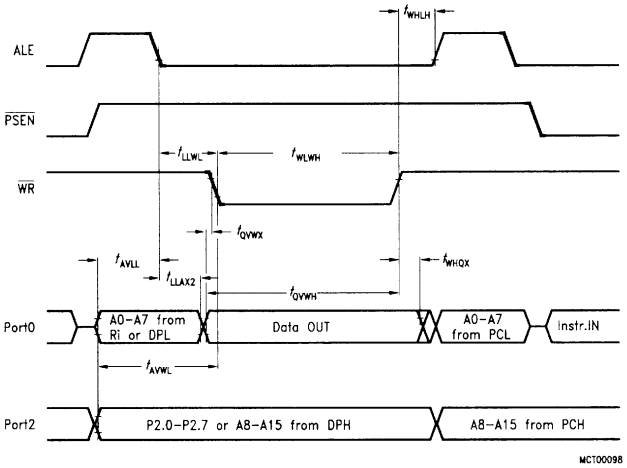
Um externen Speicher zu betreiben, werden nebst dem Daten- und Adressbus (Port 0 und 2) die 4 Steuersignale PSEN#, RD#, WR# und ALE benötigt.



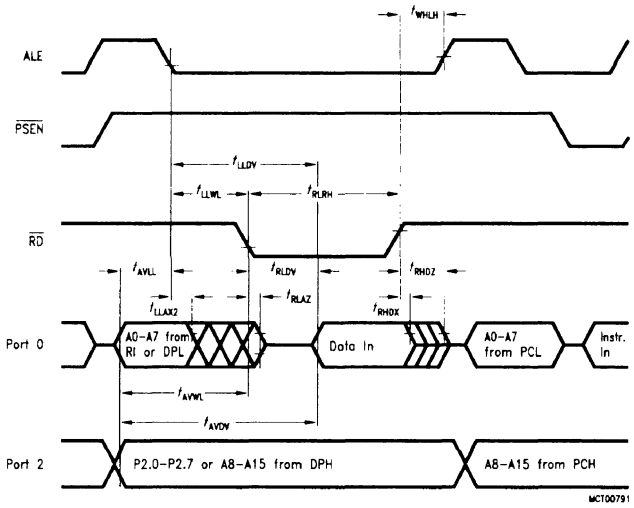
5.1-4 Steuersignale zur Speichererweiterung



5.1-5 Anschluss von externem Datenspeicher



5.1-6 Data Memory Read Cycle



5.1-7 Data Memory Write Cycle

In beiden Fällen wird das *Low Order Byte* (LOB) der 16-Bit-Adresse über Port 0 und das High Order Byte (HOB) über Port 2 ausgegeben. Die Daten werden immer über Port 0 eingelesen bzw. ausgegeben. Um dies zu ermöglichen, muss eine Auftrennung zwischen Daten- und Adressbus vorgenommen werden. Hierzu stellt der Mikrocontroller das Signal ALE (Adress Latch Enable) zur Verfügung, welches das ausgegebene LOB der Adresse in einem Latch zwischenspeichert, damit es am Speicherbaustein konstant anliegt.

5.1.3 Serielle Schnittstellen (COM0 und COM1)

5.1.3.1 Einleitung

Der 80C517A verfügt über zwei serielle RS232-Schnittstellen (USART und UART), welche unabhängige Baudraten-Generatoren besitzen. Da sie immer nur ein Bit pro Schritt transportieren können, sind Baudrate (bit/sec) und Übertragungsrate identisch. Die beiden seriellen Leitungen RxD bzw. TxD ermöglichen gleichzeitiges Senden und Empfangen, die beiden Schnittstellen sind somit VOLL-DUPLEX-Fähig.

5.1.3.2 Serielle Schnittstelle Nr. 0

Diese kann in vier unterschiedlichen Modi betrieben werden:

Einstellen der Baudrate:

- Mode 0, synchron, 8 Bit, feste Baudrate — *fest auf $f_{osc}/12$*
- Mode 1, asynchron, 10 Bit, variable Baudrate — *Timer 1 Overflow und SMOD*
- Mode 2, asynchron, 11 Bit, feste Baudrate — *Kontroll-Bit SMOD (PCON.7)*
- Mode 3, asynchron, 11 Bit, variable Baudrate — *Timer 1 Overflow und SMOD*

Zur Steuerung dient das Special Function Register (SFR) S0CON (98H)

5.1.3.3 Serielle Schnittstelle Nr. 1

Diese kann in zwei unterschiedlichen Modi betrieben werden:

Einstellen der Baudrate:

- Mode A, asynchron, 11 Bit, variable Baudrate
  (identisch mit Mode 3 der seriellen Schnittstelle Nr. 0)
- Mode 1, asynchron, 10 Bit, variable Baudrate
  (identisch mit Mode 1 der seriellen Schnittstelle Nr. 0)

Das Einstellen der Baudrate ist nur über einen besonderen internen Baudraten-Generator möglich.

Zur Steuerung dient das Special Function Register (SFR) S1CON (9BH)

Interner Baudraten-Generator

Dieser besteht bis zur Version CA aus einem freilaufenden 8-Bit-Timer, welcher mit $f_{osc}/2$ getaktet wird und im Auto-Reload-Mode arbeitet. Die Baudrate wird durch einen Eintrag im SFR S1REL (9DH) bestimmt.

... bis Version CA
S1REL (9DH)

Ab Serien-Nr. CA steht ein 10-Bit-Timer zur Verfügung, dessen Reload-Wert in die SFR S1RELH (BBH) und S1RELL (9DH) geladen werden muss.

Version CA ...
S1RELH (BBH) u. S1RELL (9DH)

5.1.3.4 Senden und empfangen

Der Sendevorgang wird in beiden Fällen durch beschreiben eines Registers ausgelöst. Der Empfangsvorgang wird initialisiert durch Setzen bzw. Rücksetzen der entsprechenden Bits im jeweiligen Steuer-Register (S0/1CON).

Senden: Beschreiben von
S0BUF (99H) / S1BUF (9CH)
Empfangen:S0CON / S1CON

5.2 Prototypen-Baugruppe MCB-517AC

Durch die Verwendung des Prototypen-Boards ist die Speicheraufteilung- sowie Verwaltung durch die Einstellungen der DIP-Schalter 1-10 auf dem Prototypen-Board definiert.

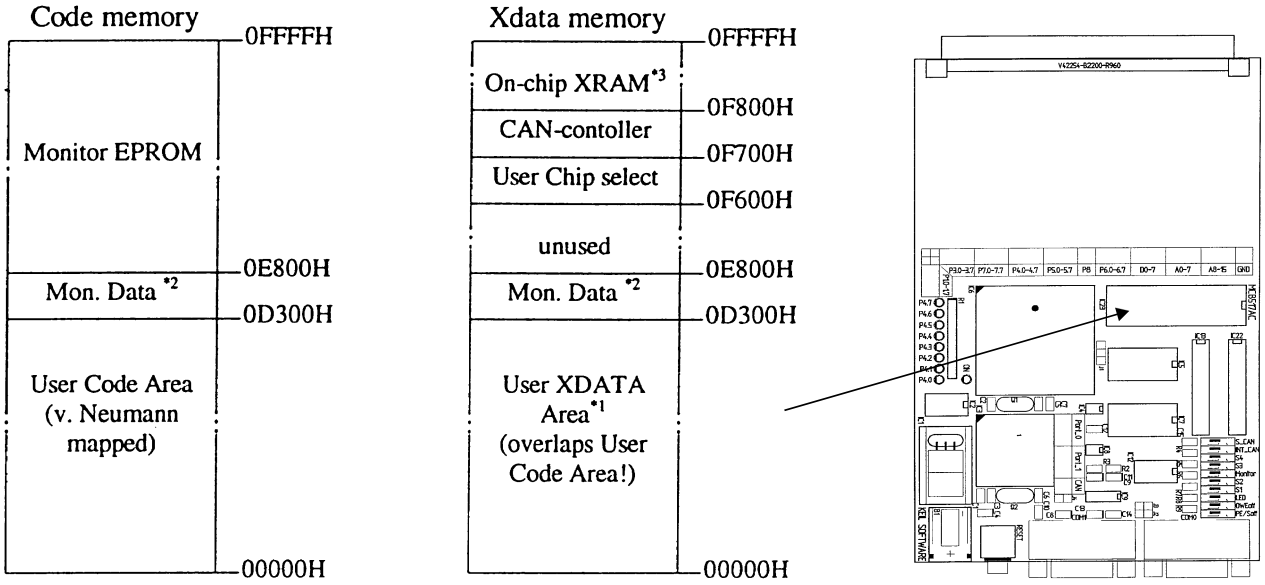
Für meine Arbeit sind die Schalterstellungen wie folgt:

Schalter	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Bez.	S_CAN	INT CAN	S4	S3	Monitor	S2	S1	LED	OWE off	PE/ Soft
ON			X	X	X			X	X	X
OFF	X	X				X	X			

Die Schalter Monitor, S1 und S2 aktivieren in dieser Stellung das Monitor Memory-Mapping des mitgelieferten EPROM's (IC 23).

5.2.1 Monitor Memory-Map

Dadurch ergibt sich die folgende Speicheraufteilung:



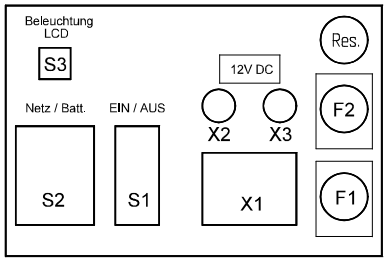
5.2-1 Monitor Memory Map

Da hier der Bereich von 0000H bis D2FF nach Von Neumann verdrahtet ist, wird der Code-Speicher durch einen Schreibvorgang in den Daten-Bereich ebenfalls beeinflusst. Der User-Chip select für die Ansprechung externer Peripherie befindet sich ab Adresse F600H (Xdata). Der 2 kByte grosse On-chip XRAM*3-Bereich kann durch löschen des SFR-Bits XMAP0 im Register SYSCON (B1H) aktiviert werden.

Für weitere Informationen siehe "User's Guide 04.97" MCB517AC im *Anhang*.

5.3 Mechanik

Nachdem das Konzept stand und die benötigten Bauteile und Peripherie-Einheiten bestimmt waren, machte ich mich sogleich an den mechanischen Teil, damit die Hardware so schnell wie möglich aufgebaut werden konnte. Dieser Bestand aus der Konstruktion der Grundplatte und des Alu-Winkels sowie der Anordnung der einzelnen Komponenten und der Bedienungselemente.



Grundplatte und Winkel

Eine detaillierte Zeichnung ist im *Anhang* ersichtlich.

5.4 Spannungsversorgung

Diese ist so aufgebaut, dass als Quelle entweder eine Autobatterie oder das 230V-Netz verwendet werden kann. Die Umschaltung erfolgt mit einem Kippschalter (S2). Folgende Spannungen müssen generiert werden:

- 2 mal 12V DC für das MCB517A-Board und GPS-Empfänger-Modul
  +5V DC für die Logik
  9V für die Hintergrundbeleuchtung des LCD-Displays u. a. (Option)

5.4.1 Beschreibung Schema

Beschreibung Schema

Da man hier genau so gut ein fertiges Modul hätte verwenden können, wurde die Schaltung weitgehend nach den Wünschen des Dozenten ausgelegt und vorgegeben. Um für sämtliche Komponenten genügend Strom liefern zu können, werden 2 Transformatoren (T1=10VA, T2=5VA) verwendet, von denen einer (T2) alleine für die Hintergrundbeleuchtung des LCD-Displays verwendet wird.

12V DC

Autobatterie oder

Die Autobatterie liefert bereits die gewünschte Spannung von 12V DC. Als Verpolschutz wird ein Gleichrichter (V4) verwendet.

Transformator T1: 2 x 12V DC für MCB-Board, GPS-Empfänger und Spannungsregler

Der Transformator T1 hat 2 Sekundärspulen, aus welchen mit 12V AC (eff.) und einer nachgeschalteten Gleichrichterschaltung (Grätz-Brückenschaltung, Glättungskondensator und Entladewiderstand) unter Belastung eine ungesiebte Gleichspannung von ca. 11.8V DC bei einem Strom von ca. 0.4A erzeugt wird. Die eine Wicklung speist das MCB517A-Prototypen-Board, die andere den GPS-Empfänger und den Spannungsregler, welcher mit 2 Kondensatoren beschaltet werden muss (angegebene Werte = *Messwerte*).

9V DC

Transformator T2: 1 x 9V DC für Hintergrundbeleuchtung LCD-Display

Der Transformator T2 liefert bei einer Sekundärspannung von 7V AC (eff.) mit derselben Grätz-Gleichrichterschaltung eine ungesiebte Gleichspannung von knapp 8.5V DC bei einem Strom von 0.3A (angegebene Werte = *Messwerte*).

Logikspannung +5V DC

Spannungsregler LM340T5
1 x 5V DC für Logikspannung

Diese wird mit einem Spannungsregler vom Typ LM340T5 generiert, welcher als handelsüblicher Serienregler gut erhältlich ist und nach dem Emitterfolger-Prinzip (Kollektorschaltung) arbeitet. Die somit vorhandenen 3 Anschlüsse sind in meiner Anwendung in einem TO-220 Gehäuse untergebracht, welches zur Abführung der Verlustleistung an einem passenden Kühlkörper befestigt werden muss.

Damit auch im Batteriebetrieb die Hintergrundbeleuchtung des Displays genutzt werden kann, ist der Plus-Pol (X3) über die 3 Leistungsdioden (V5-V7) direkt auf den Anschluss 1 des Schalters S3 weitergeführt. Letzere dienen als Rückspeisungs-Schutz (häufiges Ein- und Ausschalten der Hintergrundbeleuchtung!) und erzeugen zugleich den restlichen Spannungsabfall (ca.1.6V) um die geforderte Ausgangsspannung von 9V zu erhalten.

5.4.2 Dimensionierungen

Da die Schaltung weitgehend durch den Dozenten vorgegeben war, kamen Erfahrungswerte zum Einsatz.

Stromverbrauch

Was ich zu beachten hatte, war, dass die Speisung genügend Strom liefert. Der nach Datenblatt angegebene Stromverbrauch der einzelnen Geräte ist mit folgenden Werten Angegeben (Supply Current):

MCB517A-Board:	300-500 mA (typ. 300 mA)	8 ...12V DC
GPS-Empfänger-Modul:	150 mA max.	7 ...18V DC
LCD-Display-Modul:	5 mA max. (typ. 2.3 mA)	5.0V DC
Backlight LCD-Display:	240 mA	4.0V DC
Total:	895 mA (max.) / 692.3 mA (typ.) ⇒ **800 mA**	

Sicherungen

Der Strom für das Display-Modul ist vernachlässigbar, dafür benötigt die Hintergrundbeleuchtung satte 240 mA. Aus diesem Grunde der separate Trafo (T2). Anhand der obigen Auflistung konnten zudem die Feinsicherungen bestimmt werden:

F1 (T1, primärseitig)	= 0.1A T	(Erfahrungswert, identisch mit F2)
F3 (Autobatterie)	= 1A T	

Spannungsregler

Die Werte des Ein-und Ausgangs-Stabilisierungskondensators (C5 bzw.C6) des Spannungsreglers wurden aus bestehenden Applikationen entnommen (100µF) und erwiesen sich als vernünftig, da eine Ausgangsspannung von 5.1V gemessen werden konnte. Die durch den Kühlkörper zu vernichtende Wärme-Verlustleistung errechnet sich aus (Ue-Ua) • I und beträgt:
(11.8V-5.1V) • ~ 40mA = <u>268 mW</u>

Spannungsversorgung

Das detaillierte Schema zur HW der Laborkarte 1 ist im *Anhang* ersichtlich.

5.5 Schaltung zur Adressdecodierung

Diese ist das Kernstück meiner Hardware und stellte mich zu Beginn vor einige Probleme, da der erste durch den Dozenten grob vorgegebene Lösungs-Ansatz weitgehend modifiziert werden musste. Im folgenden wird kurz auf diesen ersten Ansatz und die dabei entstehenden Probleme mit dem LCD-Controller eingegangen, bevor die Lösung in Form der überarbeiteten Schaltung aufgezeigt wird.

5.5.1 Adressbereich

Der durch das Prototypen-Board vorgegebene User-Chip-Select-Bereich befindet sich zwischen 0F600H und 0F700H. Für diese Anwendung wurde die Adresse 0F6F0H als Startadresse bestimmt.

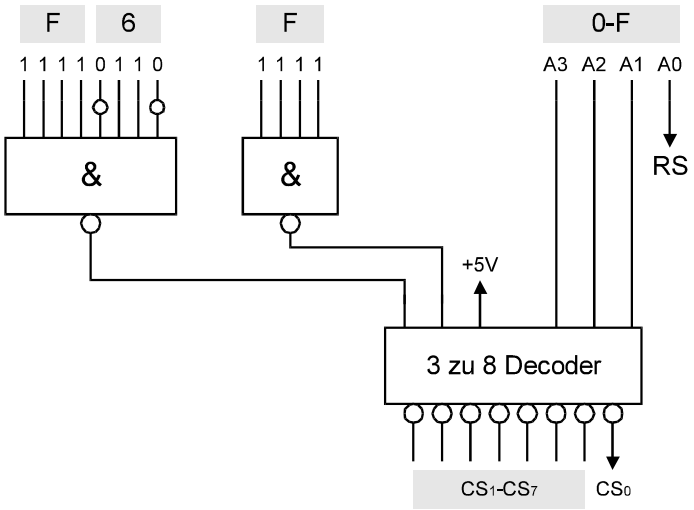
Chip-Select-Adresse: 0x0F6F0H

5.5.2 Prinzip und 1. Ansatz

Die Vorgabe bestand nun darin, ausgehend von dieser Adresse eine Decodierschaltung für 8 Peripherie-Einheiten zu je 2 Adressen zu entwickeln. Dies heisst also, dass innerhalb des Bereiches von 0F6F0H-0F6FFH die 16 zur Verfügung stehenden Bits in 8 Bereiche (Chip-Select 0-7) zu je 2 Bits (und somit auch 2 Adressen) unterteilt werden. Dabei soll das LCD-Display über $\overline{CS_0}$ und die Tastatur über $\overline{CS_1}$ angesprochen werden. Die folgende Darstellung verdeutlicht das grundlegende Prinzip:

Decodierschaltung

Der LCD-Controller benötigt zur korrekten Ansteuerung insgesamt 4 Adressen.



5.5-1 Adressdecodierung nach 1. Ansatz

Die Adressleitung A0 wird direkt mit dem RS-Signal des LCD-Controllers verbunden.

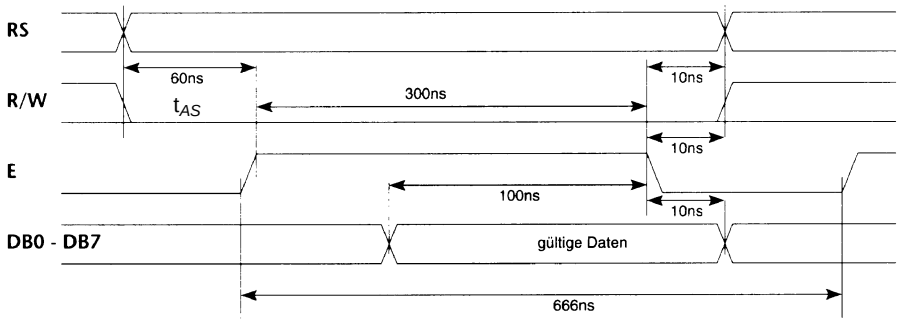
Man erkennt, dass der feststehende Teil der Adresse (F6F_) mit einfachen NAND-Gliedern (IC1/IC2) und Invertern (IC3) realisiert werden kann. Die Ansteuerung des Chip-Select-Bereiches erfolgt durch einen 3 zu 8-Decoder, welcher mit den Adressleitungen A1-A3 verbunden ist. Die Leitung A0 wird *direkt* mit dem RS-Pin des LCD-Controllers verbunden, damit dieser je nach angewählter Adresse zwischen Daten und Befehlen unterscheiden kann.

Mit dem 3 zu 8-Decoder DM74LS138 (IC4) können mit 3 Eingängen bis zu 8 Ausgänge selektiert werden.

Adressleitungen A0-A3					Aktion	Peripherie
A3	**A2**	**A1**	**A0**			
		R/W	**RS = 0**	**RS = 1**		
0	0	0	F6F0H	F6F1H	Write	LCD
0	0	1	F6F2H	F6F3H	Read	
			Befehle	Daten		
0	1	0	0	F6F4H		Keyboard

5.5.3 Das Problem mit dem Timing

Die gezeigte Ansteuerung hätte funktioniert, wenn es da nicht zu Timing-Problemen auf dem Bus des Mikrocontrollers gekommen wäre.
Bei eingehender Betrachtung der folgenden Timing-Diagramme sowie der Logikschaltung habe ich folgendes festgestellt:



5.5-2 Mindestsignalzeiten beim Schreibvorgang in das Display-RAM

Der Lesevorgang ist bis auf das invertierte R/$\overline{W}$-Signal identisch.

Das R/$\overline{W}$-Signal des LCD-Controllers muss mindestens um die Zeit t_{AS} vor dem Enable-Signal E am Controller anliegen. Da jedoch in der oben gezeigten Ansteuerung genau diese beiden Signale auf die Logikschaltung geführt sind und somit nicht einzeln behandelt werden können, kommen sie höchstens um die Zeit eines Gatters verzögert (10ns) am LCD-Controller an. Dies führt dazu, dass das E-Signal nie HIGH wird und die am Datenbus anstehenden Daten nicht erkannt werden.

Die erforderliche Zeitverzögerung t_{AS} zwischen dem R/$\overline{W}$- und dem E-Signal wird nicht eingehalten.

5.5.3.1 Lösung

Nachdem ich dies erkannt hatte, lag die Lösung auf der Hand: Um die benötigte Zeitverzögerung zu erhalten, führte ich einfach *beide* Steuersignale RS und R/$\overline{W}$ des Controllers *direkt* auf die Adressleitungen A0 bzw. A1. Somit liegen sie unmittelbar nach der Freigabe der Adressleitungen durch den Mikrocontroller am LCD-Controller an, während das E-Signal durch den “Umweg” über die Logikschaltung die erforderliche Zeitverzögerung erhält.

Die Steuersignale RS und R/$\overline{W}$ des LCD-Controllers werden beide direkt mit den Adressleitungen A0 bzw. A1 verbunden.

Einschränkungen

Durch diese Massnahme entstehen jetzt natürlich Einschränkungen in Bezug auf die Anzahl der zur Verfügung stehenden Chip-Selects. Da nur noch 2 Adressleitungen auf den Adressdecoder geführt werden, stehen jetzt 4 Bereiche zu je 4 Adressen zur Verfügung, was im Falle des LCD-Controllers wohl notwendig, für andere Peripherie-Einheiten in den meisten Fällen jedoch nicht erforderlich ist. Trotzdem wurde die Lösung nach Absprache mit dem Dozenten übernommen und die Schaltung entsprechend angepasst.

Anstelle von 8 stehen nur noch 4 Ausgänge am Decoder DM74LS138 zur Verfügung.

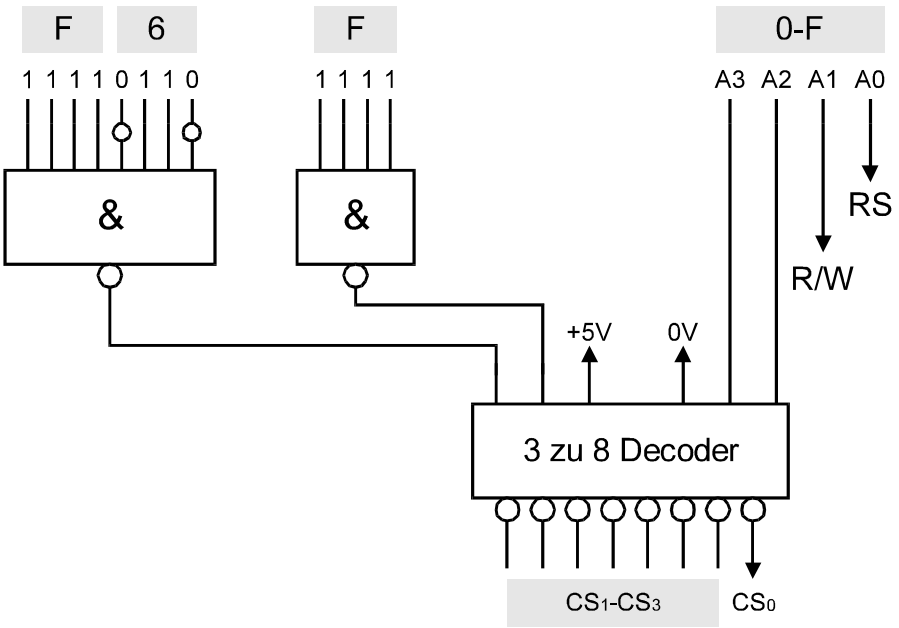
5.5.4 Modifizierte Adressdecodierung und Logikschaltung

Adressdecodierung

An der bereits bestehenden Decodierschaltung musste ich also nur die folgenden Änderungen vornehmen:

Modifizierung

- Die Adressleitung A1 wird direkt mit dem R/$\overline{W}$-Eingang des LCD-Controllers verbunden
- Um die Funktion des 3 zu 8-Decoders mit nur 2 Eingängen zu gewährleisten, muss der Eingang C auf Masse (LOW) gelegt werden
- Die Adressleitungen A2 und A3 werden somit auf die Eingänge A bzw. B des 3 zu 8-Decoders verlegt



5.5-1 Modifizierte Adressdecodierung

Die Adressleitungen A0 und A1 werden direkt mit dem RS- bzw. R/$\overline{W}$-Signal des LCD-Controllers verbunden.

Die bestehende Schaltung musste nun für eine vollständige Kommunikation des Mikrocontrollers mit dem LCD-Display um eine geeignete Logikschaltung erweitert werden, da das RAM des LCD-Controllers nicht nur beschrieben, sondern zudem auch aus dem ROM gelesen werden kann.

Logikschaltung

Diese Schaltung bewirkt, dass die Lese- und Schreibzugriffe des Mikrocontrollers nur dann ein Enable-Signal auslösen, wenn das LCD-Display durch die Ausgabe der richtigen Adresse selektiert und somit die Steuersignale RS bzw. R/W richtig gesetzt sind.
Der Aufbau erfolgte zuerst mit NAND-Gliedern und Invertern. Durch die Modifikation der Adressdecodierung konnte die definitive Schaltung jedoch optimiert und auf einen Baustein (IC5) reduziert werden. Dieser ist ein NOR-Baustein mit 4 Gliedern zu je 2 Eingängen.

Die Logikschaltung besteht aus nur einem High-Speed CMOS NOR-Baustein MM74HC02.

Die Funktion der Schaltung erhielt ich aus der folgenden Wahrheitstabelle:

$\overline{CS_0}$	$\overline{RD}$	$\overline{WR}$	R/$\overline{W}$	E
DM74LS138	Mikrocontroller		LCD-Controller	
0	0	0	X	X
0	0	1	1	1
0	1	0	0	1
0	1	1	X	0

Anhand dieser Wahrheitstabelle konnte ich mir das Zusammenspiel von Mikrocontroller und LCD-Controller besser vorstellen.

Der Chip-Select $\overline{CS_0}$ ist LOW-aktiv. Ein HIGH-Signal am Enable-Eingang des LCD-Controllers ist nur bei einem Schreib- bzw. Lesevorgang des Mikrocontrollers erwünscht.
Die Implementierung Tastatur erforderte keine zusätzliche Beschaltung, da der Key-Encoder MM74C922 lediglich durch die Selektierung mit $\overline{CS_1}$ aktiviert werden muss.

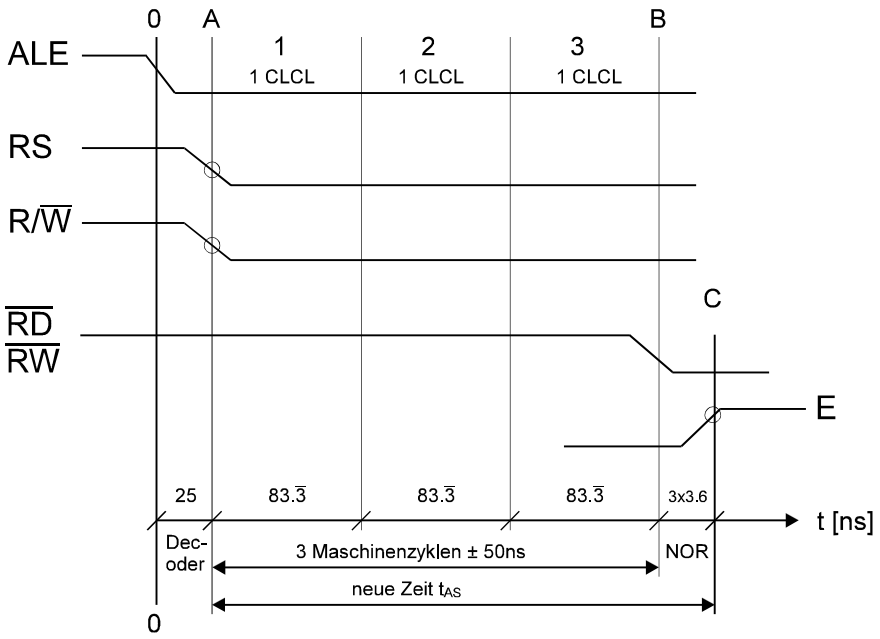
Die definitive Zuordnung der Adressen und deren Funktionen kann aus der untenstehenden Tabelle entnommen werden.

Chip-Select	Adresse [H]	Adressleitungen				Funktion
		A3	A2	A1 (R/$\overline{W}$)	A0 (RS)	
$\overline{CS_0}$	F6F**0**	0	0	**0**	**0**	Befehle auf LCD schreiben
	F6F**1**	0	0	**0**	**1**	Daten auf LCD schreiben
	F6F**2**	0	0	**1**	**0**	Befehle von LCD Lesen (Busy-Flag)
	F6F**3**	0	0	**1**	**1**	Daten von LCD Lesen (z.B. aus ROM)
$\overline{CS_1}$	F6F4	0	1	0	0	Keyboard
	-	0	1	0	1	-
	-	0	1	1	0	-
	-	0	1	1	1	-

5.5.4.1 Timing-Diagramm

Um die definitive Schaltung erneut auf Timing-Probleme zu überprüfen, verglich ich die Timing-Diagramme des LCD- sowie des Mikrocontrollers (siehe 5.1.2.3) und erhielt daraus das folgende Diagramm. Da ich in der ganzen Zeit, in der ich die Hardware zum Testen der Software benötigte, keine Probleme oder Störungen feststellen konnte, gehe ich davon aus, dass der zeitliche Ablauf bei einem Schreib- oder Lesezugriff in etwa dem Diagramm folgenden entspricht.

Um die Zeit t_{AS} nicht unnötig gross werden zu lassen, wird für die Logikschaltung ein High-Speed CMOS-IC verwendet.



5.5-4 Schreib-/ Lesezugriff auf LCD-Controller

Indem die beiden Steuersignale RS und R/$\overline{W}$ des LCD-Controllers direkt auf die Adressleitungen A0 bzw. A1 geführt werden, kann die geforderte Zeit t_{AS} bis zum E-Signal eingehalten werden.

Bis nur der Mikrocontroller ausgehend vom ALE-Signal das $\overline{RD}$ bzw. $\overline{WR}$-Signal freigibt, vergehen 3 Maschinenzyklen, was bei einem Takt von 12MHz ca. 200-300 ns entspricht. Dazu addiert sich die Signal-Laufzeit des Adressdecoders sowie 2 bis 3 x die Signal-Laufzeit des NOR-Gliedes.

5.5.4.2 Verwendete IC's

- **8-Input NAND** ***DM74LS30 (IC1)***
  Ein einfaches NAND-Glied mit 8 Eingängen und einem Ausgang.
- **Dual 4-Input NAND** ***DM7420 (IC2)***
  Ein einfaches NAND-Glied mit 4 Eingängen und einem Ausgang.
- **Hex-Inverter** ***SN7404 (IC3)***
  Inverter-Baustein mit 6 einzelnen Invertern.
- **3 zu 8 Decoder / Demultiplexer** ***DM74LS138 (IC4)***
  Mit 3 Eingängen können 8 Ausgänge angewählt werden.
- **Quad 4-Input NOR** ***MC74HC02 (IC5)***
  High-Speed NOR-Baustein mit 4 Gliedern zu je 2 Eingängen und einem Ausgang sowie einer Signal-Laufzeit von typ. 3.6ns bei Vcc = 5V.

5.6 LCD-Display-Modul

Das Display war aufgrund den Vorgaben (4 Zeilen zu je 20 Zeichen, Hintergrundbeleuchtung) schnell evaluiert und wurde mir von der Firma DISPLAY LC AG freundlicherweise gratis abgegeben. Gesteuert wird es vom weit verbreiteten HD44780-LCD-Controller/Driver der Firma HITACHI.

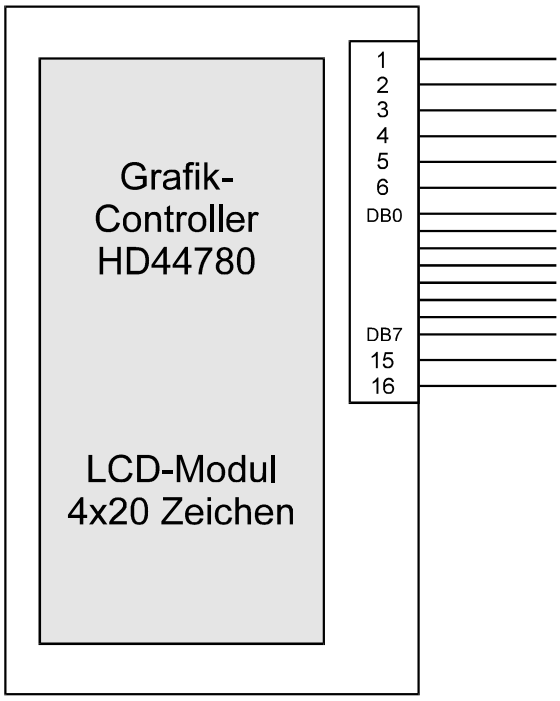
LCD-Display
Typ DMC20481NY-LY

5.6.1 Ansteuerung

Das Display besitzt insgesamt 16 Anschlüsse, wovon 8 als Daten- und 3 als Steuerleitungen Verwendung finden. Von den restlichen 5 werden 3 für die Speisung (+5V) und 2 für die Hintergrundbeleuchtung (ca. 4V) benötigt. Der Kontrast der Anzeige kann über ein Potentiometer eingestellt werden. Ebenso ist es sinnvoll, den Strom für die Hintergrundbeleuchtung mit einem Widerstand zu begrenzen. Dabei ist jedoch auf die max. Verlustleistung des Widerstandes zu achten.

Die 8 Datenleitungen können direkt mit dem Datenbus des Mikrocontrollers verbunden werden

Die folgende Darstellung zeigt die Anschlussbelegungen des Displays.



Pin	Bez.	Funktion
1	V_{SS}	GND, 0V
2	V_{CC}	+5V Logic
3	V_{EE}	+4V Backlight
4	RS	Register Select
5	R/$\overline{W}$	Read / Write
6	E	Enable
7	DB0	Data Bus Line
8	DB1	Data Bus Line
9	DB2	Data Bus Line
10	DB3	Data Bus Line
11	DB4	Data Bus Line
12	DB5	Data Bus Line
13	DB6	Data Bus Line
14	DB7	Data Bus Line
15	Anode	LED Terminal
16	Kathode	LED Terminal

5.6-1 Anschlussbelegungen des Displays

Das detaillierte Schema zur HW der Laborkarte 2 ist im *Anhang* ersichtlich.

Adressdecodierung
Ansteuerung LCD und Tastatur

5.6.2 Dimensionierungen

Da das Potentiometer für die Einstellung des Kontrastes bei Normal-Temperatur (20-30 °C) aufgrund des Datenblattes mit 10kΩ vorgegeben war, musste ich lediglich einen geeigneten Widerstand für die Srombegrenzung der Hintergrundbeleuchtung berechnen. Die "Berechnungen" beschränkten sich in diesem Fall auf das ohmsche Gesetz, da ich den Widerstand messtechnisch ausfindig machte: Ich speiste die Hintergrundbeleuchtung mit dem eigens dafür vorgesehenden Transformator T2 (9V) über ein Potentiometer von 100Ω und 5 Watt. Dabei stellte ich fest, dass für eine ausreichende Beleuchtung ein Strom von 0.2A-0.3A bei einer Anoden/Kathoden-Spannung von ca 4V erforderlich ist. Die restlichen 5V müssen über dem Widerstand abfallen ! Der Widerstandsbereich ergab sich aufgrund der gemessenen Spannungen von 9.4V sowie 8.1V mit der Formel R = U/I.

Für eine ausreichende Hintergrundbeleuchtung ist ein Strom von 0.2A bis 0.3A nötig.
Der verwendete Widerstand muss eine max. Verlustleistung von min. 1 Watt aufweisen.

$R_{min.} = 13.6\Omega$ $\qquad$ $R_{max.} = 27\Omega$

Zu beachten hatte ich in diesem Fall die maximale Verlustleistung des Widerstandes, welche sich mit der Formel $P = I^2 \bullet R$ berechnen lässt:

$P_{Rmin.} = 1.23W$ $\qquad$ $P_{Rmax.} = 1.08W$

Da ich gerade kein geignetes Potentiometer mit dieser Verlustleistung zur Verfügung hatte, wählte ich einen Drahtwiderstand von 22Ω und 2W, welcher den Strom auf 0.22A begrenzt (gemessen). Die Verlustleistung beträgt somit 1.06W. Dies ist vertretbar, wenn man bedenkt, dass die Beleuchtung ja nur bei Dunkelheit erforderlich ist; die Messungen jedoch meist in beleuchteten Räumen oder im Freien stattfinden.

5.6.3 LCD-Controller HD44780

Über diesen Baustein möchte ich nicht allzuviele Worte verlieren, da er erstens weit verbreitet und zweitens im Anhang ein ausführliches Datenblatt vorhanden ist.

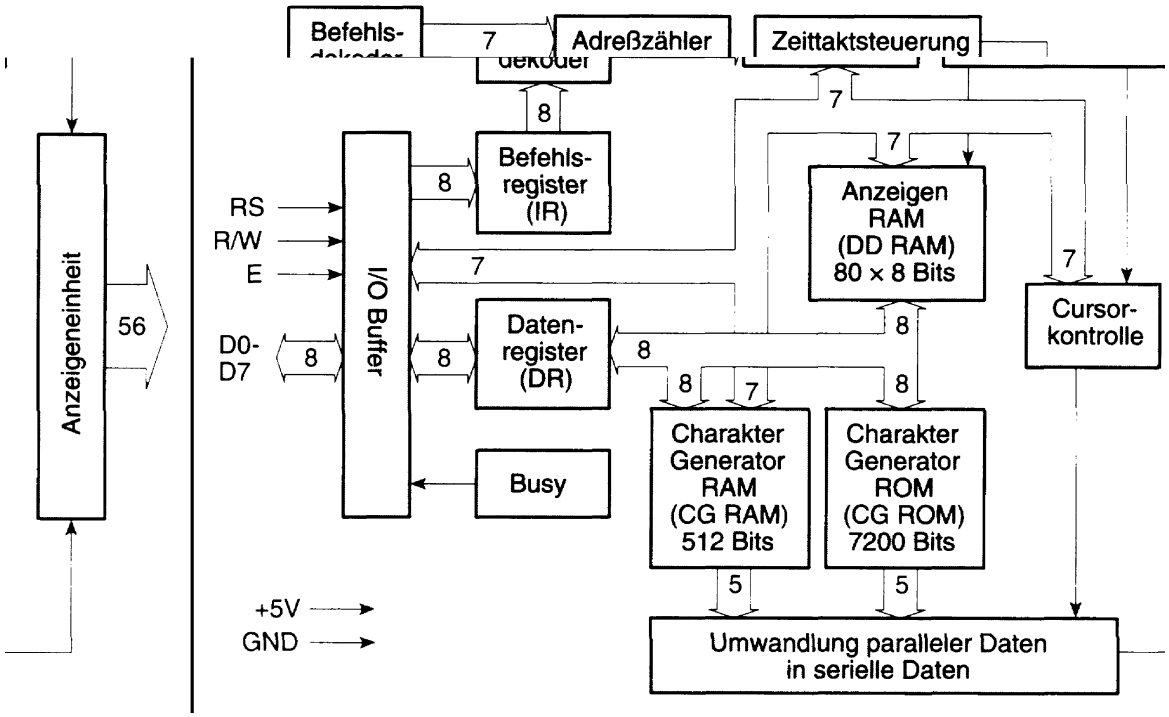
Features

Einige Features sind:

- Ansteuerung von Zeichen mit 5x7 oder 5x10 Punkt-Matrix
- Kommunikation über 4- oder 8-Bit Datenbus
- Über 200 im internen ROM abgespeicherte Zeichen
  (ASCII-Zeichensatz, Umlaute, Sonderzeichen, Kana-Zeichen)
- Max. 80 darstellbare Zeichen (80x8-Bit Display RAM)
- Programmierung von selbst definierten Zeichen möglich
- Befehlssatz zur Steuerung der Aktivitäten
  (Display clear, Display ON/OFF, Blinkfunktion für Zeichen/Cursor)

Ein Blick auf das Funktionsdiagramm des Controllers lässt seine Arbeitsweise und Fähigkeiten besser verstehen:

Funktionsdiagramm



5.6-1 Funktionsdiagramm des Grafikcontrollers HD44780

Als Nachteil ist die Arbeitsgeschwindigkeit des Controllers aufzuführen, welche sich bei der Kommunikation über den Adress-/Daten-Bus des Mikrocontrollers bemerkbar machte.

Nachteile

Zudem stellt der Controller ein Flag zur Verfügung (Busy-Flag, Bit 7 des Adresszählers), welches softwaremässig abgefragt werden muss, bevor eine neue Operation gestartet wird. Ist dieses Flag = 1, ist der Controller noch beschäftigt und gestattet keine weiteren Zugriffe; ist es = 0, kann eine weitere Operation ausgeführt werden. Die Initialisierung verlangt aus diesem Grund ebenfalls einen Programmablauf mit etlichen Verzögerungszeiten (siehe S.17).

Bei jedem Zugriff muss der Controller mit Hilfe eines Flags auf seine Bereitschaft abgefragt werden.

5.6.3.1 Schreib-/Lesezugriff

Dieser wird durch die beiden Steuersignale RS und R/$\overline{W}$ wie folgt bestimmt:

R/$\overline{W}$	RS	Vorgang
0	0	Schreiben von Befehlen
0	1	Schreiben von Daten
1	0	Lesen des Adresszählers und des Busy-Flags
1	1	Lesen von Daten

5.6.3.2 Befehlssatz

Der Befehlssatz des Grafik-Controllers ist relativ umfangreich.
Im folgenden die wichtigsten Befehle und ihr entsprechendes Bitmuster:

Befehl	RS	R/$\overline{W}$	D7	D6	D5	D4	D3	D2	D1	D0	Beschreibung	Zeit
Clear Display	0	0	0	0	0	0	0	0	0	1	Löscht die Anzeige und setzt Display-RAM-Adresse auf 0	1640 µs
Return Home	0	0	0	0	0	0	0	0	1	X	Setzt die Display-Daten-RAM-Adresse (DD) auf 0. Der DD-RAM-Inhalt bleibt unverändert.	1640 µs
Eintrage-Modus	0	0	0	0	0	0	0	1	I/D	S	Regelt die Richtung der Cursor-Bewegung und das Display-Shift.	40 µs
Display ON / OFF	0	0	0	0	0	0	1	D	C	B	Schaltet die Anzeige (Bit D) und den Cursor (Bit C) an oder aus und kontrolliert das Blinken des Zeichens unter dem Cursor (Bit B).	40 µs
Shift	0	0	0	0	0	1	S/C	R/L	X	X	Bewegt den Cursor und schiebt die Anzeige ohne Änderung des DD-RAMs	40 µs
Interface	0	0	0	0	1	DL	N	F	X	X	Setzt die Interface-Daten-Breite (DL), die Zahl der Zeilen (N) und die Pixel-Matrix (F).	40 µs
Charakter RAM Adresse setzen	0	0	0	1	Adresse Charakter Generator						Setzt die Charakter-RAM-Adresse (CG-RAM). Alle nachfolgenden Daten wandern danach ins CG-RAM.	40 µs
Anzeigen RAM Adresse setzen	0	0	1	Adresse Display Daten							Setzt die Display-Daten-RAM-Adresse (DD-RAM). Alle nachfolgenden Daten wandern danach ins DD-RAM (Anzeige).	40 µs
Busy Flag und Adresse lesen	0	1	BF	Adresszähler							Liest das Busy Flag (BF) und die Adresse der Anzeige.	40 µs
Write Data	1	0	Daten schreiben								Schreibt Daten in das Anzeigen-RAM bzw. in das Charakter-RAM.	40 µs
Read Data	1	1	Daten lesen								Liest Daten aus dem Anzeigen-RAM bzw. aus dem Charakter-RAM.	40 µs

Initialisierung

Die Initialisierungs-Routine verlangt aufgrund der Arbeitsgeschwindigkeit des LCD-Controllers etliche Warteschlaufen.

An dieser Stelle noch eine Bemerkung zu der bereits angesprochenen softwaremässigen Initialisierung des LCD-Controllers:

Nach einer ersten Wartezeit von min. 15 ms muss der entsprechend parametrierte Interface-Befehl (siehe Tabelle) **ohne** Abfrage des Busy-Flags zum Controller geschickt werden. Dies muss nach erneuten Wartezeiten von 4.1 ms und 100 µs noch 2 mal wiederholt werden, bis das Busy-Flag abgefragt und die Initialisierung mit weiteren Befehlen beendet werden kann.

Der initialisierungs-Ablauf ist im Datenblatt zum Display-Modul sowie des Controllers (siehe *Anhang*) ausführlich beschrieben und im Assembler-Listing ebenfalls gut ersichtlich.

5.7 Tastatur 4x4 Tasten

Tastatur

Bei der Evaluation dieser Eingabeeinheit achtete ich vorwiegend auf den Preis, da ansonsten keine weiteren Vorgaben erfüllt werden mussten. Das einzige Kriterium für mich war, dass die Tastatur in Matrix-Ausführung erhältlich sein musste (die 16 Tasten eingeteilt in 4 Zeilen und 4 Spalten). Dies hat seinen Grund im nachfolgend beschriebenen Key-Encoder, welcher 16 Tasten auf diese Weise elegant und einfach decodiert.

Als ich hörte, dass eine Tastatur mit 16 Tasten an den Mikrocontroller angeschlossen werden musste, hatte ich jedoch zuerst zwei Probleme: Die unangenehme Eigenschaft der Prellung einer jeden Taste, die eine softwaremässige Auswertung erschwert sowie die Anzahl der Portleitungen, die ich dafür verbrauchen würde.
Das Problem der Portleitungen wurde bereits im Vorfeld beseitigt, da die Tastatur ebenfalls über den Adress-/Datenbus des Mikrocontrollers angesprochen werden soll. Aufgrund dieser Vorgabe war die Realisierung einer Matrix mit 4 Zeilen und 4 Spalten fast schon gegeben.
Der Gedanke an die softwaremässige Entprellung liess mich nach einem geeigneten Baustein suchen, welcher nebst der Tasten-Decodierung auch dieses Problem bereits hardwaremässig berücksichtigt. Das Erkennen von mehreren gleichzeitig betätigten Tasten ist nicht erforderlich.

1	2	3	F
4	5	6	E
7	8	9	D
A	0	B	C

5.7.1 16-Key Encoder MM74C922

MM74C922 (IC6)

Fündig wurde ich mit dem MM74C922 Keyboard-Encoder. Dieser Baustein beinhaltet die ganze Logik, um eine Tastaturmatrix von 16 Tasten in einem digitalen System zu implementieren.
Dabei wird eine betätigte Taste (Schliesser) in einem entsprechenden 4-Bit-Code wiedergegeben. Die Tasten-Abtastfrequenz (OSC) und die Entprellzeit (KBM) kann durch die externe Beschaltung von Kondensatoren definiert werden.

Funktion anhand des Datenblattes

Funktion

Die 16 Tasten werden in Form einer Matrix mit 4 Zeilen und 4 Spalten am MM74C922 angeschlossen. Wird keine Taste betätigt, werden die Zeileneingänge durch die internen Pull-Up-Widerstände auf HIGH und die Spaltenausgänge sequentiell auf LOW gesetzt. Die Spaltenabtastung besteht aus einem Oszillatoreingang, einem 2-Bit-Zähler und einem 2 zu 4-Bit Decoder.
Wird nun eine Taste betätigt (z.B. Taste 0), geschieht solange nichts wie X1 offen ist, da der Pull-Up-Widerstand Y1 auf HIGH zieht. Wird nun X1 abgetastet, zieht dieser (da der Ausgang jetzt LOW-Pegel hat) den Y1-Eingang ebenfalls auf LOW. Die Entprelleinheit wird gestartet und die Y-Eingänge gesperrt. Der Tastencode setzt sich aus dem eingefrorenen Zählerstand und den decodierten Y-Eingängen zusammen. Nach dem Ablaufen der Entprellzeit wird der Tastencode ins interne Register übernommen und der DA-Ausgang Hauf HIGH gesetzt.

Der Tastencode setzt sich aus dem eingefrorenen Zählerstand und den decodierten Y-Eingängen zusammen

5.7.2 Ansteuerung

Die Zeilen- (X) und Spaltenleitungen (Y) werden direkt auf die Tastatur geführt. Somit werden nebst den 4 Datenleitungen, welche mit den Datenleitungen D0-D3 des Mikrokontrollers verbunden werden, nur noch 2 Steuerleitungen benötigt. Dies wäre zum einen die “Output Enable” ($\overline{OE}$)- Leitung, welche das Signal ($\overline{CS1}$) vom Adress-Decoder bei der Selektierung erhält.
Zum anderen wird der “Data Available”- (DA) Ausgang benötigt, um bei einer betätigten Taste mit Hilfe einer steigenden Flanke am Port-Pin 1.1 des Controllers den externen Interrupt 4 auszulösen.

Die Implementierung einer Tastatur wird durch den Key-Encoder MM74C922 erheblich vereinfacht.

5.7.3 Tastencodierung

Da die Tasten-Codes der verwendeten Tastatur nicht mit denen des Standard-Modells im Datenblatt des MM74C922 übereinstimmen, ist die entsprechende Aufschlüsselung der folgenden Darstellung zu entnehmen:

	X1	X2	X3	X4
Y1	1 / 00H / 0000B	2 / 01H / 0001B	3 / 02H / 0010B	F / 03H / 0011B
Y2	4 / 04H / 0100B	5 / 05H / 0101B	6 / 06H / 0110B	E / 07H / 0111B
Y3	7 / 08H / 1000B	8 / 09H / 1001B	9 / 0AH / 1010B	D / 0BH / 1011B
Y4	A / 0CH / 1100B	0 / 0DH / 1101B	B / 0EH / 1110B	C / 0FH / 1111B

5.7-1 Tastencodes (LOB)

Hinweis:
Die Tastencodes werden zur optischen Kontrolle an die LED’s von Port 4 ausgegeben.

5.7.4 Dimensionierungen

Die Tastatureinheit erforderte keine grossen Berechnungen. Einzig die beiden Kondensatoren, welche die Abtastfrequenz (OSC) und die Entprellperiode (KBM) definieren, müssen bestimmt werden.

Auch hier verwendete ich in Anlehnung an das Datenblatt sowie bereits bestehende Applikationen Erfahrungswerte, damit ich die Tastatureinheit so schnell wie möglich aufbauen und auf der Experimentierplatte testen konnte.

Die so eingesetzten Werte betragen für:

Oscillator Capacitor: $\mathbf{C}_{OSC}$ = 47 nF
Key Bounce Mask Capacitor: $\mathbf{C}_{KBM}$ = 4.7 µF

Den zuerst verwendeten Wert von $\mathbf{C}_{KBM}$ = 4.7 µF habe ich aufgrund dem mechanisch bedingten Schaltverhalten der Tastatur nachträglich auf 1.5 µF verringert. Infolge dessen setzte ich für $\mathbf{C}_{OSC}$ = 150 nF, da nach Datenblatt gilt:

$$\mathbf{C}_{KBM} = 10 \text{ x } \mathbf{C}_{OSC}$$

Die neu eingesetzten Werte betragen für:

Oscillator Capacitor: $\mathbf{C}_{OSC}$ = 150 nF
Key Bounce Mask Capacitor: $\mathbf{C}_{KBM}$ = 1.5 µF

Während der ganzen Zeit, in der ich meine Hardware zum Testen der Software benötigte, konnte ich keine Probleme oder Störungen mit den oben genannten Werten feststellen.

Das detaillierte Schema zur HW der Laborkarte 2 ist im *Anhang* ersichtlich.

Adressdecodierung
Ansteuerung LCD und Tastatur

6. Software

```
LCD_TXT:    CLR     A
            MOVC    A,@A+DPTR
            JZ      LCD_ENDE
            CALL    LCD_WRITE
            INC     DPL
            JMP     LCD_TXT
LCD_ENDE:   RET
TXT_ENDE:   CALL    LCD_OFF_TXT
            JMP     LCD_ENDE
```

Zu Beginn dieses Kapitels werden die beiden vom GPS-Empfänger-Modul zur Verfügung gestellten Protokolle vorgestellt. Im weiteren wird die Idee des Programm-Ablaufs aufgezeigt sowie auf einige Software-Module näher eingegangen.

6.1 GPS-Protokolle

Der folgende Abschnitt beschreibt die Struktur und das Satzformat des Standard-NMEA-Protokolles, welches aufgrund seiner Herkunft die vollständige Bezeichnung **N**ational **M**arine **E**lectronics **A**ssociation-Protokoll trägt.

6.1.1 Das NMEA-Nachrichten-Format

NMEA-Format

Das NMEA-0183-Protokoll ist ein ASCII-Format. Jeder Datensatz (oder Block) beginnt mit dem ASCII-Zeichen “$” (24H) und endet mit ASCII <CR><LF> (0DH und 0AH) [Carriage Return und Line Feed]. Der gültige Zeichensatz besteht aus allen druckbaren ASCII-Zeichen (20H bis 7EH), ausgenommen der Zeichen, welche in der Tabelle 3.4-2 aufgeführt sind.
Jede NMEA-Nachricht oder -Satz besteht aus mehreren Daten-Feldern, welche durch einen sog. Delimiter (in diesem Fall ein Komma “,”) getrennt sind. Die Datenfelder enthalten einen String aus gültigen Zeichen, können aber auch leer sein. Gültige Zeichen müssen mit den Angaben in Tabelle I-3 (siehe *Anhang zu Kapitel 3*, gelb) übereinstimmen.
Die maximale Zeichen-Anzahl pro Satz beträgt 82, bestehend aus maximal 79 Zeichen zwischen dem Start-Delimiter “$” und den abschliessenden Zeichen <CR> und <LF> (Carriage Return und Line Feed).
Dass die Anzahl der Datenfelder von Satz zu Satz varieren kann, ist es für den Programmierer wichtig zu wissen. Um Felder zu lokalisieren, ist es somit besser die Kommas zu zählen, als sich auf die Anzahl der total empfangenen Zeichen zu verlassen. Sollten die Daten, aus welchen Gründen auch immer, einmal nicht verfügbar sein, wird der Satz (Block) trotzdem gesendet. In diesem Fall folgt das nächste Komma sofort auf das vorhergehende.

Anerkannte NMEA-0183 Sätze

Ein anerkannter NMEA-0183-Satz beinhaltet die folgenden Elemente in der Reihenfolge von oben nach unten:

“$”	Startzeichen (24H)	*“$”*
<Adress-Feld>	Satz-Identifier und -Formatierer	*<Adress-Feld>*
[“,”<Daten-Feld>]	kein oder mehrere Datenfelder	*[“,”<Daten-Feld>]*
[“,”<Daten-Feld>]	das letzte Datenfeld	*[“,”<Daten-Feld>]*
[“*”<Checksummen-Feld>]	optionale Checksumme	*[“*”<Checksummen-Feld>]*
<CR><LF>	Endzeichen (0DH, 0AH)	*<CR><LF>*

Checksumme

Diese ist die 8-Bit exklusiv-oder-(XOR) Verknüpfung (keine Start- oder Stop-Bits) von allen Zeichen eines Satzes, inklusive den Datenfeld-Delimitern (,), jedoch ohne das Startzeichen ($) und den optionalen Checksummen-Delimiter (*). Der hexadezimale Wert der ersten vier (LSB) und der letzten vier (MSB) Bits des Resultates wird zwecks Übertragung zu zwei ASCII-Zeichen konvertiert (0-9,A-F). Das most significant Zeichen wird zuerst übertragen.

6.1.2 Das Binary Message Format

Binary-Format

Obwohl in dieser Arbeit die benötigten Daten des NMEA-Protokolles zur Positionsbestimmung ausgewertet werden, möchte ich kurz das ebenfalls vom GPS-Empfänger-Modul zur Verfügung gestellte *"Binary Message Format"* vorstellen und einige Unterschiede sowie Vor- und Nachteile gegenüber dem NMEA-Protokoll zusammenfassend aufzeigen.

Zodiac Binary Message Format

Jede *binary message* besteht aus einem Header- und einem Datenteil, wobei jeder Teil seine eigene Checksumme hat. Jede Nachricht besteht mindestens aus einem Header, doch nicht jede Nachricht hat auch einen Datenteil. Die Erkennung der Nachricht erfolgt durch den Header.

Header-Struktur

1000 001 1111 1111	Wort 1
Nachrichten ID	Wort 2
Datenwort-Zähler	Wort 3
DCLE QRAN 00XX XXXX	Wort 4
Header-Checksumme	Wort 5

Daten-Struktur **(Erweitert)**

Trigger (on time, on update)	Wort 6
Intervall (sec)	Wort 7
Offset (sec)	Wort 8
Nachrichten-abhängige Daten	Wort 9
...	Wort 9+n
Daten-Checksumme	Wort 10+n

Ein Wort bzw. ein Integer ist mit einer Länge von 16 Bit definiert.

6.1.3 Unterschiede

Unterschiede

Da das binary-Format ein eigens zur Kommunikation mit dem Zodiac-GPS-Empfänger-Modul geschaffenes Protokoll darstellt, bietet es mehr Möglichkeiten zur Datenauswertung als das NMEA-Protokoll. Dies ist auch an der Anzahl der verfügbaren Nachrichten in nachstehender Tabelle ersichtlich. Da es bereits in binärer Form vorliegt, würde bei einer Weiterverarbeitung der Daten die ASCII-dezimal-Wandlung entfallen.

Die bidirektionale Kommunikation (vom PC zum Empfänger und umgekehrt) ist mit beiden Protokollen möglich. Wobei natürlich die vom Empfänger auszugebenden Nachrichten vom PC aus angefordert werden müssen.

Das binary Protokoll bietet mehr Möglichkeiten zur Datenauswertung

Eigenschaft	NMEA-Protokoll	Rockwell binary Protokoll
Standard	JA	-
Hersteller-spezifisch (proprietary)	JA	JA
Datenübertragung über RS 232	[4800 bps] [8 Daten-Bits] [keine Parität (N)] [1 Stop-Bit]	[9600 bps] [8 Daten-Bits] [keine Parität (N)] [1 Stop-Bit]
Übertragungs-Format	ASCII	binär
Ausgabe-Nachrichten	7	14
Eingabe-Nachrichten	4	14
DGPS-fähig	NEIN	JA

Weiterführende Informationen über das Hersteller-spezifische (Rockwell Proprietary) *NMEA-Format* sowie das ebenfalls spezifische *binary-Format* sind im *Kapitel 3* sowie dem zugehörigen *Anhang* ersichtlich.

6.2 Der GPRMC-Datensatz

Da zur Positionsbestimmung ohne Höhenangabe nur der GPRMC-Datensatz benötigt wird, ist dieser im folgenden ausführlich beschrieben.
Im Kapitel *3* sowie dem zugehörigen *Anhang* sind jedoch sämtliche vom GPS-Empfänger gelieferte NMEA-Datensätze ebenso im Detail ersichtlich.

Recommended Minimum Specific GPS Data (RMC)

Beinhaltet Zeit-, Datums-, Positions-, Kurs- und Geschwindigkeitsdaten. Die Felder dieser Nachricht enthalten auch Daten, wenn der Empfänger nicht navigiert. Dies erlaubt Benutzer-initialisierte, gespeicherte oder Standard-Werte anzuzeigen, bevor (gültige) Daten erhältlich sind.
Ein gültiger Datensatz enthält die Daten in der folgenden Form:

RMC Message
Tabelle III-6, Anhang 3.8.4

```
$__RMC,hhmmss,a,IIII.II,a,yyyyy.yy,a,x.x,x.x,xxxxxx,x.x,a*hh
```

```
$GPRMC,155203,A,4723.1785,N,00810.0215,E,0.000,121.7,280798,13.8,W*35
```

Man erkennt, dass die Anzahl der Kommastellen gewisser Felder varieren kann. Das GPS-Empfänger-Modul liefert z.B. die Längen-und Breitengrade wie in obigem Beispiel mit 4 Nachkommastellen (4723.1785). Zu beachten ist auch, dass Positionsdaten stets in Grad, Minuten, zehntel und hundertstel Minuten (nicht in Grad, Minuten und Sekunden) ausgegeben werden.

Bedeutung der einzelnen Datenfelder an einem Beispiel:

`$GPRMC,155203,A,4723.1785,N,00810.0215,E,0.000,121.7,280798,13.8,W*35`

`$GPRMC`	GP steht für das sendende Gerät (Global Positioning R.) RMC ist der Satz-Identifier		
`155203`	UTC-Zeit in Stunden, Minuten und Sekunden	(HHMMSS)	15 : 52 : 03
`A`	Datenstatus	(A = Daten gültig, V = Daten ungültig)	
`4723.1785`	Aktueller Breitengrad	(GGMM.mmmm)	47° 23.17
`N`	N = nördliche, S = südliche Breite	(Latitude)	N
`00810.0215`	Aktueller Längengrad	(GGGMM.mmmm)	08° 10.02
`E`	E = östliche, W = westliche Länge	(Longitude)	E
`0.000`	Geschwindigkeit über Grund in Knoten	(0.0k)	
`121.7`	Aktueller Kurs über Grund in Grad	(121.7°)	
`280798`	Aktuelles Datum in Tag, Monat und Jahr	(DDMMYY)	28 : 07 : 98
`13.8`	Magnetische variation in Grad	(13.8°)	
`W`	Magn. Richtungsabweichung	(W = westlich, E = östlich)	
`*35`	Ende der Daten (*), Checksumme (35)	(*35)	

6.2.1 Auszuwertende Daten

Die softwaremässig auszuwertenden Daten sind in obiger Auflistung in der rechten Spalte dargestellt und beschränken sich auf

- Breiten- und Längengrad, die UTC-Zeit und das Datum.

6.2.1.1 Darstellungsform

Die Darstellungsform wurde aufgrund des Platzbedarfs des Displays (max. 20 Zeichen) gemäss den im Kapitel *Konzept* aufgezeigten Display-Bildern gewählt und entspricht aufgrund der software-kompatibilität der von D. Läubli. Die Rohdaten müssen also vor der Ausgabe am Display in die richtige Darstellungsform gebracht werden.
Dazu musste ich die Filter-Routine von D. Läubli wie weiter hinten beschrieben um einige Zeilen erweitern, da ich die Längen- und Breitengrade mit 4 Nachkommastellen erhielt und somit die 20 Zeichen der Anzeige nicht ausgereicht hätten.

6.3 Schweizer Landeskoordinaten

Das GPS-Empfänger-Modul liefert geographische Koordinaten in Form von Längen - und Breitengraden, welche zur Darstellung in Schweizer-Landeskoordinaten entsprechend umgerechnet werden müssen.

Die Umrechnung erfolgt mit einer C-Routine.

Die Berechnungs-Routinen für die Umrechnung der vom GPS-Empfänger gelieferten geographischen Koordinaten in Schweizer-Landeskoordinaten fand ich in der Diplomarbeit von D. Läubli [9].

Damit dieser Umrechnungs-Algorithmus auch funktioniert, muss das entsprechende Ursprungs-Koordinatensystem (Map Datum, siehe [6]) am GPS-Empfänger eingestellt werden (siehe *Kapitel 3*).
In diesem Fall ist das Koordinatensystem mit der Bezeichnung European 1979 (Europa 1979) zu verwenden. Dieses basiert auf dem Referenz-Ellipsoid mit der Bezeichnung “International” und wird durch Angabe des Codes **62** am GPS-Empfänger eingestellt (siehe *Kapitel 3*).

Modifizierung der C-Berechnungs-Routine

Aufgrund des fehlenden $PGRMZ-Datensatzes (siehe [9]) wird die Höhe (Altitude) nicht ausgewertet. Dies erforderte eine geringfügige Modifizierung der C-Berechnungs-Routinen, welche ich aus der erwähnten Diplomarbeit übernehmen konnte.

6.3.1 Umrechnungs-Formeln

Gegebene Werte

Geographische Koordinaten in Länge (λ) und Breite (β) in Grad mit Dezimalen

Gesuchte Werte

Schweizer Landeskoordinaten X und Y in Metern

Konstanten

a=1.000729 b=7.439587 c=46.952195 d=46.907731 f=0.998438 h=6378.816

$$k = a \cdot (\lambda - b)$$
$$m = \beta - c$$
$$n = d + f \cdot m + g \cdot m^2$$
$$p = \cos(d) \cdot \sin(n) - \sin(d) \cdot \cos(n) \cdot \cos(k)$$
$$q = \arcsin(p)$$
$$r = \tan\left(\frac{q}{2} + 45\right)$$
$$s = \left(\frac{\cos(n) \cdot \sin(k)}{\cos(q)}\right)$$
$$t = \arcsin(s)$$
$$x = 600 + \frac{\pi \cdot h \cdot t}{180}$$
$$y = 200 + h \cdot \ln(r)$$

6.4 Programm

Dieser Abschnitt beschreibt das definitive Software-Konzept und dessen Ausarbeitung. Im ersten Teil wird auf das Prinzip einer “ereignisgesteuerten” Programm-Architektur eingegangen, welche das Grundgerüst für eine modulare Software-Entwicklung darstellt. Anschliessend wird anhand einiger Programm-Auszüge dargestellt, wie der Lösungsansatz verwirklicht wurde.

6.4.1 Einleitung

Die Eigenschaft von umfangreichen Assembler-Programmen, relativ schnell unübersichtlich zu werden (Spaghetti-Code!), veranlasste mich, das Programm nach dem oben erwähnten Prinzip zu strukturieren.
Indem ich für jeden wiederkehrenden Ablauf eine eigene Subroutine schrieb, konnte ich diese Module wie Prozeduren handhaben, die ich zum Teil mit vorgängig zu übergebenden Parametern nur noch aufrufen musste.
Obwohl zu Beginn ein zeitlich enormer Aufwand, zahlte sich dieses Vorgehen innert Kürze aus. Vor allem das Steuern des Displays bereitete keine Schwierigkeiten mehr, nachdem sämtliche benötigten Steuerbefehle einmal in eine eigene Subroutine “verpackt” waren.

Bem.:
Mit Modul ist im folgenden ein Unterprogramm in Form einer Subroutine gemeint.

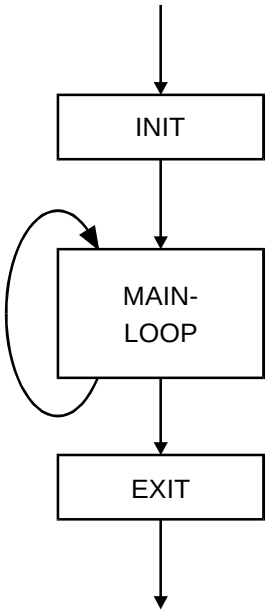
Vorwegzunehmen ist, dass ich aufgrund der Einbindung von fremden Software-Modulen häufig zwischen mehreren Konzepten hin- und hergerissen wurde, bis ich mich mit einer “zufriedenstellenden Kompromiss-Lösung” zufrieden gab.

“Kompromiss-Lösung”

6.4.2 Programm-Architektur

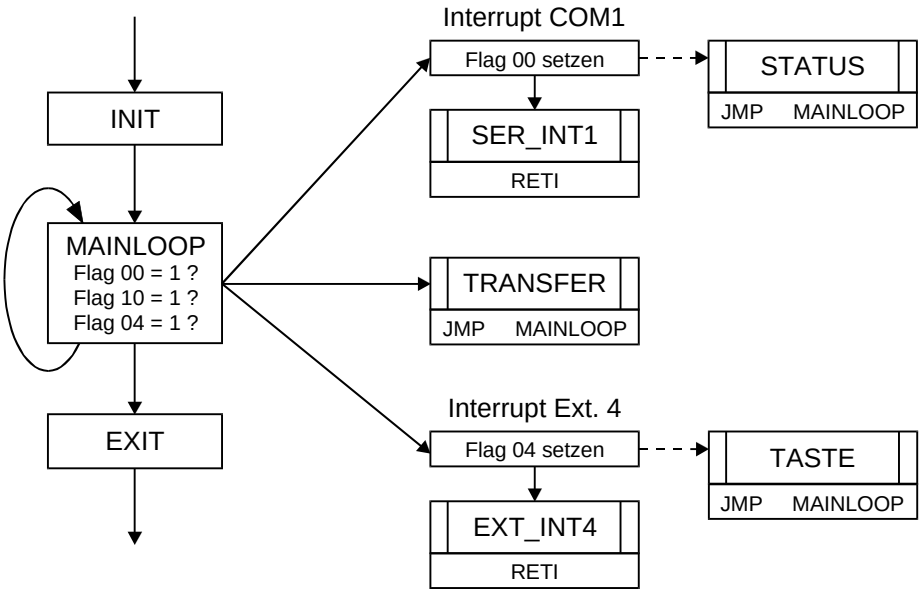
Die Idee eines “ereignisgesteuerten” Programmes ist im Gegensatz zu der etwas hochgestochen klingenden Bezeichnung relativ simpel:

Das Hauptprogramm wird in Form einer Endlosschleife programmiert, in der es lediglich abgefragt wird, ob ein Ereignis vorliegt. Ist dies der Fall, verzweigt das Programm bei der nächsten Gelegenheit in die entsprechende Subroutine und kehrt wieder in die sogenannte “MAINLOOP” (Endlosschleife) zurück.
Das Prinzip ist vergleichbar mit einem Interrupt, der ja ebenfalls durch ein Ereignis ausgelöst wird, entsprechend reagiert und wieder in “Wartestellung” geht.
Somit gibt es zwei verschiedene Arten von Ereignissen, die benutzergesteuerten und die systemgesteuerten, womit in meinem Fall eine Verschachtelung entsteht:
Nehmen wir das Beispiel des Interrupt-Handlers, welcher bei einer betätigten Taste (⇨ ext. Interrupt 4 wird ausgelöst) innerhalb der Interrupt-Routine ein Flag setzt, welches wiederum dem Hauptprogramm signalisiert, dass ein Ereignis stattgefunden hat. Dieses reagiert entsprechend, setzt das Flag zurück und ist bereit, um auf das nächste Ereignis zu reagieren.



6.4-1 Programm-Architektur

Dieses Prinzip lässt sich wie folgt auf mein Programm übertragen:



Nach der Initialisierung "wartet" das Programm in der Endlosschleife (MAINLOOP), bis ein Ereignis vorliegt. Ist dies der Fall, reagiert es entsprechend und kehrt umgehend in das Hauptprogramm zurück.
Liegt kein Ereignis vor, befindet sich das Programm durch die Endlosschleife immer an einem definierten Ort.

6.4-2 Software-Konzept

6.4.3 Allgemeines zum Source-Code

Der Programm-Kopf des Source-Code-Assembler-Listings enthält die wichtigsten Informationen, um das Programm nachvollziehen zu können. Trotzdem einige Bemerkungen zu dem im *Anhang* ersichtlichen Assembler-Listing:

Programm-Kopf

Gross-Kleinschreibung

Diese ist nicht bewusst entstanden, hat sich jedoch so ergeben, da die reinkopierten Soubroutinen alle kleingeschriegen waren. Da ich bereits einen erheblichen Teil des Codes gross geschrieben hatte, sparte ich mir die Zeit für die eh nur fehlerbehaftete Anpassung. Somit ist als Nebeneffekt ersichtlich, welche Code-Zeilen 1:1 übernommen wurden.

Die übernommenen Soubroutinen sind kleingeschrieben

Struktur

Einige Assembler-Steueranweisungen und die Einbindung der Register-Definitionsdatei sowie der C-Routinen befinden sich ganz am Anfang.

Einbindungen / Deklarationen

Dann folgt die Bestimmung der Interrupt-Vektoren sowie die Definition der am Display auszugebenden Strings, welche mit der DB-Anweisung im CODE-Speicher abgelegt werden. Diese Anweisungen erzeugen keinen Code.

Interrupt-Vektoren
String-Definitionen

Darauf folgen allgemeine Initialisierungen und die Interrupt-Kontrolle.

Initialisierungen

Dem anschliessenden Hauptprogramm folgen zahlreiche Subroutinen.

Hauptprogramm
Subroutinen

Ganz am Schluss befindet sich die Initialisierung des LCD-Displays und sämtliche zur Steuerung benötigte Subroutinen.

LCD-Display-Subroutinen

Mehrere Datenpointer

Was ich ebenfalls kurz erwähnen möchte, ist die Verwendung von mehreren Datenpointern, was ich vor allem bei der Arbeit mit dem LCD-Display ausnutzen konnte. Im Modul "Allgemeine Initialisierungen" ist gut ersichtlich, dass ich für jede der vom LCD-Controller benötigte Adresse (siehe Kap. *5*) wie auch für die Tastatur einen eigenen Datenpointer setze. Diese werden bei Bedarf aufgerufen und stellen sicher, dass die richtige Adresse selektiert wird.

Bei mehreren Chip-Select-Adressen empfiehlt sich das Arbeiten mit mehreren Datenpointern.

Flags

Ebenfalls nützlich sind die bitadressierbaren Bytes von 20H bis 2FH im Lower-RAM-Bereich, von denen ich ausgedehnten Gebrauch machte. Bei der bitweisen Adressierung besitzt in diesem Bereich jedes einzelne Bit eine eigene Adresse, welche beim ersten Bit an der Adresse 20H mit 00 beginnt.

Flags sind einzel-bitadressierbare Bits mit eigener Adresse.

6.5 Programm-Module

Da die Programm-Listings ausreichend dokumentiert sind, gehe ich im weiteren nur auf die wichtigsten Funktionen und die Programm-Fragmente ein, welche zum besseren Verständnis notwendig sind.

6.5.1 Interrupt serielle Schnittstelle COM1

Dieser wird durch das Interrupt-Anforderungs-Flag RI1 ausgelöst, welches per Software zurückgesetzt werden muss und mit dem Stopp-Bit des empfangenen Frames gesetzt wird. Unmittelbar nach dem Einsprung in die Interrupt-Vektor-Adresse wird das Flag 00 gesetzt und zu der von D. Läubli übernommenen Filter-Routine gesprungen (SER_INT1:). Diese wird komplett innerhalb der Interrupt-Routine abgearbeitet und hat die folgenden Funktionen:

Flag: 00
Label der Soubroutinen:
[Untenstehend jeweils die Angabe der Labels im Source-Code und ein Verweis auf die benützten Flags]

Die empfangenen ASCII-Zeichen werden zur optischen Kontrolle jeweils an Port 4 ausgegeben und so lange eingelesen, bis der Datensatz $GPRMC vollständig erkannt wird. Ist dies der Fall, werden die nun empfangenen Daten an das Modul FILTERN übergeben.

SER_INT1:

Dieses zählt die Kommas und entscheidet aufgrund der Anzahl, ob die weiteren Daten verworfen oder abgespeichert werden sollen. Diesen Teil musste ich so anpassen, dass nach dem Erkennen des 3. bzw. 5. Kommas nur die erforderlichen 2 Nachkommastellen eingelesen und alle weiteren ignoriert werden. Dazu wird das Register R1 bei jedem Durchgang inkrementiert und mit dem Konstanten-Wert 8 bzw. 9 verglichen. Damit alle weiteren Zeichen ignoriert werden, muss das Register erneut mit dem Wert KONSTANTE-1 geladen werden. Da ein komplett eingelesener Datensatz nicht wie üblich mit den End-Zeichen <CR><LF>, sondern mit dem 10.Komma erkannt wird, musste ich an dieser Stelle ein Flag (05) setzen, um dies dem Hauptprogramm mitteilen zu können. Zudem wird das Flag 01 gelöscht, damit die Abspeicherung des nächsten Datensatzes erneut bei 0F800H beginnt und die Daten aktualisiert werden.

FILTERN:

(INC_R1:)

(Flag 05)
(Flag 01)

An dieser Stelle werden die angenommenen Daten im XRAM-Bereich ab Adresse 0F800H abgespeichert. Das zuvor erwähnte Flag (01) wird beim ersten abzuspeichernden Zeichen gesetzt und mit dem 10.Komma rückgesetzt.

SCHREIBEN:
(Flag 01)

Diese kleine Routine löscht nach einem vollständig abgespeicherten Datensatz die Register 0-4 der Registerbank 0. Anschliessend wird das Interrupt-Anforderungs-Flag RI1 zurückgesetz und in das Hauptprogramm gesprungen.

ZERO_4R:

6.5.2 Externer Interrupt 4

Dieser wird durch das Interrupt-Anforderungs-Flag IEX4 ausgelöst, welches von der Software automatisch zurückgesetzt und mit der Erkennung einer positiven Flanke am Port-Pin 1.1 gesetzt wird. In diesem Fall geschieht das mit dem DA-Signal des Key-Encoders, nachdem dieser eine betätigte Taste korrekt erkannt hat. Unmittelbar nach dem Einsprung in die Interrupt-Vektor-Adresse, noch bevor das Flag 04 gesetzt und zum entsprechenden Interrupt-Handler (EXT_INT_4:) gesprungen wird, erfolgt die Abfrage, ob das Flag 11 gesetzt ist. Wenn JA, wird ein weiteres Flag (10) gelöscht und die Interrupt-Routine sofort wieder verlassen.
Der Grund hierfür ist die Subroutine TRANSFER (siehe weiter hinten).

Flag 04:
Label der Soubroutinen:
(Flag 11)
(Flag 10)

Im Normalfall wird direkt zum Interrupt-Handler gesprungen. Dieser tut nichts weiter, als den auf dem Datenbus liegenden Tastencode in das Register R6 zu verschieben. Anschliessend erfolgt der Sprung ins Hauptprogramm.

EXT_INT4:

6.5.3 Hauptprogramm (MAINLOOP)

Dieses beginnt mit dem Label BEGIN: und setzt als erstes das globale Interrupt-Freigabe-Bit EAL. Danach folgt eine Schleife, die erst dann verlassen wird, wenn das Flag 00 gesetzt, also der Interrupt der seriellen Schnittstelle COM1 ausgelöst wird. Wird kein Interrupt ausgelöst, wird beim ersten Durchlauf das Flag 08 gesetzt und die Meldung
'KEINE VERBINDUNG ZU GPS-EMPFAENGER!'
auf dem Display ausgegeben.

BEGIN:
WAIT:
(Flag 00)
(Flag 08)

Ist der GPS-Empfänger mit der Schnittstelle verbunden und in Betrieb, wird direkt nach dem setzen des Bits EAL zum Label MAINLOOP:, also in die Endlosschleife gesprungen. Diese wird nur noch verlassen, wenn:

MAINLOOP:

- eine Taste betätigt, — *Flag 04 = 1 ⇨JMP TASTE*
- das 10.Komma erkannt oder — *Flag 05 = 1 ⇨JMP STATUS*
- die Soubroutine TRANSFER gestartet worden ist — *Flag 10 = 1 ⇨JMP COM1_0*

In den ersten beiden Fällen wird das abgefragte Flag bereits vor dem Sprung in die entsprechende Subroutine wieder gelöscht. Das Flag 10 wird innerhalb der Interrupt-Routine des ext. Interrupt 4 zurückgesetzt (siehe oben).

6.5.4 Abfrage Tasten (TASTE)

Der im Register R6 zwischengespeicherte Tastencode wird in den Akku verschoben, das High-Nibble gelöscht und der 4-Bit-Tastencode zur optischen Kontrolle an Port 4 ausgegeben. Die Abfrage erfolgt ganz banal durch Vergleich des aktuellen Tastencodes mit dem jeder Taste zugewiesenen Konstanten-Wert. Ist der Vergleich true, werden die folgenden Zeilen der zugehörigen Taste abgearbeitet, ist er false, wird zur nächsten Taste gesprungen. *TASTEN:*

Die folgenden Tasten sind implementiert:

- Die beiden Flags 09 und 03 werden immer zurückgesetzt.
  ⇨ Bewirkt die Anzeige der L/B-Koordinaten mit UTC-Zeit — *TASTE_1:* *Flag 03=0* *Flag 09=0*
- Flag 09 wird gesetzt, Flag 03 wird zurückgesetzt.
  ⇨ Bewirkt die Anzeige der X/Y-Koordinaten mit Sommerzeit — *TASTE_2:* *Flag 03=0* *Flag 09=1*
- Die beiden Flags 09 und 03 werden immer gesetzt.
  ⇨ Bewirkt die Anzeige derX/Y-Koordinaten mit Winterzeit — *TASTE_3:* *Flag 03=1* *Flag 09=1*
- Flag 02 wird komplementiert und je nach vorherigem Zustand das Interrupt-Freigabe-Bit ES1 der seriellen Schnittstelle COM1 gesetzt oder gelöscht. — *TASTE_4:* *Flag 02 wird komplementiert*
- Die Initialisierungs-Routine für das LCD-Display wird aufgerufen. — *TASTE_0:* *LCD_INIT:*
- Der RAM-Bereich des Display-Abbildes wird mit Space überschrieben und die Reset-Routine des LCD-Displays aufgerufen — *TASTE_C:* *LCD_RESET:*
- Das Flag 10 wird gesetzt. Somit wird die Subroutine TRANSFE: zur Übertragung des NMEA-Protokolles auf einen PC nach dem Rücksprung ins Hauptprogramm gestartet. — *TASTE_F:* *(Flag 10)* *TRANSFER:*

Bei allen anderen Tasten wird die Subroutine TXT_LCD_1 aufgerufen, was die Anzeige der Meldung ‘ERROR: TASTE OHNE FUNKTION!’ bewirkt. *TXT_LCD_1:*

6.5.5 Abfrage Daten gültig (STATUS)

Der Datenpointer wird mit der Adresse geladen, an dessen Speicherstelle die Information über den Datenstatus gespeichert ist und mit dem ASCII-Zeichen 41H (‘A’) verglichen (zur Erinnerung: ‘A’ = gültig, ‘V’= ungültig). Ist der Status ungültig, wird die Meldung
‘POSITIONS-BESTIMMUNG NICHT MOEGLICH!’
ausgegeben und zurück ins Hauptprogramm gesprungen. *STATUS:*

Ist der Status gültig, wird vorbereitend das Interrupt-Freigabe-Bit ES1 gelöscht und das Flag 06 gesetzt. Danach wird die Routine L_B_DAT aufgerufen, in welcher die Rohdaten aufbereitet (Darstellungsform!) und zusammen mit den definierten Text-Strings in den Display-Abbild-XRAM-Bereich kopiert werden. Ist dies erledigt, wird mit der Routine TXT_LCD dieser gesamte Speicherbereich auf das Display-RAM abgebildet und somit zur Anzeige gebracht. Das Bit ES1 wird wieder gesetzt und zurück ins Hauptprogramm gesprungen. *(Flag 06)* *L_B_DAT:* *LCD_OUT:* *TXT_LCD*

6.5.5.1 Aufbereiten der Rohdaten (L_B_DAT)

L_B_DAT:

(Flag 03, Flag 09)

Diese Routine, welche nach einem gültigen Status aufgerufen wird, erscheint mir ebenfalls erwähnenswert, da sie die eingebundenen C-Routinen enthält. Diese werden jedoch nur aufgerufen, wenn die Flags 03 und 09 durch Betätigung der Tasten 2 oder 3 entsprechend gesetzt wurden. In diesen Software-Teil musste ich einige Zeit investieren bis mir klar war, an welcher Stelle die C-Routinen aufgerufen werden dürfen oder besser: müssen.

Die C-Routinen

Die Begründung ist relativ simpel: Da zur Berechnung der Schweizer-Landeskoordinaten ja erst Rohdaten in Form der geographischen Koordinaten vorliegen müssen, sind diese in entsprechender Form bereitzustellen.
Die Berechnungs-Routinen sind so programmiert, dass jeder zur Berechnung benötigte Wert (L/B-Koordinaten) seine feste Speicher-Adresse im XRAM besitzt. Diese Deklarationen sind auf der ersten Seite des C-Listings gut erkennbar und eine gute Idee, wenn man genau weiss, wo sich die Rohdaten befinden. Trotz diesem Wissen hatte ich an genau dieser Stelle Mühe, das Programm von D. Läubli nachzuvollziehen, was mich etliche Zeit kostete.

Berechnung der Schweizer- Landeskoordinaten

X_Y_DAT:
“umrechnung_schweiz”
(Flag 06)

Doch zurück zum Programm: Die Berechnung der X/Y-Koordinaten wird mit dem Aufruf der C-Routine “umrechnung_schweiz” gestartet. Zuvor wird das Bit 06 gesetzt, welches die Ausgabe des Speicherbereichs bewirkt, in welchem sich die Resultate in Form der X- und Y-Koordinaten befinden. Zudem müssen die Flag-Register 20H und 21H mittels PUSH-Befehl gerettet werden, da die C-Routine das interne RAM zur Zwischenspeicherung von temporären Daten benötigt (dies ist jedenfalls meine Erfahrung ...). Aus demselben Grund wird vor dem Aufruf auch ein noch freier Datenpointer selektiert. Der Rücksprung beendet gleichzeitig die Routine L_B_DAT.

Berechnung der Sommer-/Winterzeit

X_Y_S: Flag 03=0
“sommerzeit”
X_Y_W: Flag 03=1
“winterzeit”

Die Routine beginnt bei dem nebenstehendem Label und fragt als erstes das Flag 03 ab, ob die Sommer- oder Winterzeit gewünscht ist. In beiden Fällen wird anschliessend der entsprechende Text-String in den dafür vorgesehenen XRAM-Bereich kopiert und die Berechnungs-Routine aufgerufen. Danach wird die Routine L_B_DAT fortgesetzt.

6.5.6 Verbindung COM1 ⇨ COM0 (TRANSFER)

Diese Routine ermöglicht im Polling-Verfahren die direkte “Durchschlaufung” von Daten von COM1 zu COM0.

Diese Routine ermöglicht die “Durchschlaufung” des an der Schnittstelle COM1 empfangenen NMEA-Potokolles zur Schnittstelle COM0, welche die ASCII-Zeichen umgehend wieder ausgibt. Sinn und Zweck dabei ist, das empfangene NMEA-Protokoll ohne Umstecken der Verbindungskabel auf einem PC-Monitor zu visualisieren und bei Bedarf aufzuzeichnen.

Terminalprogramm

Dazu muss lediglich das zum herunterladen des HEX-Files benötigte Programm (MON51 od. TS51) korrekt beendet und ein geeignetes Terminalprogramm gestartet werden. Wenn mehr Informationen über das empfangene NMEA-Protokoll oder den Zustand des Empfängers gewünscht sind, kann dies auch mit der LABMON-Evaluation-Software getan werden (siehe *Kapitel 3*).

LABMON-Evaluation-Software

TRANSFER:

Da diese Routine im Polling-Verfahren arbeitet, werden die Interrupts beider Schnittstellen mit dem Rücksetzen der Freigabe-Bits ES1 bzw. ES0 gesperrt. Das Flag 11 wird gesetzt und die Schnittstelle COM0 für eine asynchrone Datenübertragung mit 4800 Baud und 10 Bit (Mode 1) vorbereitet.

(Flag 11)

LOOP:

Dann folgt der Eintritt in die Schleife LOOP, welche die Zeichen vom SFR S1BUF über das Register R1 in das SFR S0BUF verschiebt. Da die Interrupts deaktiviert sind, müssen die Bits RI1 bzw. TI0 jeweils so lange abgefragt werden, bis das Frame ganz empfangen oder fertig gesendet wurde. Die Schleife ist so lange aktiv, bis das Flag 10 zurückgesetzt wird. Wie bereits erwähnt, ist das bei der Betätigung einer beliebigen Taste der Fall. Sobald die Schleife verlassen wird, wird auch das Flag 11 wieder zurückgesetzt. Damit weitere Eingriffe mit dem Monitor-Programm MON51 oder anderen Emulatoren möglich sind, muss die Schnittstelle COM0 wieder in ihren Ursprungszustand versetzt werden. Die Freigabe-Bits ES1 bzw. ES0 werden gesetzt, somit die Interrupts wieder freigegeben und die Routine verlassen.

REC:
SEND_1:
(Flag 10)

(Flag 11)

6.5.7 “allgemeine” Subroutinen

Subroutinen

Die jetzt nicht erwähnten Subroutinen sind für das Verständis des Programmes nicht relevant und erledigen häufig wiederkehrende “Arbeiten” wie z.B. das Kopieren eines definierten Strings aus dem CODE- in den XRAM-Bereich oder die Abfrage des Busy-Flags, welches den LCD-Controller auf seine Bereitschaft überprüft. Die Funktion der jeweiligen Routine sollte bei gegebenem Interesse anhand des Assembler-Listings erkennbar sein.
Auf die C-Routinen werde ich aus diesem Grund ebenfalls nicht näher eingehen, da man Subroutinen oder in diesem Fall Module ja genau deshalb schreibt, um diese als “Black Box” betrachtend bei Bedarf (mit ev. zu übergebenden Parametern) aufzurufen ... !
Zudem gibt es sicherlich Leser, die anstelle dieser ganzen textlichen Abhandlung schon lange lieber ein Flussdiagramm hätten ... Ich hoffe, dass ich die angesprochenen Leser mit den graphischen Darstellungen am Ende dieses Kapitels zufriedenstellen kann.
Zuerst jedoch noch einige Informationen zu der Datenablage im RAM:

6.6 Datenablage im RAM-Speicher

Datenablage

Aufgrund der bereits erwähnten Zuweisung von definierten Speicheradressen auf Variablen und der Display-Daten, wird in diesem Abschnitt auf die Ablage der Daten im RAM bzw. im XRAM eingegangen.

Die Daten, welche nach einem erkannten $GPRMC-Datensatz im XRAM abgespeichert werden, stehen als “Rohdaten” für die weitere Berechnungen zur Verfügung. Die Zwischenspeicherung der bereits in die richtige Darstellung gebrachten Daten wird im internen RAM vorgenommen, bevor diese zur Ausgabe auf dem Display erneut in einen dafür reservierten Speicherbereich ins XRAM kopiert werden müssen.

Die Daten werden vom XRAM ins interne RAM und von da erneut in einen reservierten XRAM-Speicherbereich kopiert.

Datenablage im XRAM nach Datenfilterung **XRAM**

```
Memory
Addr    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 0123456789ABCDEF
X:F800: 31 30 34 34 30 36 56 34 37 32 31 2E 36 39 4E 30 104406V4721.69N0
X:F810: 30 38 30 39 2E 38 31 45 30 33 30 38 39 38 F7 FF 0809.81E030898..
X:F820: 08 00 F7 FF 08 00 F7 FF 08 00 F7 FF 08 00 F7 FF ................
```

Zeit UTC:	F800H-F805H	Länge:	F80FH-F816H	*Rohdaten*
Status:	F806H	E/W:	F817H	
Breite:	F807H-F80D	Datum:	F818H-F81D	
N/S:	F80EH			

Datenablage im internen RAM **RAM**

```
Memory
Addr    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 0123456789ABCDEF
I:0080: FE FB FF FF FF FF FF FF FF FE FF FB FF FF DF FF ................
I:0090: 31 30 34 34 30 36 FF 34 37 DF 32 31 2E 36 39 4E 104406.47.21.69N
I:00A0: 20 30 30 38 DF 30 39 2E 38 31 45 FF FF FF FF FB  008.09.81E.....
I:00B0: 30 33 30 38 39 38 FF FF F7 FF FF FF FE EF FF FF 030898..........
```

Zeit UTC:	90H-95H	Breite / Länge:	97H-AAH	*Temporäre Daten*
Datum:	B0H-B5H			

Display-Abbild-Bereich im XRAM **XRAM**

```
Memory
Addr    00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 0123456789ABCDEF
X:F900: 20 20 4C 2F 42 2D 4B 4F 4F 52 44 49 4E 41 54 45   L/B-KOORDINATE
X:F910: 4E 3A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 N:
X:F920: 20 54 49 4D 45 20 55 54 43 3A 20 31 30 3A 34 32  TIME UTC: 10:42
X:F930: 3A 32 35 20 20 20 20 20 20 20 20 20 20 20 20 20 :25
X:F940: 34 37 DF 32 31 2E 36 34 4E 20 30 30 38 DF 30 39 47.21.64N 008.09
X:F950: 2E 38 33 45 20 20 20 20 20 20 20 20 20 20 20 20 .83E
X:F960: 20 44 41 54 55 4D 3A 20 20 20 20 30 33 3A 30 38  DATUM:    03:08
X:F970: 3A 39 38 20 20 20 20 20 20 20 20 20 20 20 20 20 :98
X:F980: 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
X:F990: 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 FF                .
X:F9A0: 08 00 F7 FF 08 00 F7 FF 08 00 F7 FF 08 00 F7 FF ................
```

```
0123456789ABCDEF
  X/Y-KOORDINATE
N:
 SZEIT CH: 02:50
:47
47.21.55N 008.09
.82E
 DATUM:    03:08
:98
X=000000m  Y=000
000m           .
................
```

JederZeile des Displays sind 20 Speicherstellen zugewiesen: *Ausgabe-Daten*

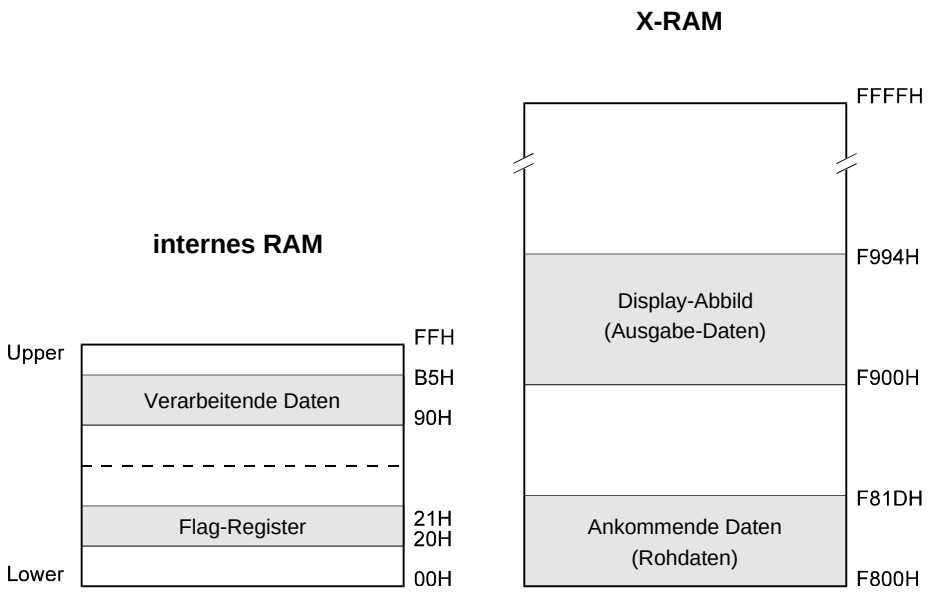
Zeile 1:	F900H-F913H			Überschrift	`L/B-KOORDINATEN:`
Zeile 2:	F940H-F953H	oder	F980H-F993H	L/B- oder X/Y-Werte	`47°23.17N 08° 10.02E`
Zeile 3:	F960H-F973H			Datum	`DATUM: 28:07:98`
Zeile 4:	F920H-F933H			Zeit	`TIME UTC: 15:52:03`

Der gesamte Bereich erstreckt sich also von F900H bis F993H. Dieser Bereich wird zu Beginn des Programmes mit ‘Space’ (20H) beschrieben. Ebenso kann dies auf Wunsch durch Betätigung der Taste C (Clear) getan werden.

Taste C (Clear)

6.6.1 Memory-Map

Überblick über die Speicheraufteilung:



Die ankommenden Rohdaten werden ab F800H gespeichert, im internen RAM verarbeitet und zur Ausgabe auf das Display in einen dafür vorgesehenen Bereich im XRAM kopiert.

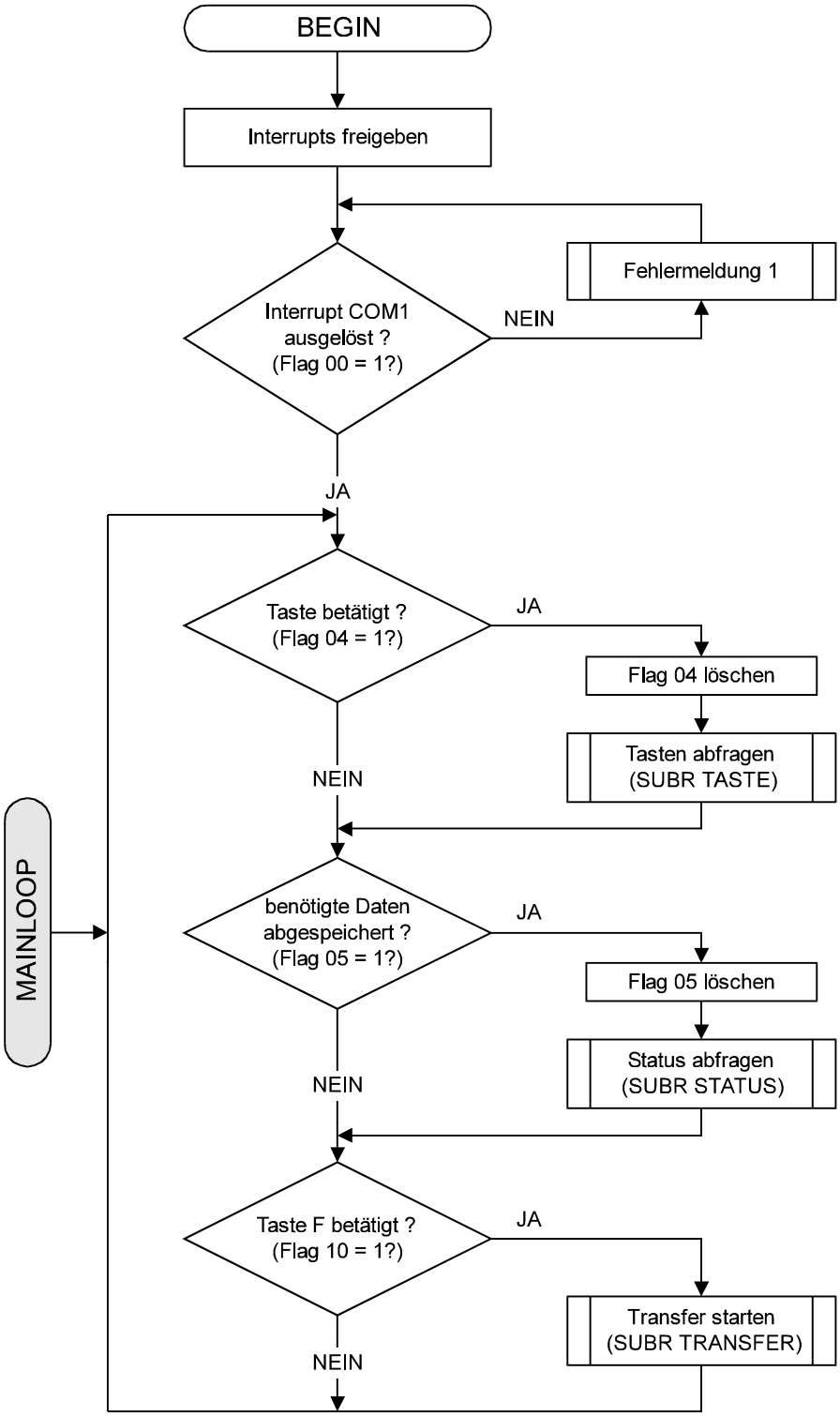
6.6-1 Memory Map

Flag-Register	20H und 21H
Verarbeitende Daten	90H - B5H
Ankommende Daten (Rohdaten)	F800H- F81DH
Display-Abbild (Ausgabe-Daten)	F900H - F994H

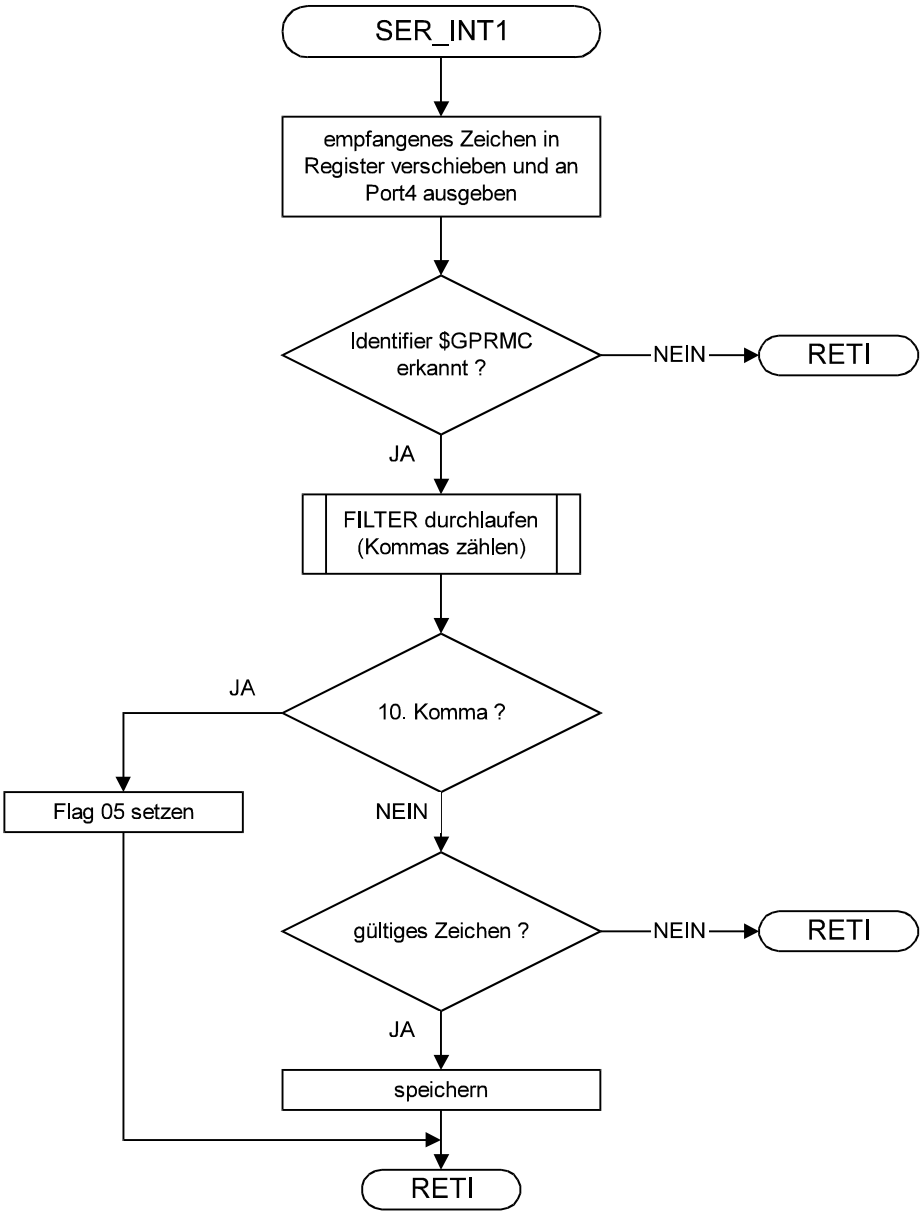
6.7 Flussdiagramme

Die Flussdiagramme beschränken sich auf die Hauptfunktionen des Programmes und stützen sich auf das in Abbildung 6.4-2 aufgezeigte Konzept.

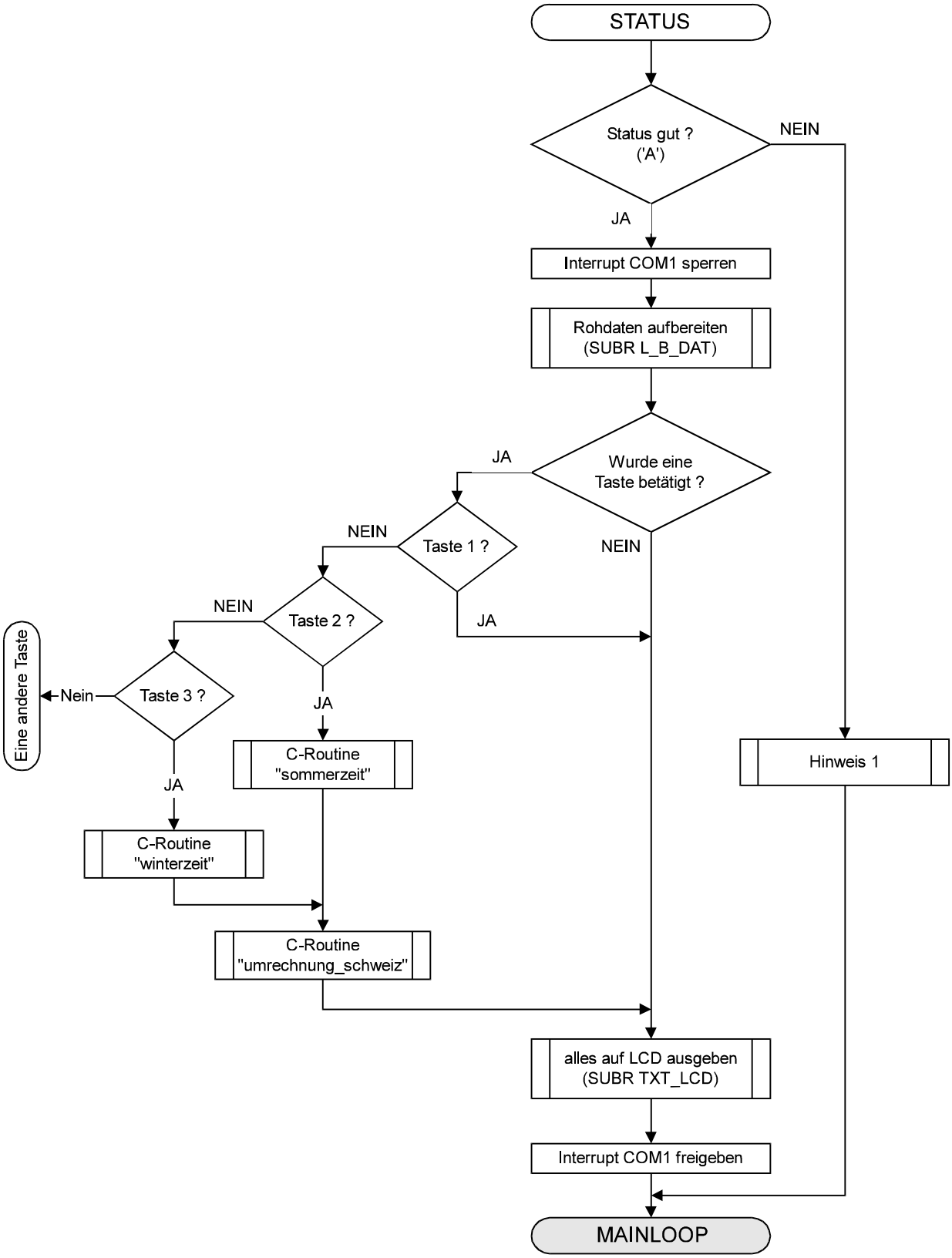
Hauptprogramm



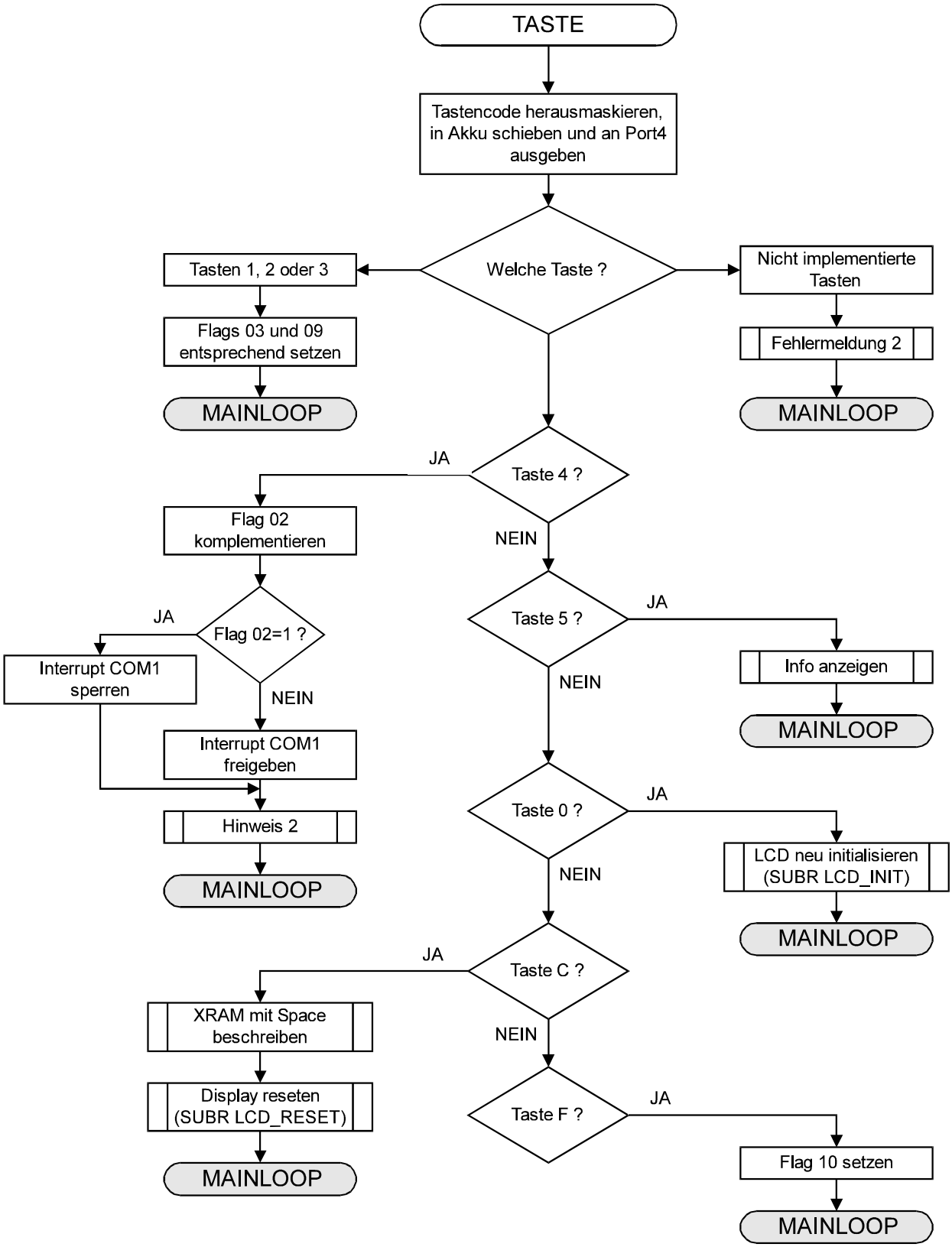
Interrupt-Handler serielle Schnittstelle COM1



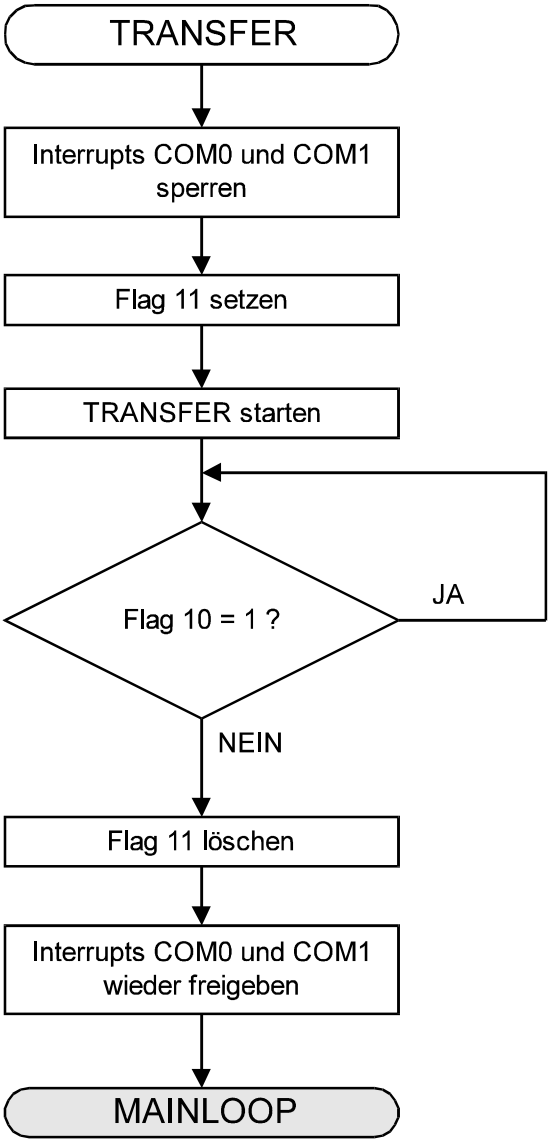
Abfrage Daten gültig (STATUS)



Abfrage Tasten (TASTE)



Visualisierung NMEA-Protokoll (TRANSFER)



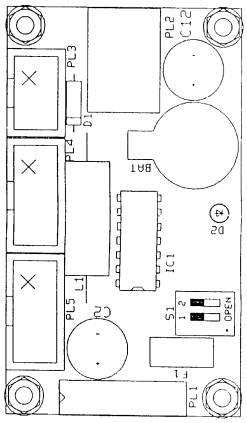
7. Inbetriebnahme

Dieses Kapitel geht kurz auf die Inbetriebnahme der Hardware-Komponenten-ein, bevor die Bedienung der fertigen Einrichtung in Form einer Bedienungs-anleitung erklärt wird.

7.1 der Hardware-Komponenten

7.1.1 GPS-Empfänger-Modul

GPS-Empfänger-Modul



Dieses nahm ich in den ersten Wochen mit einem PC in Betrieb, da ich die Funktionen der LABMON-Software und natürlich des Empfänger nicht nur beschreiben (Übersetzung !), sondern auch testen wollte.
Die Installation sowie die Ausführung der LABMON-Software bereitete auf keinem der Systeme (Laptop 286er mit DOS 3.3, 486er DX mit Win3.11, P233 MMX mit Win95) Schwierigkeiten, da es eine reine DOS-Applikation ist. Einzig unter Win3.11 musste ich das Programm im DOS-Modus (nicht DOS-BOX) starten, da mir unter Windows die Maus abstürzte und das Programm einfror. Doch dies kann auch nur an der Konfiguration meines Rechners liegen ...
Weitere Bemerkungen zu der Arbeit mit dem GPS-Empfänger-Modul (Probleme und deren Behebung) sind im Kapitel *3* unter im Abschnitt *Allg. Hinweise und Bemerkungen* ersichtlich.

7.1.2 Spannungsversorgung

Spannungsversorgung

die Inbetriebnahme der Spannungsversorgung (Print 1) beschränkte sich auf das Überprüfen und ausmessen der generierten Spannungen unter Belastung. Dazu schloss ich das Prototypen-Board sowie das GPS-Empfänger-Modul an den vorgesehenen Prüfklemmen (X8) an, schaltete ein Ampère-Meter dazwischen und konnte nach dem nachträglichen Einlöten einer fehlenden Verbindung die folgenden Werte messen:

U [V]	**I [A]**	**Last**	***Messobjekt:***
17.73		ohne	
13.1	0.239	MCB 517AC	**T1:**
17		ohne	*2x12V DC*
14.4	0.115	GPS-Empfänger	
10.5		ohne	**T2:**
8.62	0.220	LCD-Backlight	*1x9V DC*
5.1		Logikspannung	*1x5V DC*

Die gemessenen Werte erfüllen die Vorgaben, wobei die “12V”-Spannungen um 1-2V grösser sind. Da die mit diesen Spannungen versorgten Geräte jedoch über einen internen Spannungsregler verfügen, befinden sich die Werte im erlaubten Bereich.

Bemerkung

Obwohl für die Hintergrundbeleuchtung des LCD-Displays eine Anoden/Kathoden-Spannung von nur ca. 4V benötigt wird (nach Datenblatt), wurde ein Transformator mit höherer Ausgangsspannung verwendet, da dieser vorhanden war. Somit besteht auch die Möglichkeit, weitere Geräte zu speisen.

7.2 Adressdecodierschaltung

Adressdecodierschaltung

Auch hier gab es ausser den gewrapten Verbindungen (Print 2) nicht viel zu testen, da die Schaltung ja bereits auf der Experimentierplatte ihren Dienst verrichtete.
Trotzdem setzte ich die nun fast fertige Hardware komplett zusammen. Der KO zeigte an den erforderlichen Pins eine Logikspannung von 5.1V an. Ich überprüfte so gut es ging noch einmal die Verbindungen, bevor ich erwartungsvoll ein kleines Testprogramm startete, welches das LCD-Display initialiseren und die Ausgabe einiger Zeichen durch Betätigung diverser Tasten ermöglichen sollte. Aber eben- nur sollte ! Ich bekam zwar Zeichen auf dem Display zu sehen (soweit sogut), jedoch das gleiche Zeichen bei Betätigung verschiedener Tasten. Es schien, als würden die Y-Zeilen der Tastatur in 2er-Gruppen “aneinanderkleben”. Dank der optischen Kontrolle des Tastencodes an den LED’s von Port 4 fand ich schnell heraus, nach welchem System die Tasten untereinander “verbunden” waren.

Es schien, als würden die Y-Zeilen der Tastatur in 2er-Gruppen “aneinanderkleben”.

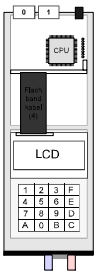
Als erstes kam mir der Gedanke, dass die “fehlenden” Pull-Up-Widerstände zwischen den Datenleitungen schuld sein könnten. Da sich dies nicht bestätigte, konnte der Fehler nicht mehr weit sein ... Jetzt blieb nur noch das Verbindungskabel, welches von den Datenausgängen des Key-Encoders direkt auf die Datenleitungen D0-D3 führt und an den entsprechenden Pins des Displays angelötet ist. Mit dem Durchgangsprüfer eines guten Digitalmultimeters prüfte ich in allen Variationen diese 4 Verbindungen - und siehe da, ein Litzenbruch in der Nähe einer Lötstelle erwies sich als des Rätsels Lösung ! Da musste ich mich wohl selbst an der Nase nehmen ...

Ein Litzenbruch in der Nähe einer Lötstelle macht mir das Leben schwer ...

Das Perfide ist, dass ich zuvor (noch während der Fertigung des Kabels) genau aus diesem Grund nach einer anderen Möglichkeit suchte, die angesprochenen Komponenten flexibel miteinander zu verbinden. Da mir aber in nützlicher Frist keine Alternative einfallen wollte, fertigte ich ein neues Kabel und lötete es mit grösster Vorsicht erneut ein.

Der nächste Versuch gelang, womit ich die Fehlersuche als Erfolg verbuchen konnte.

7.3 Bedienungsanleitung



Bevor die Einrichtung in Betrieb genommen werden kann, muss sichergestellt werden, dass die folgenden Komponenten in betriebsbereitem Zustand vorhanden sind:

7.3.1 Benötigte Komponenten

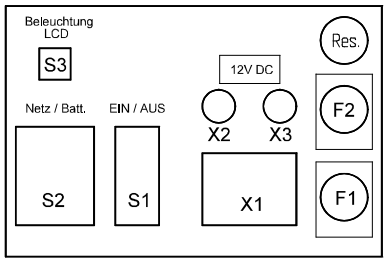
- Hardware-Aufbau (Diplomarbeit) mit *1*
  - Netz-Anschlusskabel oder *2*
  - entsprechenden Verbindungen für die Auto-Batterie (4mm Anschlussbuchsen) *3*
  - mitgelieferte Flachbandkabel zur Ansteuerung des Prototypen-Boards *4*
  - 9-Pin D-SUB-Übergangsstecker mit Null-Modem-Kreuzverbindung *5*
  - Diskette mit Source-Files oder HEX-File GPS_OEM.HEX *6*
- GPS-Empfänger-Modul mit *7*
  - eingestecktem 9-Pol-Flachbandkabel an PL5 (RS-232) *8*
  - aktiver oder passiver Antenne *9*
  - je nach Verwendung das zusätzliche 6-Pol-Flachbandkabel zu PL6 *10 (optional)*
- Prototypen-Board MCB-517A mit *11*
  - RS-232 Verbindungskabel und wenn nötig 25 zu 9-Pin- Adapter *12*
- PC oder Laptop mit mindestens einer freien seriellen Schnittstelle *13*
  - Monitor-Pragramm MON51 oder TS51 (Windows oder DOS) *14*
  - einem geeigneten Terminal-Programm zur Visualisierung des NMEA-Protokolles *15 (optional)*

7.3.2 Gerätebeschreibung (1)

Frontplatte

Frontplatte

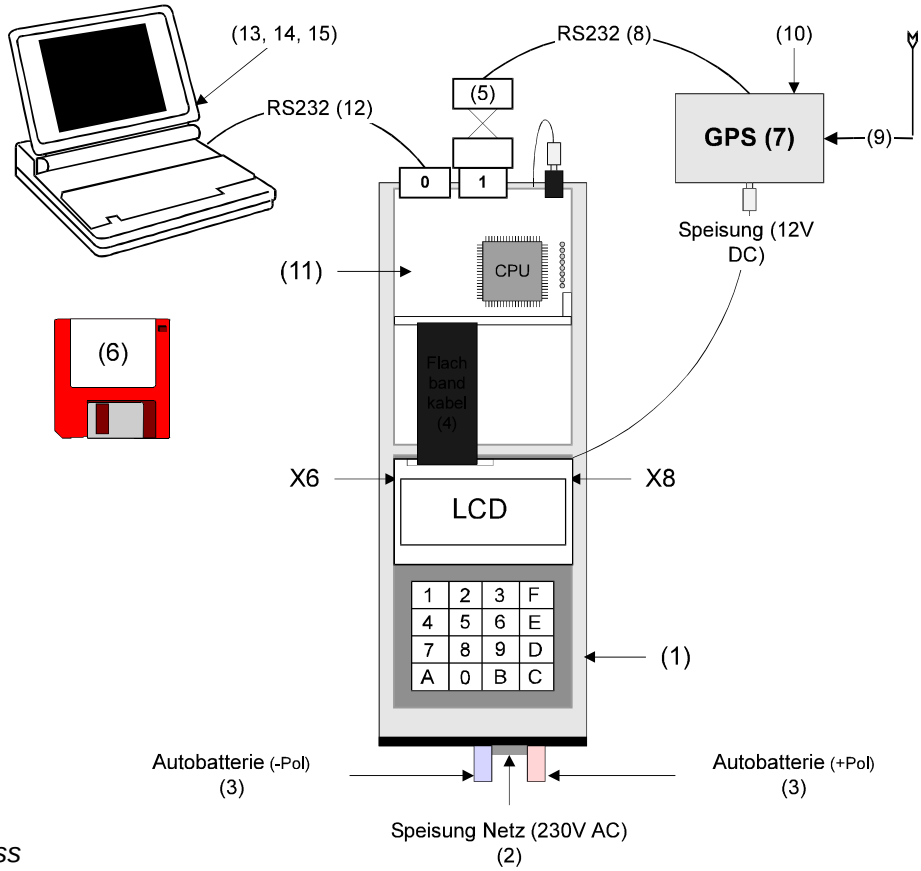
An der stirnseitig befestigten Frontplatte des Hardware-Aufbaus befinden sich die folgenden Elemente:



- S1 Hauptschalter (EIN/AUS)
- S2 Kippschalter mit Nullposition (I=Netz / II=Batt.)
- S3 Taster (Beleuchtung LCD)
- X1 Gerätestecker-Dose (230V Netz-Anschluss)
- X2/X3 4mm Anschlussbuchsen (12V Batterie-Anschluss)
- F1 Sicherung (Trafo1, 0.1AT)
- F2 Sicherung (Trafo2, 0.1AT) + Reserve-Aussparung

Das Gerät wird mit dem Hauptschalter S1 eingeschaltet. Die an der Speisung angeschlossenen Komponenten werden jedoch erst nach Betätigung des Kippschalters mit der benötigten Spannung von 12V DC versorgt.

7.3.2.1 Übersicht und Anschluss



7.3-1 Anschluss

Das Prototypen-Board wie auch der GPS-Empfänger werden direkt über die vorhandenen Kleinspannungs-Buchsen gespiesen.

Sind keine entsprechenden Anschlussmöglichkeiten vorhanden, kann die 12V DC-Spannung ebenfalls an den Klemmen X8/3 bzw. X8/4 abgegriffen werden. Um spätere Erweiterungen zu ermöglichen, wurde auf der unteren Laborkarte ein Pin-Stecker angebracht (X6), an welchem das optionale 6-Pol-Flachband-kabel auf der linken Seite eingesteckt werden kann. Die restlichen freien Pins sind ebenfalls für ev. Erweiterungen gedacht. (siehe *Übersicht Steckerbelegung* im *Anhang*)

Hinweis:
Option
- Klemme X8
- Stecker X6

Konfiguration der Schnittstellen

Das GPS-Empfänger-Modul wird im Idealfall bereits vorher mit Hilfe der mitgelieferten LABMON-Software konfiguriert bzw. die beiden Schalter S1/1 und S1/2 für den Empfang eines NMEA-Protokolles eingestellt (siehe Kapitel *3* und besonders Abschnitt *3.7*)
Ist dies bereits geschehen, kann der PC und die HW-Einrichtung eingeschaltet werden. Die Schnittstelle am PC muss für den Betrieb des Prototypen-Boards mit folgenden Parametern konfiguriert werden:

[9600 bps] [8 Daten-Bits] [keine Parität (N)] [1 Stop-Bit]

Ausrichten der Antenne

Wie in Kapitel *3* beschrieben, muss die Antenne so ausgerichtet werden, dass möglichst viel freier Himmel sichtbar ist.

7.3.3 Bedienung

Starten des Programmes (DOS)

Sind alle Vorbereitungen getroffen, kann das HEX-File mit der Eingabe LOAD:\[PATH]\GPS_OEM.HEX ins Monitor-EPROM geladen werden. Das Starten des Programmes erfolgt mit dem Befehl GO, sobald sich der Cursor wieder auf der Eingabezeile befindet.

Anzeige der geographischen Koordinaten

Die flackernden LED's an Port 4 zeigen nun an, dass die NMEA-Datensätze im 2-Sekunden Takt eingelesen werden. Sobald gültige Daten vorhanden sind, werden die Längen- und Breitengrade, die UTC-Zeit und das Datum auf das Display übertragen. (Ebenso bei späterer Betätigung der Taste 1).

Taste 1
Time UTC

```
L/B-KOORDINATEN:
47°23.17N  08°10.02E
DATUM:     28:07:98
TIME UTC:  15:52:03
```

Sollte der GPS-Empfänger aus irgendwelchen Gründen nicht aktiv sein, wird die folgende Fehlermeldung angezeigt:

Fehlermeldung 1
GPS-Empfänger nicht aktiv !

```
KEINE VERBINDUNG

ZU GPS-EMPFAENGER!
```

Ist der GPS-Empfänger aktiv, jedoch noch *keine gültigen* Daten vorhanden, wird dies mit der folgenden Meldung signalisiert:

Hinweis 1
Daten sind ungültig !

```
POSITIONS-BESTIMMUNG

NICHT MOEGLICH!
```

Diese Meldung erscheint in den meisten Fällen, bevor gültige Daten angezeigt werden können. Die benötigte Zeit dafür ist vom aktuellen Zustand des GPS-Empfänger-Moduls (verfügbare Daten im SRAM, EEPROM oder ROM) sowie von der Empfangsqualität der Antenne abhängig und kann einige Minuten dauern.

Das Wandern des Cursors sowie das zeitweise Verschwinden der Meldung wird durch den Interrupt COM1 hervorgerufen und ist kein Fehler !

Tastenbelegung

Sind die L/B-Koordinaten einmal vorhanden, erzeugen die implementierten Tasten folgende Anzeigen:

Anzeige der Schweizer-Landeskoordinaten

```
X/Y-KOORDINATEN:
X=716107m Y=233458m
DATUM:      28:07:98
SZEIT CH:   17:52:03
```

Taste 2
Sommerzeit

2

```
X/Y-KOORDINATEN:
X=716107m Y=233458m
DATUM:      28:07:98
WZEIT CH:   16:52:03
```

Taste 3
Winterzeit

3

Wenn das Verarbeiten der Daten gestoppt werden soll, kann mit der Taste 4 die Annahme des Interrupts der seriellen Schnittstelle COM1 verweigert werden. Das Programm läuft nun in seiner Endlosschleife weiter, kann aber nicht mehr auf ankommende Daten reagieren. Die zuletzt angezeigten Informationen bleiben dennoch gespeichert. Dies wird mit folgendem Hinweis angezeigt:

Sperren des Interrupts COM1

```
ANNAHME INTERRUPT

COM1 VERWEIGERT!
```

Hinweis 2
Taste 4

4

Bei Betätigung der Taste 5 erscheint eine Info-Meldung mit einigen Informationen über die Diplomarbeit.

Infos zur Diplomarbeit

```
<DIPLOMARBEIT>
"ERWEITERUNG ZU
GPS-OEM-MODUL"
SS 1998 TS/6N S.Sch.
```

Info
Taste 5

5

Mit der Taste 0 kann das Display neu initialisiert werden (Software-Reset). Gemäss den Angaben im Datenblatt des LCD-Controllers ist dies nötig, wenn beim Hardware-Reset die Zeit von 10ms beim Ansteigen der Logikspannung von 0.2V auf 4.5V überschritten wird. Der Cursor steht danach blinkend und Matrix-füllend an 1.Stelle der Zeile 1.

Display Reset
Taste 0

0

Ab und zu ist es erforderlich, das falsch beschriebene Display-RAM komplett zu löschen. Zugleich wird der LCD-Abbild-Bereich im XRAM definiert mit 'Space' (20H) beschrieben. Dazu muss die Taste C betätigt werden.

Display Clear
Taste C

C

Visualisierung des NMEA-Protokolles (Transfer-Mode)

"Durchschlaufen" des NMEA-Protokolles

Um das an der Schnittstelle COM1 empfangene NMEA-Protokoll zu visualisieren, sind die folgenden Punkte zu befolgen:

Monitorprogramm beenden

- Nach dem Starten des Programmes das Monitor-Programm korrekt beenden (bei Verwendung eines zweiten PC's ist dies nicht nötig. In diesem Fall die aktive Verbindung trennen und das serielle Verbindungskabel einfach an der entsprechenden Schnittstelle des zweiten PC's anschliessen)

Terminal-Programm öffnen

- Öffnen eines geeigneten Terminal-Programmes. Erstellen einer Direktverbindung zu der entsprechenden Schnittstelle. Diese muss mit folgenden Parametern konfiguriert werden:

[4800 bps] [8 Daten-Bits] [keine Parität (N)] [1 Stop-Bit]

Schnittstelle konfigurieren

Der Einzige Unterschied zum Betrieb mit dem Prototypen-Board besteht also in der Höhe der Baud-Rate. (In den meisten Fällen muss daher die Schnittstelle nicht einmal neu konfiguriert werden).

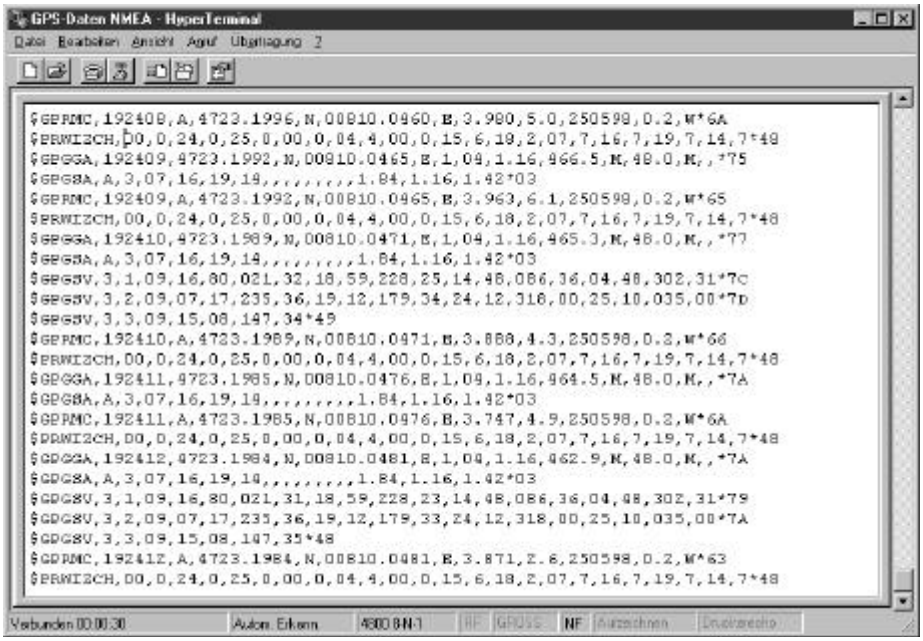
Taste F*: TRANSFER starten*

- Die Übertragung durch Betätigung der Taste F starten.

F

Vergleichen und abspeichern der NMEA-Daten mittels Terminal-Programm

Ist alles korrekt eingestellt, muss auf dem PC-Monitor das NMEA-Protokoll sichtbar sein. Das LCD-Display zeigt weiterhin das zuletzt aktive Display-Bild an. Dies kann genützt werden, um die aufbereiteten Daten mit den auf dem Monitor ersichtlichen aktuellen Daten zu vergleichen. Dazu muss lediglich die Verbindung des aktiven Terminalprogrammes getrennt (angehalten) werden. Zudem kann das NMEA-Protokoll aufgezeichnet und für späteren gebrauch als Textdatei (.txt) abgespeichert werden. Das unten ersichtliche Beispiel zeigt die Aufzeichnung mit dem Programm HYPER-TERMINAL (WIN95).



Das Programm HYPER-TERMINAL wird unter WIN95 automatisch installiert und befindet sich in der Gruppe START\PROGRAMME\ ZUBEHÖR

LABMON-Evaluation-Software

Wenn zudem Informationen über den aktuellen Zustand des GPS-Empfänger-Moduls gewünscht sind, kann das NMEA-Protokoll mit der im *Kapitel 3* beschriebenen LABMON-Evaluation-Software visualisiert werden. Somit erhält man z. B. Informationen über die Anzahl der sichtbaren oder momentan "angezapften" Satelliten. Weiter können die elementarsten Grundfunktionen genützt werden, welche **keine** bidirektionale Kommunikation erfordern (Empfänger-Abfragen).

Informationen über den Zustand des GPS-Empfänger-Moduls

Zurück in den Normalbetrieb (Standard-Mode)

Abbruch durch beliebige Taste

Da im Transfer-Mode keine weiteren Funktionen der Einrichtung genutzt werden können, erfolgt der Rücksprung in den Normalbetriebs-Modus sofort nach der Betätigung einer beliebigen Taste. Das Programm läuft in seiner ursprünglichen Endlosschleife weiter.
Wenn also das Programm einmal ins EPROM geladen und gestartet wurde, kann beliebig oft zwischen den beiden Betriebsmodi "Standard" und "Transfer" hin- und her gewechselt werden.

Erneutes starten des Monitor-Programmes bringt den Controller zum Absturz.

Schade ist nur, dass das Monitor-Programm bei einem erneutem Start-Versuch den Controller zum Absturz bringt und somit keine weiteren Eingriffe erlaubt. Dies wäre jedoch z.B. für nachträgliches Hardware-Debugging wünschenswert.

Nicht implementierte Tasten

Fehlermeldung 2
Nicht verwendete Tasten

Bei der Betätigung einer bis jetzt nicht verwendeten Taste wird die folgende Fehlermeldung angezeigt:

"ERROR:
TASTE OHNE FUNKTION!"

ACHTUNG:

Die* Taste F *darf* nur bei korrekt beendetem Monitorprogramm *betätigt werden !

Die **Taste F** zur Visualisierung des NMEA-Protokolles (Transfer-Routine) darf **nur bei korrekt beendetem Monitorprogramm** (MON51) betätigt werden ! Ansonsten wird das NMEA-Protokoll in das noch immer aktive Monitorprogramm geschrieben und bewirkt die Ausgabe von unkontrollierten Steuerzeichen auf dem Bildschirm. Dieser Zustand kann nur durch einen **HW-Reset** des Boards wieder aufgehoben werden. Zur Freigabe der Schnittstelle genügt es nicht, nur aus dem Monitorprogramm auszusteigen - die DOS-Ebene muss **ebenfalls verlassen** werden !

Programm-Status

LED's an Port 4:
Optische Kontrolle über den Status des Programmes.

Werden Daten eingelesen und verarbeitet, flackern die LED's an Port 4. Stoppt die Verarbeitung, wird der 4-Bit Tastencode der betätigten Taste an den Leuchtdioden angezeigt. Sind alle LED's dunkel, ist Taste 1 betätigt oder das Programm abgestürzt. In diesem Fall muss das Board hardwaremässig resetet und das Programm neu gestartet werden.

8. Zusammenfassung

Einleitung

Basierend auf dem Rockwell *Jupiter™ GPS-Receiver-OEM-Modul* wurde unter Verwendung der *Prototypen-Baugruppe MCB-517AC* eine Hardware-Plattform für den Empfang und die Auswertung von GPS-Daten realisiert.
Das **GPS** (**G**lobal **P**ositioning **S**ystem) ist ein vom Verteidigungsministerium der U.S.A **DoD** (**D**epartment **o**f **D**efense) installiertes und betriebenes, weltweit verfügbares Navigationssystem. Es besteht aus 24 Satelliten, die in einer Höhe von etwa 20'200 km die Erdkugel auf 6 verschiedenen Umlaufbahnen umkreisen.

Handbuch zu Jupiter™ GPS-Receiver

Die zum GPS-OEM-Modul mitgelieferten technischen Unterlagen wurden in eine deutschsprachige "Kurzanleitung" abgefasst, welche integrierender Bestandteil der Dokumentation ist, jedoch bei Bedarf auch als eigenständiges Handbuch verwendet werden kann. Ausgehend von dem dabei erlangten Wissen wurde das GPS-Empfänger-Modul mit der zugehörigen *LABMON-Evaluation-Software* in Betrieb genommen und die dabei gemachten Erfahrungen als Einstiegshilfe für nächste Anwender protokolliert.

Aufgabe / Zielsetzung

Ziel war es, eine erweiterungsfähige und mobil betriebbare Einrichtung zu entwickeln, welche unter Verwendung der *Prototypen-Baugruppe MCB-517AC* weitere Untersuchungen und Experimente von GPS-Daten erlaubt.

Hardware / Betrieb

Die Hardware besteht aus einer PVC-Grundplatte, einer stirnseitig befestigten Bedieneinheit sowie zwei übereinander angeordneten Laborkarten im Europaformat, von welchen die untere mit der Spannungsversorgung und die obere mit einem 4x20 Zeichen **LCD-Display** und einer **Tastatur** mit 16 Tasten bestückt ist. Die Ansteuerung erfolgt über den Adress-/Datenbus des Mikrocontrollers und erlaubt daher den Anschluss von weiteren Peripherie-Einheiten.
Zum Betrieb wird nebst dem Prototypen-Board ein PC oder Laptop sowie ein geeigneter GPS-Empfänger benötigt, der seine Daten über eine RS-232 Schnittstelle zur Verfügung stellen kann.

Software

Die Software wurde in **Assembler-Sprache** geschrieben und unter Zuhilfenahme von bereits bestehenden C-Berechnungs-Routinen realisiert. Durch die Aufbereitung von NMEA-0183-Protokoll-Daten ermöglicht die fertige Einrichtung die Anzeige von *geographischen Koordinaten* (Längen- und Breitengrade) und der *UTC-Zeit* (**U**niversal **T**ime **C**oordinated) sowie die Umrechnung derselben in *Schweizer Landeskoordinaten* (X-/Y-Koordinaten) mit Anzeige der MEZ (**M**ittel**e**uropäische **Z**eit) bzw. MESZ (**M**ittel**e**uropäische **S**ommer **Z**eit).

Bedienung und Funktionen

Nach dem Starten des Programmes zeigen die flackernden LED's an Port 4 an, dass die NMEA-Datensätze im 2 Sekunden-Takt eingelesen werden. Sobald *gültige* Daten vorhanden sind, werden die *Längen- und Breitengrade*, die *UTC-Zeit* und das *Datum* auf das Display übertragen. Wenn der GPS-Empfänger *keine* Daten an der Schnittstelle zur Verfügung stellt, eine *nicht implementierte Taste* betätigt wird oder die ankommenden Daten *ungültig* sind, wird dies mit einer entsprechenden **Meldung** angezeigt. Sind die Rohdaten in Form der *geographischen Koordinaten* vorhanden, können die weiteren Informationen durch Betätigung der entsprechenden Taste abgerufen werden. Die angezeigten Werte werden bei aktivem Empfang *laufend aktualisiert*. Mit einer weiteren Taste kann die Annahme des Interrupts der seriellen Schnittstelle verweigert und somit die *Datenverarbeitung unterbrochen* werden. Zudem ist es möglich, durch Betätigung einer anderen Taste das empfangene NMEA-Protokoll mit Hilfe eines geeigneten PC-Terminalprogrammes *zu visualisieren* und bei Bedarf *aufzuzeichnen*.

Schlusswort

Bis auf die Anzeige der *Schweizer Landeskoordinaten*, welche als **Resultat** an jeder Stelle den Wert '**0**' liefern, (ausstehendes Problem mit C-Routine) konnte die Aufgabe im vorgegebenen Zeitrahmen gelöst werden.

8.1 Randprobleme

GPS-Empfänger-Modul

- Die Inbetriebnahme sowie die anschliessende Benützung des GPS-Empfänger-Moduls kostete mich aufgrund von z. T. fehlerhaften Unterlagen (siehe Abschnitt *3.7*) kostbare Zeit.

DOS-Vollversion C-Compiler und Linker

- Ebenso die DOS-Vollversion des KEIL C-Compilers und des Linkers, die ich zur Kompilierung bzw. Einbindung der C-Routinen benötigte. Erst nach Durcharbeitung des halben Handbuches hatte ich es endlich geschafft, eine fehlerfreie Batch-Datei zur generierung des benötigten HEX-Files zu erstellen. Zudem benötigte der Linker noch weitere Source-Files (u.a. STARTUP.A51), welche ich an meine Applikation anpassen, mit den erforderlichen Parametern aufrufen und ebenfalls in den Link-Vorgang einbinden musste.
  Hier erwies sich der Internet-Anschluss zur elektronischen Kommunikation mit der Firma KEIL als sehr nützlich.

8.2 Ausstehendes / Verbesserungen

Angabe der X/Y-Koordinaten

- Mein grösstes Problem, das ich trotz der erwähnten Unterstützung der KEIL-Leute bis heute noch **nicht lösen konnte**, ist die Angabe der X/Y-Koordinaten mittels der C-Berechnungs-Routine. Da ich aufgrund von intensivem Hardware-Debugging zu 90% von der Richtigkeit meines Programmes überzeugt bin, habe ich diesen “Fehler” in der Dokumentation bisher niemals erwähnt. Zudem hoffe ich, dieses relativ gewichtige Problem bis zur Taxation noch irgendwie lösen zu können.
  Es ist so, dass die C-Berechnungs-Routine zwar aufgerufen wird, die benötigten Rohdaten zur Berechnung an der entsprechenden Speicherstelle “abholt” und verarbeitet, jedoch als Resultat **immer** den **Wert ‘0’** (30H) ausgibt. Dasselbe in ähnlicher Weise bei der Berechnung der Sommer- bzw. Winterzeit. Da ist es so, dass die Einer-Stelle zwar um 1 bzw. 2 erhöht wird, jedoch die Zehner-Stelle stets den **Wert ‘0’** erhält.

Die Resultate der C-Berechnungs-Routinen ergeben immer ‘0’ (30H), obwohl die Routinen korrekt aufgerufen werden und die Rohdaten vorhanden sind !

Vermutung 1

der “unerlaubte” Befehl SUBB A,ACC

Vermutung 2

Die Inkompatibilität von der Windows- zur DOS-Version

- Was ich bis jetzt herausgefunden habe und nach meiner Ansicht ein Grund für dieses “Verhalten” der C-Routine sein könnte, ist die Tatsache, dass innerhalb der Routine der vom Compiler erzeugte Befehl SUBB A,ACC auftaucht. Nach Buch [1] ist der Befehl MOV A,ACC jedenfalls nicht erlaubt, womit sich meine Vermutung zwar erhärtet hat, mir jedoch bis jetzt niemand bestätigen konnte (der Linker hat den Befehl ohne Mucks übersetzt). Da die übernommenen C-Routinen ja bereits einmal in dieser Form von D. Läubli mit Erfolg verwendet wurden und ich die Funktion der Routinen anhand des Listings und eingesetzten Zahlenbeispielen überprüfen konnte, muss der Fehler woanders liegen. Als weitere Vermutung meinerseits käme nur noch der Unterschied in Betracht, dass D. Läubli mit der Windows-Version gearbeitet hat und gewisse Librarys “meiner” etwas älteren DOS-Version gewisse Funktionen nicht enthalten oder unterstützen. Laut den KEIL-Mitarbeitern sollte dies jedoch nicht der Fall sein. Zudem hätte der Linker sicherlich auch hier entsprechende Fehlermeldungen ausgegeben.

Aus Zeitgründen musste ich die Fehlersuche momentan beenden, werde aber bemüht sein, bis zur Taxation wenigstens eine handfeste Erklärung für dieses Problem (oder noch besser dessen Lösung) zu finden.

Verbesserungen

- Was man eindeutig verbessern könnte, wäre die *Handlichkeit* des Hardware-Aufbaus, was den mobilen Betrieb erleichtern würde. Ebenso wäre bei häufigem Gebrauch die Konstruktion eines *Schutzgehäuses* sicherlich wünschenswert.

Hardware

- Das gesamte Programm könnte anstelle von *absoluten* Adressangaben (ORG) mit *relativen* Adressangaben und *Symbolen* (SEGMENT CODE / RSEG ?CO?TEXT) realisiert werden, was die Modularität erheblich steigern würde. Dazu fehlte mir jedoch erstens die Erfahrung im Programmieren und zweitens genügend Kenntnisse im Umgang mit der Entwicklungsumgebung.

Software

8.3 Kommentar

- Wie meistens bei solchen Arbeiten ist die erste Zeit recht mühsam, bis das Konzept steht, alle benötigten Informationen zusammengetragen und die kleinen aber für später gewichtigen Details geklärt sind. Wenn die Sache einmal “am laufen” ist, geht’s Schritt für Schritt aber kontinuierlich vorwärts.

Schritt für Schritt ...

- Der Aufwand, um eine solche Arbeit zu realisieren, steht natürlich in keinem Verhältnis mit dem Ertrag (besonders wenn die C-Routine nicht will!), dafür ist der Lern-Effekt umso grösser. Dies dünkt mich eine der wichtigsten Erkenntnisse, die ich bis jetzt aus dieser Arbeit ziehen konnte.

Aufwand / Ertrag

- Das erste Mal wirklich nützlich war mir das *Internet* zur Beschaffung von Informationen und vor allem zum herunterladen von *Datenblättern* im PDF-Format, was mir das zeitraubende Zusammenkopieren und “zusammenkleben” weitgehend ersparte. Zudem gibt es gerade über GPS enorme Mengen von Informationen, von denen ich einige in Form von WWW-Adressen im *Literaturverzeichnis* aufgeführt habe.

Internet / Datenblätter

- Hinzufügen möchte ich noch, dass die Hardware-Einrichtung ebenfalls gut für Versuche und Experimente mit dem *LCD-Controller* eingesetzt werden kann, da dieser noch einige Funktionen mehr bietet, als ich genützt habe. Für Semesterarbeiten oder um den Umgang mit dem Prototypen-Board zu vertiefen, wäre der Einsatz dieser Einrichtung ebenfalls denkbar.

Experimente mit dem LCD-Controller

- An dieser Stelle sei mir noch ein kleiner Abstecher in den Alltag erlaubt, der aufzeigt, was mit den heutigen technischen Mitteln möglich ist (vor allem dann, wenn sie in die falschen Hände geraten). Diesen Zeitungsartikel habe ich zufällig entdeckt, als ich mich gerade selbst über das GPS kundig machte.

Hackergesteuert

Einer weltweiten Gruppe von Hackern ist es offenbar gelungen, in ein Computersystem des amerikanischen Verteidigungsministeriums einzudringen, das GPS- und Militärsatelliten steuert. Als Beweis habe die Gruppierung mit dem Namen MOD (Masters of Download) geheime Daten aus dem Rechner vorgelegt.

8.4 Verdankung

Ein herzliches Dankeschön geht an

- Herrn Bazali, der mir die Welt der Assembler-Programmierung näher brachte und mir zudem ermöglichte, eine Diplomarbeit über ein technisch hochinteressantes System zu schreiben, dessen Weiterentwicklung noch immer unaufhaltsam voranschreitet
- die Firma W. Fehlmann AG in 5703 Seon, für die Benützung der Infrastruktur
  besonders an Herrn Z. Basic, der mir das Arbeiten in der Lehrlingswerkstatt zermöglichte und Herrn A. Gaffuri, welcher mir in kritischen Situationen mit seinen Anregungen neue Inspirationen gab
- die Firma LC Display AG in 5412 Gebenstorf, welche mir das Display-Modul unentgeltlich zur Verfügung stellte
- die Firma Thau Computer AG in Dietikon für die kostenlose Zustellung der KEIL Demonstration Tools CD-ROM.
- meine Arbeitskollegen Otto (Korrektur des Skripts), Thomas und Patrick, die mich während meinen “Ferien” trotz Sprüchen wie “MR. GPS”, “lauft’s ?” oder “vollgas!” immer mit Kaffee versorgten
- Vilia Armellino für den zur Verfügung gestellten 286er Laptop (DOS 3.3 rules !)
- Petrus für das “passende” Diplomarbeits-Wetter
- Hans und Sonia für die Kaffeemaschine
- Frau Rohr für den frischen Münzentee
- Vreni Bachfischer für das Kopieren der vielen Dokumente
- Boris und Lea Basic für die Unterstützung im Gebet
- Maja Basler für die mentale Unterstützung und gegenseitige Motivation
- Mölli, Wobme, Zopf und Zucki für den Ausgleich beim musizieren ...
- die KRUPPS, Foo Fighters, Smashing Pumpkins, Exhorder, Mich Gerber und die Stone Temple Pilots
- Manuela Cadruvi für die schöne Zeit während meiner 3-tätigen Sommerferien ...
- und an alle, die mich während den letzten 3 Jahren nicht vergessen haben (you know who you are !)

Danke

8.5 Schlusswort

Trotz einigen schlaflosen Nächten und nervenaufreibenden Randproblemen hat die Realisierung dieser Diplomarbeit Spass gemacht. In der Hard- wie auch in der Software-Entwicklung konnte ich mir viele neue Kenntisse aneignen, die ich trotz (oder gerade wegen) der momentanen Betätigung mehr im EDV-Bereich nicht missen möchte. Die z. T. ewas fehlende Verbindung Schule/Arbeit hat sich bei mir vor allem im Hardware-Bereich bemerkbar gemacht, da die “praktische” Elektronik bis auf die Labor-Versuche während des Studiums eigentlich Neuland für mich war. Umso mehr Grund zur Freude hatte ich, als die Adressdecodierschaltung nach der Modifizierung tatsächlich funktionierte.
Zudem habe ich einige neue Bauteile kennengelernt, auf die ich bei einer nächsten Gelegenheit ev. zurückgreifen kann.

Gestaunt habe ich über das “High-Tech” der Satelliten-Technologie beim Studium der GPS-Literatur, was natürlich fördernd für das Interesse war, da ich bis anhin nicht einmal genau wusste, wofür die Bezeichnung GPS überhaupt steht.

Obwohl ich von Beginn an ziemlich intensiv gearbeitet habe, benötigte ich satte 4-5 Wochen nur für die Übersetzung der Manuals. Die verbleibende Zeit schien mir trotz den “Ferien” recht knapp.
In Anbetracht dessen bin ich mit der Diplomarbeit selbst bis auf das beschriebene Problem mit den C-Routinen zufrieden und hoffe, dass die Einrichtung trotzdem ihren Zweck erfüllen und in einigen weiteren Versuchen eingesetzt werden kann.

Staufen, 10. August 1998

Stefan Schallenberger

9. Literaturverzeichnis

9.1 Literatur “Papier”

[1] Microcontroller Praxis
Norbert Heesel, Werner Reichstein
VIEWEG, 2. Auflage 1996
ISBN 3-528-15366-0

[2] Das MIKROCONTROLLER Kochbuch
Andreas Roth
IWT, 6. Auflage 1997
ISBN 3-88322-225-9

[3] Tabellenbuch Elektronik
Wilhelm Benz, Peter Heinks, Lothar Starke
KOHL + NOLTEMEYER, 7. Auflage 1993
ISBN 3-88173-059-1

[4] Halbleiterschaltungstechnik
Ulrich Tieze, Christoph Schenk
SPRINGER, 10. Auflage 1993
ISBN 3-540-56184-6

[5] Einführung in das Global Positioning System (GPS) und Überblick über heutige Anwendungen
Jean-Marie Zogg-Weber
Gastvortrag an der HTL Chur
siehe Anhang

[6] Vom Satellitensignal bis zur Positionsbestimmung auf der Karte
Jean-Marie Zogg-Weber
SEV Bulletin 3/98, Teil 1 und SEV Bulletin 9/98, Teil 2
siehe Anhang

[7] GPS-Satelliten-Navigation
Dr. Ing. Frank Schrödter
FRANZIS’, 1994
ISBN 3-7723-6682-1
leider vergriffen, Kopie ist bei mir erhältlich

[8] GPS Global Positioning System, Technik und Anwendung
Jürgen Strobel
FRANZIS’, 1995
ISBN 3-7723-4202-7

[9] GPS - “Klare Position”
Clive Giovannini, Field Application Engineer für Rockwell-Produkte
MEGA Link 10/96
siehe Anhang

[10] Diplomarbeit “Schweizer-Landeskoordinaten mit GPS”
Daniel Läubli
Ingenieurschule Zürich, Wintersemester 1997/98
ISZ Zürich

[11] User’s Guide 04.97 zu MCB-517AC
KEIL Software
REDACOM AG, Hauptstrasse 96, CH-2560 Nidau
Prototypen-Baugruppe MCB-517AC
siehe Anhang

9.2 Literatur “Bauteile”

[12] User’s Manual 05.95 Microcmputer Components SAB 50C517 / 80C537 SIEMENS, Edition 05/95 — *B258-H6075-G2-X-7600*

[13] Addendum to User’s Manual SAB 50C517 / 80C537 Microcmputer Components SAB 50C517 A / 80C517 A-5 SIEMENS, 04/95 — *B158-H6612-G1-X-7600*

9.3 Literatur “elektronisch”

World-Wide-Web:

- http://www.siemens.ch/vb/halbleit/micro.htm#Produkte — *SIEMENS-Microcontroller*
- http://www.keil.com — *KEIL-Software u. Demo-CD*
- http://julius.ngdc.noaa.gov/seg/potfld/geomag.html — *International Geomagnetic Reference Field (IGRF) 95*
- http://www.swisstopo.ch — *Bundesamt für Landestopographie*
- http://www.utexas.edu/depts/grg/gcraft/notes/gps/gps.html — *Global Positioning System Overview*
- http://www.sintrade.ch/sintrade/index.htm — *Sintrade AG/LTD (Rockwell)*
- http://www.access.ch/tramsoft/gps_orderform.html — *TRAMsoft Ambühler & Müller*
- http://www.phytec.de/techtec/starter.htm — *TechTec Sicherheitstechnik GmbH*
- http://home.t-online.de/home/flexgps/german.htm — *flex® GPS Neue GPS Freeware*
- http://www.oberon.ethz.ch/cs/group/eberle/StudSS95/Cronimund.html — *GPS Receiver Semester Project Summer 1995*
- http://www.unitel.ch/news/gps.html — *UNITEL AG Mobile Kommunikationstechnik*

10. Anhang

10.1 Benutzte Hilfsmittel

- Prototypen-Baugruppe MCB-517A mit einem 80C517A-Mikrocontroller von SIEMENS
- VISYS® GPS-Adaptor-Card und Rockwell Jupiter™ GPS-Receiver mit Zodiac Chipsatz
- Keil Demonstration Tools CD-ROM mit DScope-51Debugger (Windows)
- C51-Professional Developers Kit mit C51-Compiler, A51-Macro Assembler, BL51-Linker, dScope-51 Source-Level Debugger und MON51 (DOS)
- KO Gould, 100 Mhz
- PC 486/66 und P233 MMX

10.2 Kopie der erwähnten Berichte [5,6,9]

10.3 Zeitplan

10.4 Schemata Hardware

Spannungsversorgung	*Laborkarte 1*
Adressdecodierung (Ansteuerung Display und Tastatur)	*Laborkarte 2*
Übersicht Steckerbelegung / Bestückung	
Mechanik Grundplatte und Winkel	

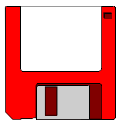
10.5 Stückliste

10.6 Software-Listings

Assembler Source-Code und List-File
C Source-Code und List-File (Berechnungs-Routinen)

10.7 Datenblätter

LM340T5	**DM7420**	**DM74LS138**	
*** HD44780U**	**DM74LS30**	**MC74HC02**	** (nur auf CD-ROM)*
*** SAB80C517A**	**SN7404**	**MM74C922**	
User's Guide 04.97 zu MCB-517AC		**LCD-Modul**	

10.8 Diskette / CD-ROM

Inhalt Diskette / CD-ROM:	**nur CD-ROM:**
Source/List-Files (Assembler und C)	gesamte Dokumentation
Batch-Datei	im Corel Ventura 5 - Format
benötigte Dateien für den Linker	Datenblätter im PDF-Format