

EINE SPIEGELKORRELATIONSMETHODE ZUR BESTIMMUNG  
VON RADIALGESCHWINDIGKEITEN MIT  
DOPPELGITTERPRISMEN

Diplomarbeit  
vorgelegt von Thomas G.M. Mainka  
aus Erftstadt

Observatorium Hoher List  
der Sternwarte  
der Rheinischen Friedrich-Wilhelm-Universität  
zu Bonn 1989

## Zusammenfassung:

Nach einem Einleitungskapitel in dem die Entwicklung der Radialgeschwindigkeitsbestimmung aus der Objektivprismenspektroskopie beschrieben und die bisherige Auswertemethode der Nelles'schen Doppelgitterprismen-Spektren kurz dargestellt wird, beschreibt der erste Teil der Arbeit den Aufbau und die Justierung des neuen außeraxialen Schmidtsystems für den Fokalreduktor am 1m-Cassegrain-Teleskop des Observatoriums Hoher List.

Dieses außeraxiale Schmidtkamerasystem ermöglicht es, trotz der geteilten Eintrittspupille, die Doppelgitterprismen-Spektren silluettierungsfrei aufzunehmen.

Im zweiten Teil der Arbeit wird dann eine neue Auswertungsmethode entwickelt, die es ermöglichen soll, Radialgeschwindigkeiten auch von Sternen späteren Spektraltyps (F8 und später) mit ausreichender Genauigkeit aus den mit dem neuen außeraxialen Schmidtkamerasystem aufgenommenen Doppelgitterprismen-Spektren zu gewinnen.

Diese neue Auswertungs-Methode beruht darauf, daß bei den Nelles'schen Doppelgitterprismen-Spektren keine Führungsfehler in Dispersionsrichtung auftreten können und somit die Koinzidenzposition  $x_c$  (die Position, an der für normales und revertiertes Spektrum die Wellenlänge übereinstimmt) nur von der Feld-Position abhängt.

Wird nun neben einer Objektivprismen-Aufnahme mit Lochmaske im Fokus des Fokalreduktors, zur Unterdrückung des Himmelshintergrundes und überlappender Sternspektren, auch die Referenz-Aufnahme mit eben diese Lochmaske durchgeführt, so haben jeweils Stern-Licht und Referenz-Licht die gleiche Koinzidenzposition und es wird über die Bestimmung dieser Koinzidenzposition möglich, trotz getrennter Aufnahmen, jedem Stern-Spektrum sein Referenz-Spektrum mathematisch aufzuprägen. Dieses ermöglicht es Objektivprismen-Spektren, die mit Doppelgitterprismen und Maskentechnik gewonnen wurden, wie Spaltspektrographen-Spektren auszuwerten und damit die Vorteile beider Geräte zu vereinigen.

## Inhaltsverzeichnis:

|       |                                                                                                |    |
|-------|------------------------------------------------------------------------------------------------|----|
| 1.    | Einführung                                                                                     | 1  |
| 1.1   | Entwicklung der Radialgeschwindigkeitsbestimmung<br>mit Hilfe der Objektivprismenspektroskopie | 1  |
| 1.2   | Motivation der Arbeit                                                                          | 5  |
| 2.    | Das neue außeraxiale Schmidtkamerasystem                                                       | 8  |
| 2.1   | Der Aufbau der Kamera                                                                          | 8  |
| 2.2   | Die Justierung der Kamera                                                                      | 14 |
| 3.    | Die Auswertungsmethode auf Korrelationsbasis                                                   | 16 |
| 3.1   | Das Ausmessen der Spektren mit dem Photomikrodensiometer                                       | 17 |
| 3.2   | Die Bestimmung der Koinzidenz-Position                                                         | 21 |
| 3.3   | Die Wellenlängen-Bestimmung                                                                    | 27 |
| 4.    | Ergebnisse und Ausblick                                                                        | 32 |
| A.    | Literatur                                                                                      | 34 |
| B.    | Tabelle der in der Arbeit verwendeten Spektren                                                 | 35 |
| C.    | Programmbeschreibung                                                                           | 36 |
| C.1   | Auswertungsroutinen in MIDAS                                                                   | 36 |
| C.1.1 | Routine DGPMAN.PR                                                                              | 36 |
| C.1.2 | Routine KOINZ.PR                                                                               | 40 |
| C.1.3 | Routine WELEICH.PR                                                                             | 42 |
| C.2   | Fortran-Programme für MIDAS                                                                    | 44 |
| C.2.1 | Programm LSQUARE.FOR                                                                           | 44 |
| C.2.2 | Programm PDSMID.FOR                                                                            | 47 |
| C.3   | Bildschirm-Ausdrucke des PDS-Steuerrechners                                                    | 73 |

## 1. Einführung

### 1.1 Entwicklung der Radialgeschwindigkeitsbestimmung mit Hilfe der Objektivprismenspektroskopie

Jede Weiterentwicklung einer Meßmethode hat das Ziel, einerseits die Genauigkeit zu verbessern, andererseits den Zeitaufwand für Messung und Auswertung zu verkürzen.

Die bedeutenste Entwicklung in der astronomischen Meßtechnik war jedoch vor circa 100 Jahren die Einführung der Photoplatte als flächenhaften Detektor. War bis dahin der Beobachter gezwungen, jedes einzelne Objekt nacheinander zu beobachten, so ermöglichte ihm nun die Photographie mehrere Objekte am Himmel, und zwar alle, die sich im Kamerafeld befinden, simultan zu beobachten. Dadurch wurde nicht nur eine Reichweitensteigerung der Beobachtungen und eine weitgehende Lösung vom subjektiven Eindrucks des Beobachters, sondern auch, durch die Simultanität der Messung, die Unabhängigkeit von zeitlich variablen Fehlern wie Justierungszustand, Durchbiegung oder Führungsfehler möglich.

Aus diesem Grund setzte, nach Vorarbeiten durch Fraunhofer (1816/19) und nachdem A. Secci (1878) ein Objektivprisma zur systematischen visuellen Spektralklassifikation benutzte, E.C. Pickering (1896) neben dem Spaltspektrographen, um das ganze Bildfeld des Teleskops ausnutzen zu können, eine Objektivprismenkamera zur photographischen Spektralklassifikation ein. Pickering erkannte auch die Möglichkeit, daß mit einer solchen Kamera teleskopzeit-sparend Radialgeschwindigkeiten mehrerer Sterne gleichzeitig bestimmt werden können.

Diese Methode leidet jedoch unter der Problematik, daß es bei der normalen Objektivprismenspektroskopie unmöglich ist, eine Wellenlängeneichung für jeden Stern durchzuführen. Da aber die Kenntnis der Dispersionskurve am Ort des Spektrums auf dem Detektor (Photoplatte, Bildverstärker, CCD) zur Umwandlung der Spektren in die Wellenlängenskala und damit zur Bestimmung der Dopplerverschiebung notwendig ist, machte Pickering vier grundlegende Vorschläge zur Lösung des Problems. Von diesen Vorschlägen ist die mögliche Einblendung von Vergleichslicht durch Hilfsprismen aus Problemen der Justierung, aufgrund von thermischen Effekten und der mangelnden Reproduzierbarkeit unbedeutend geblieben. Auch die Methode der Spektrenlänge (unabhängig auch von A. Orbinski 1895 vorgeschlagen) ist wegen ihrer zu geringen Genauigkeit bedeutungslos geblieben. Bedeutung haben nur die Absorptionsfiltermethode und die Reversionsmethode gewonnen.

Bei der Absorptionsfiltermethode wird vor dem Detektor ein schmalbandiger Absorptionsfilter angebracht, der aus jedem Spektrum einen charakteristischen Satz von Referenzlinien ausfiltert, von denen geeignete d.h. besonders scharfe Linien zur Nullpunkts-Bestimmung der aus dem Sternspektrum gewonnenen Dispersionskurve benutzt werden. Lange Zeit wurde hierfür ein Flüssigkeitsfilter, bestehend aus einer wässrigen Lösung von  $\text{NdCl}_3$  benutzt, der eine scharfe Absorptionslinie bei  $4273\text{\AA}$  zeigt. Da es sich herausgestellt hat, daß diese Moleküllinie als nicht symmetrische Absorptionsbande, eine schwärzungsabhängige Lage der Ruhewellenlänge lieferte (Fehrenbach 1947), sucht man heute nach anderen, geeigneteren Chemikalien für Flüssigkeitsfilter, bzw. man bedient sich Fabry-Perots als Absorptionsfilter.

Bei der Reversionsmethode schließlich wird ein zweites Spektrum mit entgegengesetzt verlaufender Dispersionsrichtung aufgenommen. Um dieses revertierte Spektrum zu erzeugen, wird entweder das dispergierende Medium um 180° gedreht oder das Teleskop durchgeschwenkt und dann die Photoplatte um 180° gedreht.

Da diese beiden Aufnahmen zeitlich versetzt durchgeführt werden und beim Drehen des dispergierenden Mediums Justierungsfehler unkontrollierbar in die Position der Spektren eingehen, kann mit der Reversionsmethode für jede Photoplatte zwar die Differenz der Radialgeschwindigkeiten jeweils zwischen zwei Sternen mit hoher Genauigkeit bestimmt werden, nicht aber die absolute Radialgeschwindigkeit. Um aus diesen relativen Radialgeschwindigkeiten die absoluten Radialgeschwindigkeiten zu erhalten, müssen diese noch geeicht werden. K. Schwarzschild wies 1913 in seiner grundlegenden Arbeit über die Bestimmung von Radialgeschwindigkeiten nach, daß eine solche Eichung über (mindestens) drei Sterne mit bekannter Radialgeschwindigkeit oder über die gleichzeitige Anwendung der Absorptionsfiltermethode möglich ist. In dieser Arbeit beschrieb Schwarzschild auch die vollständige Theorie des Objektivprismas.

Um aus den revertierten Spektren die relativen Radialgeschwindigkeit zu bestimmen kann man verschiedene Auswertemethoden benutzen. Eine von F. Giesecking (1976) vorgeschlagene und heute häufiger benutzte Methode ist, die Wellenlänge zu bestimmen, deren Position in Dispersionsrichtung im normalen und revertierten Spektrum übereinstimmt (Koinzidenzwellenlänge  $\lambda_c$ ). Die Verschiebung der für das bewegte System des Sternlicht bestimmten Koinzidenzwellenlänge von der des Ruhesystems ist über den Dopplereffekt proportional zur Radialgeschwindigkeit.

$$v_R = \frac{\lambda_c - \lambda_{c0}}{\lambda_{c0}} * c \quad (1)$$

Leider birgt die Reversionsmethode noch weitere Schwierigkeiten: Zum einen ist die Lage der Wellenlängen und damit auch der Koinzidenzwellenlänge im Ruhesystem wegen der Verzeichnung und der Dispersionsänderung über das Feld abhängig vom Ort des Spektrums auf der Photoplatte. Weiter ist bei normalen Prismen der ablenkende Winkel des Prismas bei der Einstellung des neuen Leitsterns nach dem Drehen des Prismas zu berücksichtigen und schließlich besteht in dichtbesetzten Sternfeldern ("crowded fields") die Gefahr, daß auf Grund von Überlappung der Spektren nur von wenigen Sternen die Spektrallinien ausgemessen und damit ihre Radialgeschwindigkeit bestimmt werden können.

Ein bedeutender Fortschritt in der Anwendung der Objektivprismen zur Radialgeschwindigkeitsbestimmung wurde durch die Einführung von gradsichtigen, verzeichnungsfreien Prismenkombinationen durch Ch. Fehrenbach (1947) erreicht. Da infolge der Gradsichtigkeit, vor und nach der Drehung der Prismenkombination, der gleiche Leitstern verwendet werden kann und durch die Verzeichnungsfreiheit nur noch die Änderung der Dispersion als von dem Plattenort abhängige Größe auftritt, ist die Bestimmung der relativen Radialgeschwindigkeiten mit einem viel geringeren Meß- und Reduktionsaufwand möglich.

Da die Herstellung der verzeichnungsfreien und gradsichtigen Prismenkombinationen nach Fehrenbach für große Teleskopöffnungen (Apertur  $\geq 60$  cm) praktisch unmöglich ist, wurde am Observatorium Hoher List ein anderer Weg beschritten und zwar die des Feldspektrographen auf Fokalreduktorbasis (Abb 1).

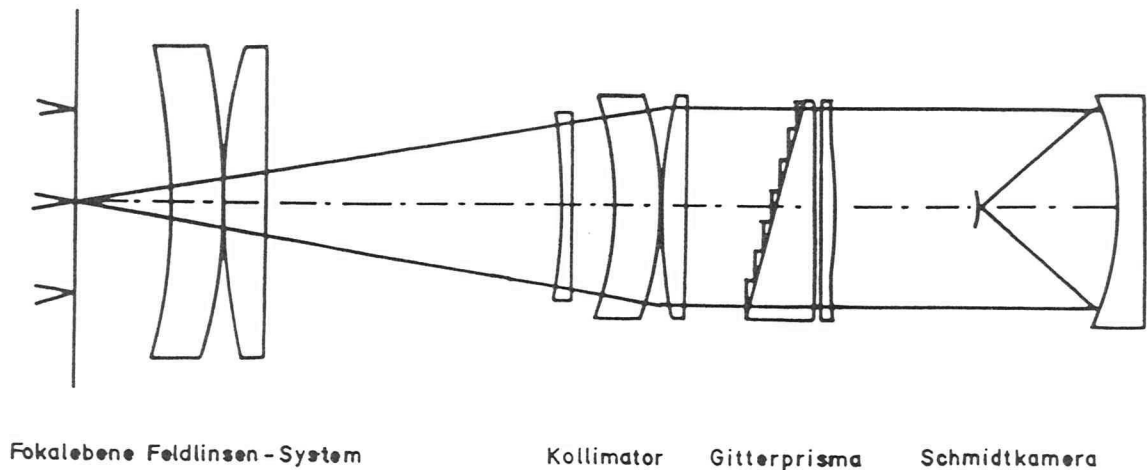


Abb 1: Feldspektrograph auf Fokalreduktorbasis (Schema)

Der am Hohen List benutzte Fokalreduktor ist nach dem Prinzip eines Kellner Okulars aufgebaut und besteht aus einem Feldlinsensystem, welches die Eintrittspupille des 1 m-Cassegrain-Teleskops auf die Austrittspupille des Fokalreduktors abbildet und einem geeignet gewähltem Kollimator, um ein paralleles Strahlenbündel zu erzeugen. Gleichzeitig bewirkt dieser Kollimator zusammen mit der an den Fokalreduktor angeschlossenen Kamera eine Verkürzung der effektiven Teleskopbrennweite  $f_{eff}$  um den Quotienten von Kamera-Brennweite  $f_{kam}$  zu Kollimator-Brennweite  $f_{koll}$ .

$$f_{eff} = f_{tel} * (f_{kam} / f_{koll}) \quad (2)$$

Mit der Verkürzung der Effektiv-Brennweite wird neben der Reduzierung des, die Auflösung des spaltlosen Feldspektrographen verschlechternden, Seeings gleichzeitig eine Verkleinerung der Öffnungszahl  $O_{eff}$  und damit eine Verbesserung der Flächenhelligkeit erreicht.

Da sich die Austrittspupille des Fokalreduktors ca. 16 cm hinter der letzten Linsenfläche befindet ist ausreichend Platz vorhanden um ein dispergierendes Medium in die Austrittspupille einzufügen. Dieses dispergierende Medium braucht nur noch die Größe der Austrittspupille des Fokalreduktors zu haben und ist möglichst dicht vor der Eintrittspupille der die Spektren aufnehmenden Kamera zu montieren, um auch deren Apertur und damit auch die Gesamtgröße klein halten zu können.

Um eine ausreichende reziproke lineare Dispersion zu bekommen wurden am Hohen List als dispergierendes Medium für eine bestimmte, vorausberechenbare Wellenlänge gradsichtige Gitterprismen (GRISM) verwendet. Dabei wird die hohe Dispersion durch das Transmissionsgitter, die Gradsichtigkeit für eine Wellenlänge durch die Keilwirkung des Prismas erreicht. (E.H.Geyer, H.Schmidt 1976)

Diese Konstruktion des Feldspektrographen mit afokalem Fokalreduktor nach Kellner-Prinzip hat gegenüber der Objektivprismenkamera den Vorteil, daß im Fokus des Teleskops eine frei zugängliche reelle Zwischenabbildung auftritt, so daß durch Einbringung einer Loch- oder Spaltmaske (Positivmaske) in den Teleskopfokus sowohl der Himmelshintergrund reduziert, als auch störende Sternspektren ausgeblendet werden können. Damit kann sowohl die Reichweite um bis zu 5<sup>m</sup> gesteigert werden, als auch die Zahl der vermeßbaren Spektren erhöht werden, was in dicht besetzten Sternfeldern von Bedeutung ist. Gleichzeitig erlaubt die Benutzung einer Mehrfachspaltplatte (Negativmaske) die simultane Gewinnung von Spektren an ausgewählten Positionen diffuser Objekte, wie Nebeln, Galaxien, planetare Nebel und Kometen.  
(E.H.Geyer, M.Hoffmann, B.Nelles 1979)

Auch dieses ursprüngliche Konzept eines Feldspektrographen konnte, da die revertierten Spektren ebenfalls zeitlich versetzt und nicht simultan aufgenommen wurden, nur relative Radialgeschwindigkeiten liefern, die mittels Anhaltsternen mit bekannter Radialgeschwindigkeit im Aufnahmegebiet oder durch die Absorptionsfiltermethode in absolute Radialgeschwindigkeiten umgewandelt werden mußten.

Aus diesem Grund schlug B.Nelles in seiner Diplomarbeit (1978) eine völlig neuartige Methode vor, die auf Doppelgitterprismen beruht. Dabei wird sozusagen das Gitterprisma geteilt und umgekehrt zusammen gesetzt (Abb 2). Mit diesen Doppelgitterprismen wurde nun erstmalig die simultane Aufnahme des normalen und revertierten Spektrums und mit der Simultanität der Aufnahmen und dem Wegfall des variablen Justierungsfehlers durch den Wegfall des Drehens des dispergierenden Mediums, die Bestimmung von absoluten Radialgeschwindigkeiten mit Objektivprismen-Spektroskopie möglich.  
(B.Nelles 1982, Dissertation 1983)

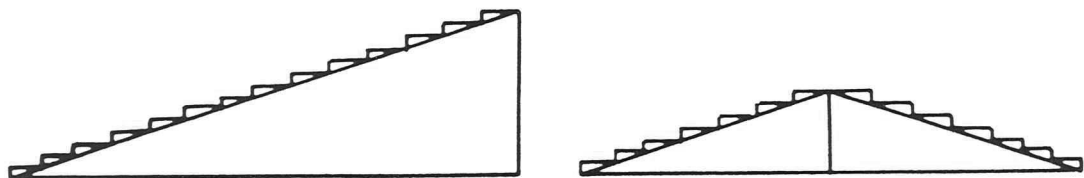


Abb 2: Gitterprisma (links) und Doppelgitterprisma (rechts)

## 1.2 Motivation der Arbeit

Da auf Grund des Themas dieser Arbeit, die Nelles'sche Methode zur Bestimmung von absoluten Radialgeschwindigkeiten modifiziert und verbessert werden soll, wird hier eine kurze Zusammenfassung gegeben, wie bisher auf Grund dieser Methode am Hohen List Radialgeschwindigkeiten bestimmt wurden <sup>1)</sup>.

Zuerst wird von dem Sternfeld, in dem Radialgeschwindigkeiten bestimmt werden sollen, eine Direktaufnahme im Fokus des 1 m-Cassegrain-Spiegels aufgenommen. Diese dient zur Herstellung der Lochmaske, durch die für jeden zu messenden Stern ein Loch gebohrt wird.

Mit der so erstellten Lochmaske im Fokus des 1 m-Spiegels werden bei der Nelles'schen Methode eine bzw. mehrere Aufnahmen von dem Sternfeld mit dem Doppelgitterprisma durchgeführt. Da die Sternspektren durch Teleskopbewegung verbreitert werden, haben sich für die Lochmaske Löcher mit einem Lochdurchmesser von 1 - 2 mm bewährt.

Zu jeder Aufnahme des Sternfeldes wird mindestens eine Referenzaufnahme in der gleichen Teleskopstellung durchgeführt. Für diese Referenzaufnahme wird eine Spektrallampe (Bewährt hat sich eine Balmer-Lampe mit Linien der Balmerreihe sowie H<sub>2</sub>O Moleküllinien und Linien des neutralen Sauerstoffs.) und eine feste Spaltplatte mit 42 regelmäßig angeordneten Spalten von 0,2\*1,5 mm, deren Spaltposition nicht mit den zu spektroskopierenden Sternen zusammenfällt, benutzt.

Die Linienpositionen der so gewonnenen Spektren wurden mit einem Abbe-Komparator mit angeschlossenen Tischrechner (HP9835A) ausgemessen, wobei der Rechner hier zur Speicherung der gemessenen Linienpositionen, zur Linien-Identifikation und zur Vorreduktion, d.h. Korrektur fehlerhafter Meßwerte benutzt wird.

Die nun folgende Reduktion der Spektren beruht auf dem funktionalen Zusammenhang zwischen der Differenz von übereinstimmenden Spektrallinien der revertierten Spektren und ihrem Feldort (A,D):

$$x(\lambda) - x'(\lambda) = G(A,D) + F(A,D) * M(\lambda) + H(A,D,\lambda) + \frac{d(x-x')}{d\lambda} * \frac{d\lambda}{dv} * v$$

mit G = Verzeichnung  
F = Dispersions Faktor  
M = Dispersionskurve in der Plattenmitte  
H = Wellenlängenabhängige Korrektur der Dispersionskurve (3)

Die hierbei auftretenden Größen G(A,D), F(A,D), M(λ) und H(A,D,λ) lassen sich als Potenzreihen ihrer Parameter darstellen, deren Koeffizienten während der Reduktion bestimmt werden müssen.

---

<sup>1)</sup> Ein ausführliches Rechenschema der Auswertung befindet sich in B.Nelles: Die Messung absoluter Radialgeschwindigkeiten mit spaltlosen Spektrographen durch Einsatz von Doppelgitterprismen, Dissertation Bonn 1983 (Anhang D, S.131ff).



Dazu wird zuerst für die Referenzaufnahme, durch Ausgleichung aus den effektiven Linienpositionen  $(x_j(\lambda_i) + x'_j(\lambda_i))/2$  der einzelnen Spektrumpaare  $j$ , deren funktionale Abhängigkeit vom Feldort  $(A_j, D_j)$  und der Wellenlänge  $\lambda_i$  bestimmt <sup>2)</sup>. Es ergeben sich dabei die Koeffizienten  $p_{ijk}$ . Danach wird aus den Liniendifferenzen der Dispersionsfaktor  $F$  und die wellenlängenabhängige Korrektur  $H$  errechnet, deren Koeffizienten  $\varepsilon_{ij}$  bzw.  $\eta_{ijk}$  durch Ausgleichung bestimmt werden <sup>3)</sup>. Nun wird die Plattenmitte durch ein Ausgleichsverfahren bestimmt. Die aus den verschiedenen Koeffizienten gewonnene Dispersionskurve  $M$  wird dann, anhand der Residuen, aus gemessenen und errechneten Linienpositionen überprüft. Schließlich wird die Verzeichnung  $G_j$  bestimmt, daraus der Koinzidenzfehler und durch Ausgleichung die Koeffizienten  $L_{ij}$  seines Potenzreihenansatzes berechnet <sup>4)</sup>.

Die Reduktion der Stern-Spektren entspricht der Reduktion der Referenz-Aufnahme bis auf die Tatsache, daß die Koeffizienten  $L_{ij}$  des Potenzreihenansatzes des Koinzidenzfehlers von der Referenz-Aufnahme übernommen werden. Die Verschiebung der Koinzidenzwellenlänge  $\lambda_c$  im bewegten System des Sternlichts gegenüber der Koinzidenzwellenlänge  $\lambda_{c0}$  im Laborsystem ergibt die Radialgeschwindigkeit  $v_R$ , die noch durch die heliozentrische Korrektur modifiziert werden muß.

Die Nelles'sche Methode birgt, in dieser Form durchgeführt, mehrere Probleme:

- Der Zusammenbau von Fokalreduktor, Doppelgitterprisma und Schmidtkamera ist durch die jeweils zur Justierung der Lochplatten notwendige Trennung von Schmidtkamera mit Doppelgitterprisma vom Fokalreduktor instabil, so daß zur Stabilitätskontrolle des Feldspektrographen durchschnittlich drei Referenz-Aufnahmen erforderlich werden.
- Zur Verbreiterung der Spektren muß das Teleskop bewegt werden, was die Führungsfehler in Dispersionsrichtung vergrößert und damit wie das Seeing die spektrale Reinheit der Spektren verschlechtert. Gleichzeitig müssen die Löcher der Lochmaske größer sein, als für die eigentliche Nachführung notwendig ist, was neben der damit verbundenen Reichweitenreduzierung des Feldspektrographen eine Referenz-Aufnahme mit Hilfe der Lochplatte selbst verbietet und damit eine empirische Wellenlängenbestimmung am Ort des Spektrums (s.u.) unmöglich macht.
- Sterne von späterem Spektraltyp (F8 und später) zeigen bei der benutzten reziproken Dispersion, der vorhandenen Detektorauflösung des Filmmaterials und dem vorhandenen Seeing, auf Grund von Blendlinien und nicht ausreichender spektraler Reinheit der Linien, eine nicht genügend große Anzahl zur Linienpositionsmessung geeigneter Linien, um die Koinzidenzwellenlänge  $\lambda_c$  im bewegten Liniensystem der Spektren mit ausreichender Genauigkeit zu bestimmen.

---

<sup>2)</sup> vergl. B.Nelles: Dissertation 1983 (aaO), Formel 99 u. 100 sowie Formel-Tabelle 7 (S.59f)

<sup>3)</sup> vergl. B.Nelles: Dissertation 1983 (aaO), Formel 86, 87, 90 u. 91 sowie Formel-Tabelle 4 (S.53ff)

<sup>4)</sup> vergl. B.Nelles: Dissertation 1983 (aaO), Formel 97 u. 98 sowie Formel-Tabelle 6 (S.57f)

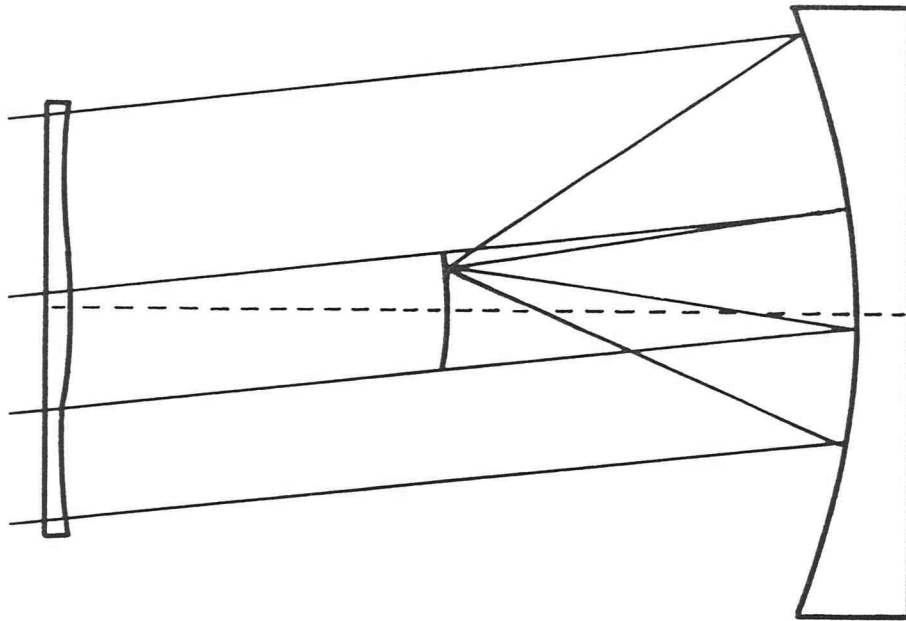


Abb 3: Silhouettierung bei einer Schmidtkamera mit geteilter Eintrittspupille

Um das Silhouettierungsproblem zu vermeiden, wurde eine neue außeraxiale Schmidtkamera von E.H.Geyer und B.Nelles (1984) vorgeschlagen, deren Konstruktion unter Berücksichtigung der anderen technischen Probleme 1987 in Angriff genommen wurde und deren Aufbau und Justierung in dieser Arbeit beschrieben werden soll.

Um nun gerade Radialgeschwindigkeiten von Sternen späteren Spektraltyps mit besserer Genauigkeit bestimmen zu können, sollte im zweiten Teil der Arbeit ein neues Reduktionsverfahren auf Korrelationsbasis, entwickelt und getestet werden.

Bei diesem Verfahren sollen, die Möglichkeiten des neuen außeraxialen Schmidtkamerasystems nutzend, nicht mehr die Positionen einzelner brauchbarer Linien mit einem Abbe-Komparator ausgemessen werden, sondern es soll das ganze Spektrum mit dem Photomikrodensitometer (PDS) der Bonner Universitätssternwarte ausgemessen werden. Mittels eines Korrelationsverfahrens soll dann, unter Ausnutzung möglichst vieler spektralen Elemente, die Koinzidenzposition bestimmt werden und durch ein Vergleichsspektrum, welches mit der Lochplatte erzeugt wird, die Wellenlängeneichung empirisch, d.h. ohne explizite Bestimmung der Verzeichnungskoeffizienten oder des Koinzidenzfehlers, erfolgen.

## 2. Das neue außeraxiale Schmidtkamerasystem

### 2.1 Der Aufbau der Kamera

Der Vorteil des außeraxial benutzten Schmidtsystems liegt darin, daß bei einer geteilten Eintrittspupille keine ungleichmäßige Abschattung der Teilbündel außerhalb der Kameraachse durch die Kasette auftritt. (Abb 4)

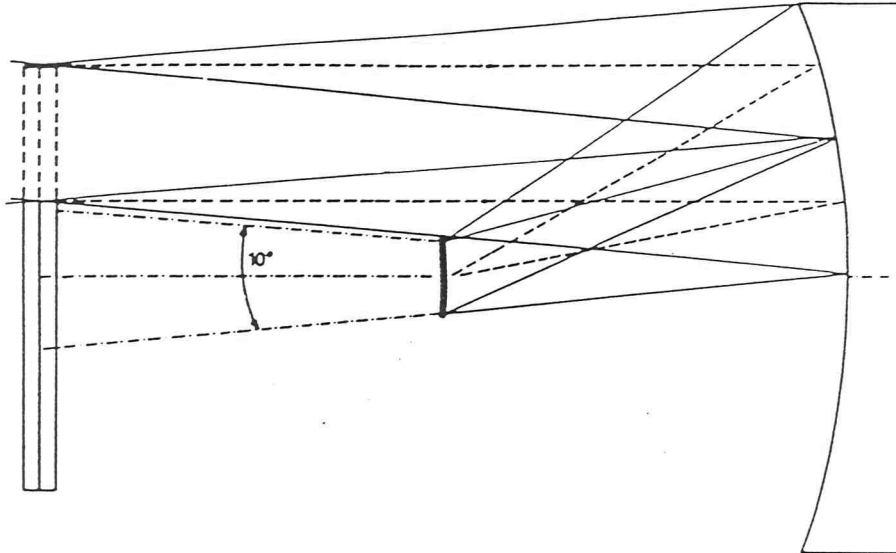


Abb 4: Vermeidung der Silhouettierung durch ein außeraxiales Schmidtkamerasystems

Dies ist nicht nur bei der Radialgeschwindigkeitsbestimmung mit Doppelgitterprismen grundlegend, sondern auch für Messungen mit der Doppelkeilplatte zur Photometrie (F.T.Lentes 1980) oder für Messungen mit einer geteilten Kalkspatplatte zur gleichzeitigen Bestimmung der Stokes-Parameter von polarisiertem Sternlicht (priv. Mitteilung von E.H.Geyer) unumgänglich. Gleichzeitig hat die Benutzung einer Schmidtkamera gegenüber einer Linsenkamera, die ebenfalls Abschattungsfrei ist, den Vorteil, frei von Koma, Astigmatismus und chromatischen Fehlern zu sein und somit über einen größeren Wellenlängenbereich benutzt werden zu können.

Wie man aus der Schemazeichnung (Abb 5) sieht, besteht das außeraxiale Schmidtkamerasystem aus zwei Teilen:

- dem Adapterstück
- der Schmidtkamera

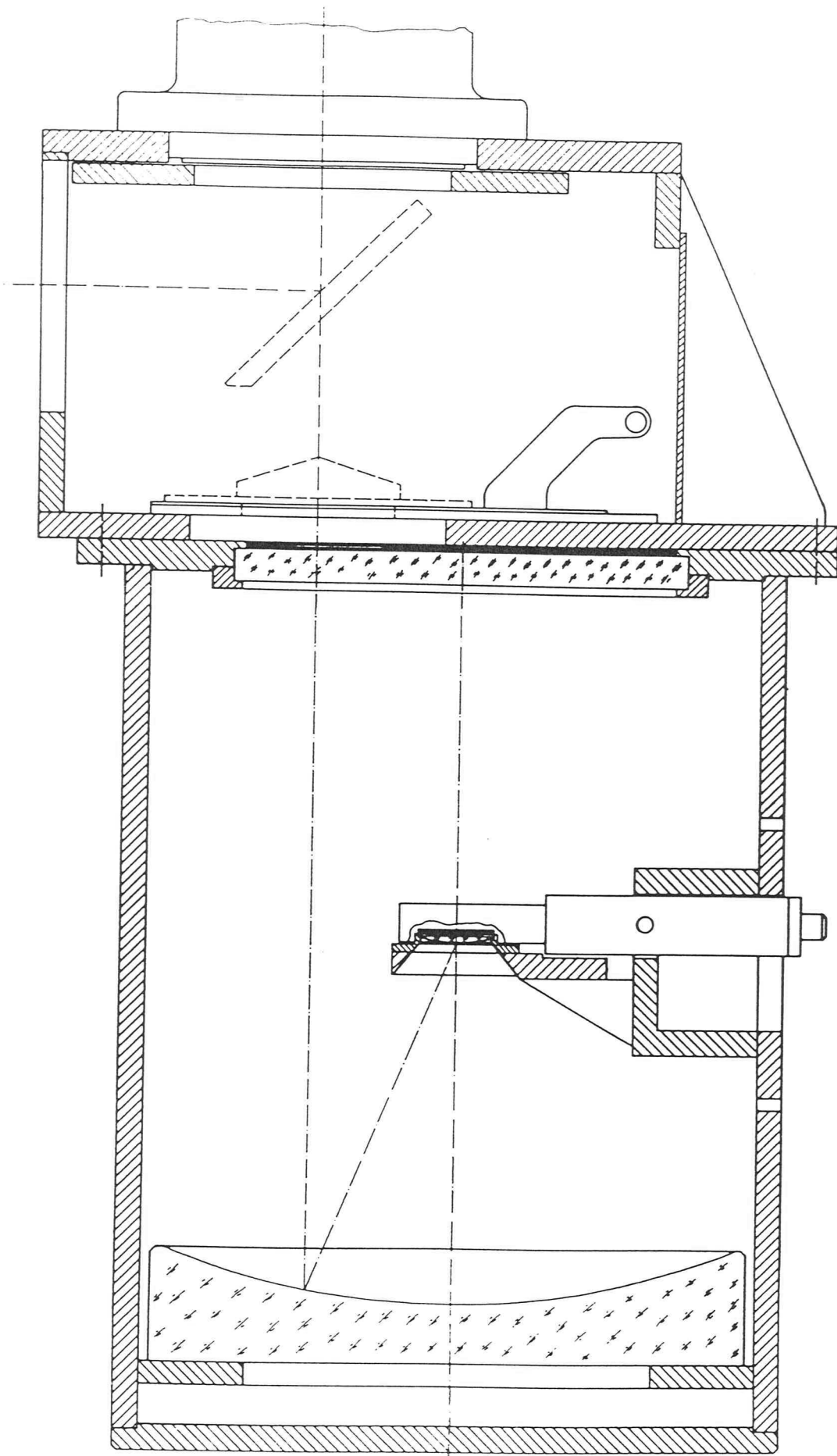


Abb 5: Schemazeichnung des neuen außersaxialen Schmidtsystems

Das Adapterstück, welches hier zur besseren Darstellung seines Aufbaus um 90° gegenüber dem normalen Gebrauch verdreht gezeichnet wurde, enthält neben einem Kameraverschluß (Firma Prontor), einem Filterschieber, zur Aufnahme von Filter- bzw. Gitterprismen-Filter-Kombinationen, einen schwenkbaren Umlenkspiegel. Dieser Spiegel ermöglicht es das parallele Lichtbündel des Kollimators vor der Austrittspupille zu einer zweiten seitlichen Austrittspupille zu lenken. An dieser Öffnung können neben dem 150mm f/2,8 Xenostar-Objektiv als Sucher auch anderer Objektive (z.B. das 85mm f/1,4 Zeiss Planar-Objektiv oder das 50mm f/1 Leitz Noktilux-Objektiv) angeschlossen werden und erlauben damit neben dem einfachen Einstellen des gewünschten Sternfeldes auch, wie schon von Nelles in seiner Dissertation gefordert, das Einjustieren der Lochplatte, ohne die Schmidtkamera vom Fokalreduktor abmontieren und vom Doppelgitterprisma trennen zu müssen. Alle diese Objektive können mit den entsprechenden Kassetten auch zur Direktaufnahme benutzt werden, jedoch ohne Filter im parallelen Strahlengang. Weiterhin kann nach Anschluß des Fokussierfernrohrs an dieser seitlichen Austrittspupille, das 1m-Cassegrain-Teleskop bequem auf den dingseitigen Fokus des Fokalreduktor fokussiert werden. Da sich dieser Spiegel nur beim Umlenken zur seitlichen Austrittspupille im Strahlengang befindet, ist für die Qualität der Aufnahmen mit der außeraxial montierten Schmidtkamera die Reflektivität und der Justierungszustand des Umlenkspiegels nicht entscheidend.

Der Kameraverschluß befindet sich direkt hinter der letzten Linsenfläche des Kollimators des Fokalreduktors. Dies hat den Vorteil, daß der Verschluß für beide Flanschanschlüsse wirksam ist. Er wird elektromagnetisch betrieben, was den Vorteil hat, daß die Öffnungszeit von einer Schaltuhr geregelt werden kann. Bei Verwendung des seitlichen Flansch-Anschlusses als Sucher kann es allerdings zu Fehlbelichtungen kommen, wenn vergessen wird, vor dem Zurückklappen des Umlenkspiegels, aus dem Strahlengang, den Verschluß zu schließen. An der Lösung dieses Problems wird noch gearbeitet, insbesondere da schon ein Schalter montiert wurde, um neben der Verschluß- auch die Spiegelstellung durch LED darzustellen.

Technische Daten des Verschlusses:

Verschlußöffnung : 100 mm  
Betriebsspannung : 12-24 Volt  
Betriebsstrom : 1,3 Ampere  
Antrieb : elektromagnetisch  
Typ.Verschlußzeit: 34 ms

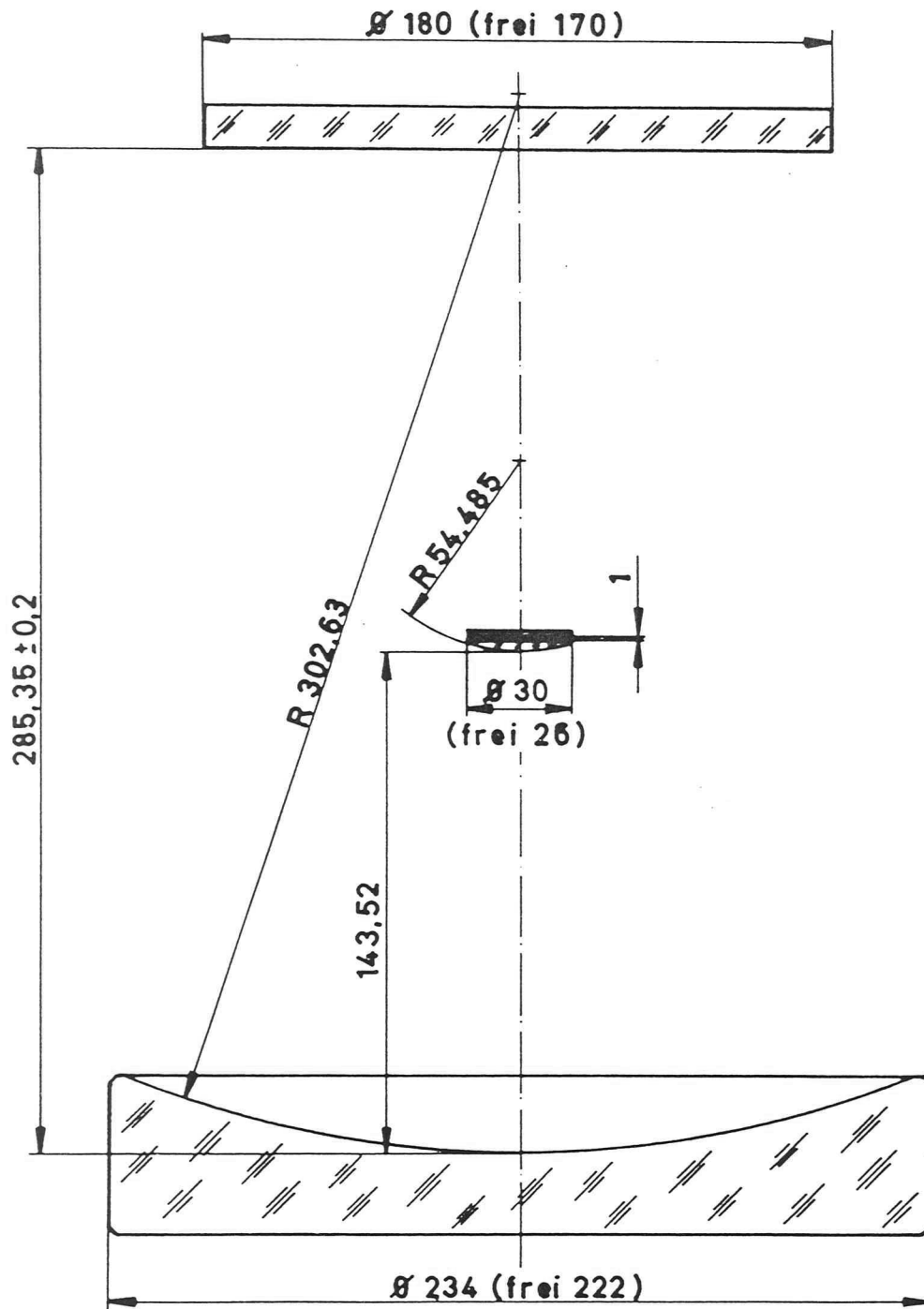
Der Filterschieber ist so konstruiert, daß er sich nach vollständigem Einführen in die Schwalbenschwanzführung selbst arretiert. Die dazugehörige Bedienungsklappe hat zur leichteren Bedienung zwei feste Stellungen und ist mit einem Filzstreifen gegen Streulicht abgedichtet. Mit dem Filterschieber können fast alle Einsätze der alten Schmidtkamera für den Fokalreduktor weiterverwendet werden. Diese sind:

- Der Filtereinsatz zur Verwendung von 2 Filtern
- Gewinde d. Filterfassung : M70x1,5
- 1. Filterebene : 22,80 mm vor der AP
- 2. Filterebene : 45,00 mm vor der AP
- max. Fassungsdicke : 9,70 mm über dem Gewindeansatz

- Das Doppelgitterprisma 1 mit Filteransatz für 1 Filter
  - Prismenwinkel  $\varphi$  : 18°00
  - Strichzahl/mm : 400
  - rez. lin. Dispersion : 145 Å/mm
  - Koinzidenzwellenlänge  $\lambda_c$  : 4102 Å
  - Prismenunterkante : 18,00 mm vor der AP
  - Filterebene : 46,50 mm vor der AP
  - max. Filterdicke : 8,20 mm über dem Gewindeansatz
- Das Doppelgitterprisma 2 mit Filteransatz für 1 Filter
  - Prismenwinkel  $\varphi$  : 14°00
  - Strichzahl/mm : 300
  - rez. lin. Dispersion : 197 Å/mm
  - Koinzidenzwellenlänge  $\lambda_c$  : 4257 Å
  - Prismenunterkante : 18,00 mm vor der AP
  - Filterebene : 43,00 mm vor der AP
  - max. Filterdicke : 11,70 mm über dem Gewindeansatz
- Das Gitterprisma 1 ohne Filteransatz
  - Prismenwinkel  $\varphi$  : 17°95
  - Strichzahl/mm : 600
  - rez. lin. Dispersion : 97 Å/mm
  - Gradsichtigkeit  $\lambda_g$  : 3050 Å
  - Prismenunterkante : 17,00 mm vor der AP
- Das Gitterprisma 2 ohne Filteransatz
  - Prismenwinkel  $\varphi$  : 12°33
  - Strichzahl/mm : 300
  - rez. lin. Dispersion : 195 Å/mm
  - Gradsichtigkeit  $\lambda_g$  : 3814 Å
  - Prismenunterkante : 17,00 mm vor der AP
- Das Gitterprisma 3 mit Filteransatz für 1 Filter
  - Prismenwinkel  $\varphi$  : 10°55
  - Strichzahl/mm : 200
  - rez. lin. Dispersion : 304 Å/mm
  - Gradsichtigkeit  $\lambda_g$  : 4850 Å
  - Prismenunterkante : 16,80 mm vor der AP
  - Filterebene : 49,80 mm vor der AP
  - max. Filterdicke : 5,00 mm über dem Gewindeansatz

Die Doppelkeilplatte (Lentes 1980) liegt gefaßt in einer M70x1,5mm Filterfassung von 8,8 mm Höhe über dem Gewindeansatz, ein UBGVR-Farbfiltersatz inklusive einem 6mm BG37-Filter in Filterfassungen von 7 mm Höhe über dem Gewindeansatz vor. Weiter existiert zur Verwendung in der 1. Filterebene des 2-fach Filtereinsatzes ein Halle Ha-Filter mit HWB 5 nm und Ta 80% in einer Filterfassung von 13 mm Höhe und ein z.Z. ungefasster Interferenzverlaufsfiler mit Maske. Das Fabry-Perot Interferometer des Hohen Listes kann mit der neuen außeraxialen Schmidtkamera nicht verwandt werden, da es eine Höhe von mehr als den durch die Konstruktion des Filterschiebers erlaubten 41 mm hat.

Die eigentliche Schmidtkamera hat eine Soll-Brennweite von 150mm und eine freie Apertur der Korrekturplatte von 170mm. Diese Öffnung wird jedoch von einer dünnen Metallblende bis auf eine Öffnung von 52mm außeraxial so abgedeckt, daß der Schatten der Plattenkassette für die Aufnahmen fast keine Rolle mehr spielt.



Optisches System der Schmidt Kamera  
nach Angaben der Herstellerfirma Jenaoptik  
(geom. Öffnungszahl  $N_{\text{geom.}} = 0.88$ ;  
bei außeraxialer Verwendung  $N = 3$   
für 50 mm - Bündel)

Ihre Technischen Daten (vergl. auch Abb 6) sind:

|                                         |             |
|-----------------------------------------|-------------|
| Brennweite (soll)                       | : 150,00 mm |
| Spiegeldurchmesser                      | : 222,00 mm |
| Spiegelradius                           | : 302,63 mm |
| Scheitelabstand Spiegel Korrekturplatte | : 285,35 mm |
| Freie Apertur der Korrekturplatte       | : 170,00 mm |
| Außeraxial genutzte Öffnung             | : 52,00 mm  |
| Abstand der Öffnung von der Mitte       | : 57,00 mm  |
| Brennweite der Ebnungslinse             | : 54,49 mm  |
| Durchmesser der Ebnungslinse            | : 26,00 mm  |
| Scheitelabstand Spiegel Ebnungslinse    | : 143,52 mm |
| Abstand Photoplatte Ebnungslinse        | : 1,00 mm   |
| Plattengröße                            | : 30*30 mm  |

Eine Besonderheit an dem Kamerasystem ist die spezielle Konstruktion der Kassettenhalterung und die Konstruktion des Kassetteneinschubs. Die Kassettenhalterung ermöglicht, neben der Fokussierung durch Verschieben der Kassetten-Ebnungslinsen-Einheit mittels Mikrometerschraube, durch mehrfache Verriegelungen das sichere und doch einfache Wechseln der Kassette auch in ungünstigen Teleskoppositionen.

Der Kassetteneinschub dagegen (Gebaut als Gesellenstück von Herrn B.Jores, siehe Abb 7) ermöglicht es, Spektren durch Verschieben der Platte in der Kassette anstatt durch Winkelverstellung des Teleskops zu verbreitern, was die Nachführgenauigkeit erhöht, kleinere Löcher in der Lochplatte zuläßt und damit die Reichweite des Feldspektrographen erhöht.

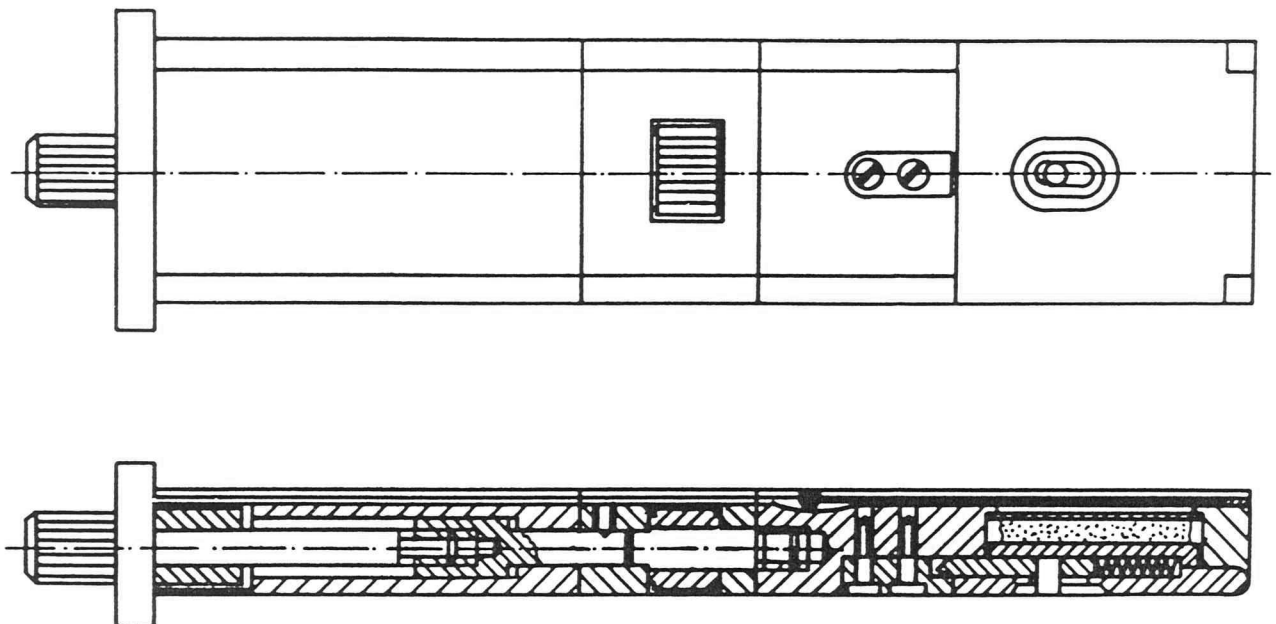


Abb 7: Aufsicht und seitlicher Schnitt durch den Kassetteneinschub



Erreicht wird dies durch eine Schraube, die in zehn Rasten einen Spalt in dem Kassetteneinschub in Schritten von 30  $\mu\text{m}$  vergrößert bzw. verkleinert. Um den Aufwand für die mehrfache Herstellung des Verbreiterungsmechanismus' zu ersparen, wurde der Kassetteneinschub so ausgelegt, daß die eigentliche Kassette von dem Verbreiterungsteil abgeschraubt werden kann. Von dieser Kassette wurden mehrere gebaut, so daß es möglich ist, mit effektiv mehreren Kassetten zu arbeiten, obwohl der Verbreiterungsmechanismus nur einmal vorhanden ist.

## 2.2 Die Justierung der Kamera

Jedes optische Gerät muß, um optimal arbeiten zu können, justiert werden. Für die Bildqualität einer Schmidtkamera sind zwei Einstellungen besonders wichtig:

- das Zentrum der Korrekturplatte muß mit dem Krümmungsmittelpunkt des Spiegels zusammenfallen.
- die Verbindungslinie vom Scheitel der Ebnungslinsen zum Spiegelscheitel muß durch den Krümmungsmittelpunkt des Spiegels gehen.

Zur Justierung des Spiegel-Krümmungsmittelpunktes auf den festgehaltenen Bezugspunkt der Korrekturplattenmitte wurden bei ausgebauter Kassettenhalterung (d.h. ohne der Kassetten-Ebnungslinsen-Einheit) zwei Fäden so über die Korrekturplatte gespannt, daß sie sich in der Mitte der Korrekturplatte kreuzten. Der Kugelspiegel wurde nun so lange mit Hilfe der Korrekturschrauben verkippt, bis das im Kugelspiegel reflektierte Bild des so entstandenen Fadenkreuzes mit dem Fadenkreuz vor der Korrekturplatte übereinstimmte.

Danach wurde die Kassettenhalterung wieder eingebaut und die Fokusslage provisorisch bestimmt. Hierzu wurde die Kassette mit einer Photoplatte, auf der ein Kreuz mittels feinem Bleistift aufgezeichnet war eingelegt. Die Schmidt-Kamera als Kollimator verwendend, wurde nun das Bild der Photoplatte mit dem auf unendlich eingestellten Fernrohr eines Autokollimationsfernrohres betrachtet und dabei der Fokus der Schmidtkamera so lange verstellt bis die Plattenstruktur und das Bleistiftkreuz scharf zu sehen war. Zur Beleuchtung des Bleistiftkreuzes wurde ein Teil der Korrekturplatte offen gehalten und mit einer Lampe ausgeleuchtet.

Da bei diesem Verfahren keine deutliche Bildfeldneigung, d.h. keine Fokusänderung über das Bildfeld, zu erkennen war, wurde auf eine sofortige Überprüfung und Justierung der Kassetten-Ebnungslinsen-Einheit verzichtet und in der folgenden Nacht eine Fokusreihe aufgenommen.

Eine Auswertung dieser Fokusreihe zeigte zwar eine gute Übereinstimmung der Fokusslage mit der vorher bestimmten Fokusslage, jedoch eine deutliche Bildfeldneigung, so daß eine Justierung der Kassetten-Ebnungslinsen-Einheit notwendig war.

Hierzu wurde das oben erwähnte Autokollimationsfernrohr mittels eines auf die Schmidtkamera gelegten planen Spiegels vorjustiert und dann, nach Entfernen des Planspiegels, die Reflexion an der Planfläche der Ebnungslinse betrachtet. Da die Schmidtkamera über keine speziellen Justierschrauben zur Justierung der Ebnungslinse verfügt, wurde mit Unterfüttern der Linsenfassung sowie Lockern und Festziehen der Befestigungsschrauben versucht, das beleuchtete Fadenkreuz des Autokollimationsfernrohres mit dem durch die Spiegelung an der Photoplatte entstandene Fadenkreuzbild in Einklang zu bringen.

Dies gelang auch bis auf einen geringen Rest-Fehler, der jedoch in der gleichen Größenordnung wie die Fokusabweichungen durch die Doppelgitterprismen und Filter lag und damit für den Anfang vernachlässigt wurde. Trotzdem ist es sinnvoll den Justierungszustand der Ebnungslinse von Zeit zu Zeit, insbesondere nach längeren Transporten und längeren Ruhepausen, zu überprüfen und zu verbessern, da sich die Unterfütterungen mit der Zeit ändern können.

### 3. Die Auswertungsmethode auf Korrelationsbasis

Die Bestimmung von absoluten Radialgeschwindigkeiten aus Doppelgitterprismen-Spektren soll in zwei funktionelle Blöcke geteilt werden:

- Die Aufnahme der Spektren mit der außeraxialen Schmidtkamera.
- Die Auswertung der Spektren.

Da sich bei der Aufnahme der Doppelgitterprismen-Spektren wenig gegenüber der bisherigen Methode geändert hat, wird hier nur kurz darauf eingegangen:

- Wie vorher wird eine Direktaufnahme im Fokus des 1 m-Cassegrain-Spiegels durchgeführt.
- Es wird aus der Direktaufnahme als Matrize wieder eine Lochmaske erstellt, jedoch mit Löchern von ca. 0,2 mm Durchmesser.
- Die Lochplatte wird im Fokus des 1m-Cassegrain-Spiegels eingeführt und mit dem 150mm Xenostar-Objektiv als Sucher einjustiert.
- Es werden mit der außeraxialen Schmidtkamera die Doppelgitterprismen-Spektren aufgenommen und dabei mit der Kassette verbreitert.
- Danach werden ebenfalls mit der Lochplatte und mit Verbreiterung durch die Kassette auf getrennter Platte die Referenzspektren aufgenommen.

Da zur Zeit noch keine Loch- bzw. Spaltfräse zur schnellen und positionsgenauen Herstellung der Maskenplatten an Hand der Aufnahmen beschafft werden konnte, war eine Prüfung der Handhabbarkeit von Lochplatten mit diesen kleinen Lochdurchmessern noch nicht möglich.

Die Auswertung der Doppelgitterprismen-Spektren kann weiter in drei Blöcke unterteilt werden:

- Das Ausmessen der Spektren mit dem Photomikrodensimeter PDS 1010A der Bonner Universitätssternwarte.
- Die Bestimmung der Koinzidenzpositionen als 0-Positionen für die Stern- und die Referenz-Spektren.
- Die Wellenlängenbestimmung der Stern-Spektren mit Hilfe der Referenz-Spektren und die Bestimmung der Radialgeschwindigkeit aus der Dopplerverschiebung.

Diese drei Auswertungsblöcke werden zusammen mit ihrer Entwicklung in den nachfolgenden Abschnitten (s.3.1-3.3) beschrieben, die dabei erstellten Routinen in kommentierter Form in Anhang C.1 aufgelistet.

Das Verfahren wurde während der Entwicklung immer wieder durch Testspektren auf seine Brauchbarkeit überprüft. Dabei wurden Tageslichtspektren, die mit der Nelles'schen Vielfachspaltplatte und dem Doppelgitterprisma II mit BG 37-Filter auf IIa-O bzw IIIa-J Platten aufgenommen wurden, benutzt. Zur spektralen Kalibrierung wurden verschiedene Spektrallampen erprobt (s.3.3). Für die Erstellung der Arbeit werden für die Abbildungen und Ergebnistabellen von diesen Spektralaufnahmen nur Spektren der 4. Aufnahmeserie (28.12.1988) bei 4 Spaltpositionen der Vielfachspaltplatte benutzt. Eine Tabelle mit den Bezeichnungen dieser Spektren (vergl.3.1-3.3) befindet sich in Anhang B.

Zur Realisierung dieser neu zu gestaltenden Auswertemethode wurde das Programmpaket MIDAS der Europäischen Südsternwarte (ESO) gewählt. Dieses hat den Vorteil, daß wesentliche Routinen der astronomischen Bildverarbeitung, wie graphische Ausgabe der Spektren, Suche von Spektrallinien, Smoothfilter, arithmetische Verarbeitung von Spektren und Tabellen und vieles mehr, schon vorprogrammiert sind. Außerdem lag die Fortran-Source der Routinen vor, so daß bei Bedarf einzelne Routinen umgeschrieben werden konnten (Anhang C.2).

### 3.1 Das Ausmessen der Spektren mit dem Photomikrodensiometer

Wie schon oben aufgeführt sollten die Spektren mit dem Bonner Photomikrodensiometer PDS 1010A von der Firma Perkins & Elmer ausgemessen werden. Diese Meßmaschine wurde seit ihrer Anschaffung 1974 in einer ganzen Reihe von Arbeiten beschrieben (M.Becker 1978: Optik & Photometrische Genauigkeit; H.J.Becker 1979: Positionsgenauigkeit; R.J.Dettmar 1981: neues Objektiv und neue Meßspalte). Seitdem ist jedoch die in den Arbeiten als Steuerrechner beschriebene PDP 8/m durch ein Kontron PSI 80 System ersetzt worden, welches über ein umfangreiches Menüsystem die einfache Einstellung der Scan-Parameter für die verschiedensten Meßaufgaben ermöglicht. Die Daten werden dabei mit dem Hilfsprogramm READPDS über DMA-Zyklen auf die Vax11/780 (AIBONN) übertragen.

Da zur Zeit außer ausgedruckten Hilfstexten keine geeignete Dokumentation über das Steuerprogramm der PDS und das Hilfsprogramm READPDS existiert sollen hier die benutzten Menüs beschrieben werden. Zusätzlich sind im Anhang C.3 Bildschirmausdrucke der beschriebenen Menüs mit Default-Werten, wie sie bei Start des Steuerrechners auftreten, hinzugefügt.

Wenn man den PSI 80 Computer startet, wird nach selbsttätigem Laden des Programms PDS der Benutzer nach seinem Benutzernamen gefragt. Dieser dient jedoch nur zur Identifikation der Daten und kann beliebig in Groß- und Kleinschreibung eingegeben werden. Nach Eingabe des Namens wird die PDS 1010A initialisiert und sollte deshalb auf "online" geschaltet sein.

Nach dieser Erst-Initialisierung wird automatisch das Header-Menü angezeigt, in dem man folgende Informationen über den einzelnen Scan eingeben bzw. korrigieren kann:

|                  |     |
|------------------|-----|
| - Benutzer       | [B] |
| - Plattenname    | [P] |
| - Typ der Platte | [T] |
| - Instrument     | [I] |
| - Objekt         | [O] |
| - Filter         | [F] |
| - Spalt          | [S] |
| - Vergrößerung   | [V] |
| - Kommentar      | [K] |

Dabei ist mit dem Instrument das Teleskop mit dem die zu messende Platte aufgenommen wurde und mit Filter, Spalt und Vergrößerung die entsprechenden Stellungen der Filter- bzw. Spalträder und die der Objektivrevolver gemeint.

Durch Betätigung der <Home> Taste gelangt man ins Hauptmenü, welches verschiedene Befehle und Untermenüs anbietet. Neben dem schon beschriebenen Header-Menü [H] sind für die Ausmessung der Spektren nur das Menü Zusätze [Z] und das Menü Scan-Definition [S] von Bedeutung.

Im Menü Zusätze können verschiedene maschinentypische Parameter gesetzt bzw. abgefragt werden. Die wichtigsten Befehle sind:

- Optimierung von Gain und Adjust-Spannung [O]
- Auswahl von linearer bzw. logarithmischer Messung [M]
- Setzen der DMA-Nummer [J]
- Auswahl der Geschwindigkeits Optimierung [G]
- Auswahl des Buffermodes [B]

Im Menü Scan-Definition können schließlich alle Scan-Parameter gesetzt werden. Die wichtigsten sind hierbei:

- Eingabe der Mittenkoordinaten eines Scans [K]
- Übernahme der aktuellen Koordinaten von der PDS [U]
- Eingabe der Scanschrittweite (Pixelgröße) [P]
- Eingabe der Scanlänge [L]
- Eingabe der Scan-Anzahl [S]
- Eingabe der Scan-Verschiebungen für Mehrfachscans [D]
- Auswahl von Meander-Scans [M]
- Start eines X-Scans [X]

Alle Koordinaten und Längen werden in Mikrometern eingegeben.

Die Position des zu messenden Spektrums kann im "offline" Modus manuell eingestellt werden und dann vom Steuerrechner übernommen werden. Vor dem Start eines Scans muß die PDS allerdings wieder "online" geschaltet werden und die AIBONN durch Starten des Hilfsprogramms READPDS auf die Datenübertragung vorbereitet werden.

Dieses Hilfsprogramm erwartet die Eingabe des Datenfiles (default READPDS.DAT) und des zu reservierenden Plattenplatzes (default 0 d.h. automatische Zuweisung nach Bedarf). Nach Durchführung des Scans muß das Programm READPDS mit <^C> beendet und bei einem weiteren, neuen Scan wieder neu gestartet werden.

Um die Spektren mit dem Programmpaket MIDAS weiterverarbeiten zu können, müssen die Daten konvertiert werden, da daß Aufzeichnungsformat der PDS-Daten nicht mit dem von MIDAS kompatibel ist. Hierzu war von P.Müller innerhalb des MIDAS-Programmpaketes ein Konvertierungsprogramm PDSMID erstellt worden, das sich jedoch nach einigen Tests als nicht ausreichend zeigte, da weder die PDS-, auch Weltkoordinaten genannt, noch die Scanschrittweite (im weiteren Gebrauch zur Anpassung an die MIDAS- und PDS-Literatur Pixelgröße genannt) nicht richtig übernommen wurden. Da diese aber eine große Rolle in der restlichen Auswertung spielen können, wurde innerhalb der Arbeit das Konvertierungsprogramm so modifiziert, daß neben der richtigen Behandlung der Weltkoordinaten auch noch die Informationen aus dem PDS-Header richtig in die MIDAS-Descriptoren übernommen wurden. Ein Programmausdruck des modifizierten Programms befindet sich in Anhang C.2.2.

Bei der Ausmessung der Doppelgitterprismen-Spektren wurden normales und revertiertes Spektrum jeweils gemeinsam gemessen. Die Koordinaten wurden dabei manuell angefahren und dann in den Steuerrechner übernommen.

Für die Auswahl des richtigen Spaltmaßes für den Meßspalt sind zwei Dinge zu beachten. Zum einen ist für ein optimales Signal-zu-Rausch Verhältnis die Fläche des Meßspaltes so groß wie möglich zu wählen, zum anderen ist die Breite des Meßspaltes kleiner als die Detektorauflösung zu halten, da sonst spektrale Information auf Grund der Verschmierung über den Meßspalt verloren geht. Empirisch wurde bei einer Verbreiterung von ca. 300  $\mu\text{m}$ , aus den vorhandenen Spaltabmessungen, für IIA-0 Platten eine Größe von 250\*25  $\mu\text{m}$ , für IIIA-J Platten eine von 200\*10  $\mu\text{m}$  als das günstigste Spaltmaß ausgewählt. Gleichzeitig wurde der Abstand der Meßpunkte in Dispersionsrichtung (Pixelgröße s.o.) auf 2  $\mu\text{m}$  festgelegt.

Die Photomultiplierspannung wurde nach einigen Versuchen und auf Empfehlung anderer PDS-Benutzer, auf Grund von Drifterscheinungen in der Spannungsversorgung bei anderen Spannungsbereichen, auf 520 Volt festgelegt.

Da als Meßergebnis mittels AD-Wandler digitalisierte Spannungen eines linear bzw. logarithmisch arbeitenden Meßverstärkers gespeichert werden, ist bei der PDS-Messung die Auswahl des Verstärkungsmodos und die Optimierung der Verstärkung (Gain) und der Nullpunktsspannung (Adjust) besonders kritisch.

Hier ergab sich, daß bei Platten des Typs IIA-0 die übliche Messung mit linear arbeitendem Verstärker, nach Durchführung einer automatischen Optimierung an einer klaren Plattenstelle, ausreichende Ergebnisse, d.h. außer den Ca H u.K-Linien genügend weitere spektrale Informationen, zeigt (Abb 8a, T4CS1). Dies mag sowohl an dem leichten Grauschleier ( $S \approx 0,2$ ) der benutzten IIA-0 Platten als auch an deren weicherer Gradation liegen, denn für die IIIA-J Platten erbrachte dies keine brauchbaren Ergebnisse (Abb 8b, T4SP1). Nach einem Versuch mit der doppelten Verstärkung (Abb 8c, T4SV1) zeigte sich, daß um auf den IIIA-J Platten schwächere Linien im Tageshimmelspektrum messen zu können mit dem logarithmisch arbeitenden Verstärker gearbeitet werden muß (Abb 8d, T4SL1).

Weil sich die bisher übliche Befestigungsmethode des Anklebens der Platten durch Klebestreifen für die 30\*30 mm großen Plättchen verbietet, wurde durch die Werkstatt des Hohen Listes eine spezielle Plattenhalterung konstruiert und gebaut. Diese wird nach Entfernen zweier Halteschrauben der Meßtisch-Glasplatte mit Hilfe deren Schraublöcher angeschraubt. Mit dieser Halterung können bis zu 3 Plättchen von 30\*30 mm über Federn befestigt werden. (Nach gleichem Prinzip können auch Halterungen für die 50\*14 mm großen Plättchen des Universal-Astro-Gitterspektrographen (UAGS) oder andere Platten, deren geringes Format das Ankleben verbietet, gebaut werden.)

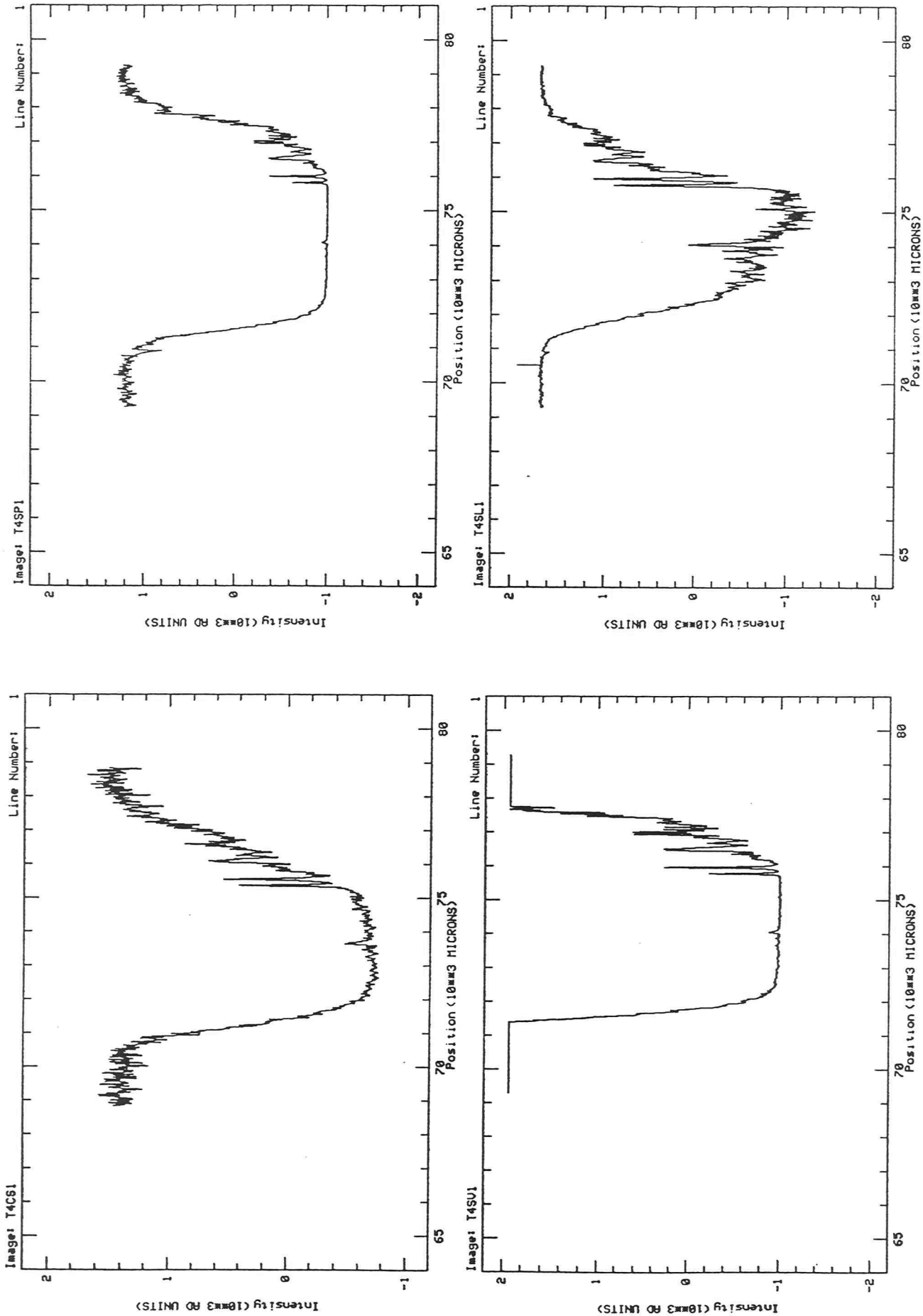


Abb 8a-d: Tageslichtspektren auf Iia-O Platte (T4CS1) und IIIa-J Platte (T4SP1 lin., T4SV1 2x, T4SL1 10g.)

### 3.2 Die Bestimmung der Koinzidenz-Position

Da die Bestimmung der Koinzidenzposition für Stern- und Referenz-Spektren identisch ist, wird in diesem Abschnitt nur allgemein von Spektren gesprochen.

Bei der Bestimmung der Koinzidenzposition  $x_c$ , also der Position an der die Wellenlänge  $\lambda$  sowohl im normalen wie im revertierten Spektrum übereinstimmt, wird davon ausgegangen, daß das normale und das revertierte Spektrum bei einer Spiegelung eines der beiden Spektren an dieser Position bis auf sehr geringe Fehler (auf Grund von Verzeichnung und Dispersionsänderung) übereinstimmt (Abb 9).

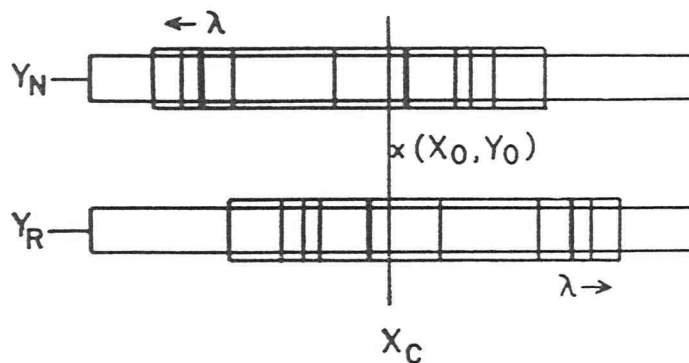


Abb 9: Schematischer Scan über ein Doppelgitterprismen-Spektrum

Um die Koinzidenzwellenlänge zu bestimmen wird nun das revertierte Spektrum an einer beliebigen  $x$ -Position (gewählt wurde wegen der einfacheren Realisierung die Mittenposition  $x_0$ ) gespiegelt und solange verschoben bis die maximale Übereinstimmung gefunden wird. Die Koinzidenzposition ergibt sich dabei als:

$$x_c = x_0 + dx/2 \quad (4)$$

mit  $x_c$  = Koinzidenzposition  
 $x_0$  = Mittenposition  
 $dx$  = Verschiebung bei maximaler Übereinstimmung

Aus diesem Grund werden in der innerhalb der Arbeit erstellten MIDAS-Kommando-Routine DGPMAN zuerst das normale und das revertierte Spektrum aus dem Doppelgitterprismen-Spektrenpaar extrahiert (EXTRACT/LINE). Nach der Spiegelung (FLIP/IMA) des revertierten Spektrums um die  $X$ -Mittenposition  $x_0$  (Abb 10) wird die Unterroutine KOINZ, in der die Verschiebung des revertierten Spektrums zum normalen Spektrums bei maximaler Übereinstimmung bestimmt wird, aufgerufen, um grob die Koinzidenzposition zu bestimmen.

Diese vorläufige Koinzidenzposition wird benutzt, um normales und revertiertes Spektrum so umzuskalieren, daß die Koinzidenzposition als Weltkoordinaten-Nullpunkt benutzt wird (REBIN/LIN). Der Benutzer wird nun, sofern er nicht vorher Default-Werte im MIDAS-Keyword SGRENZE bzw. RGRENZE abgelegt hat, gefragt, welcher Bereich der Spektren ausgewertet werden sollen.



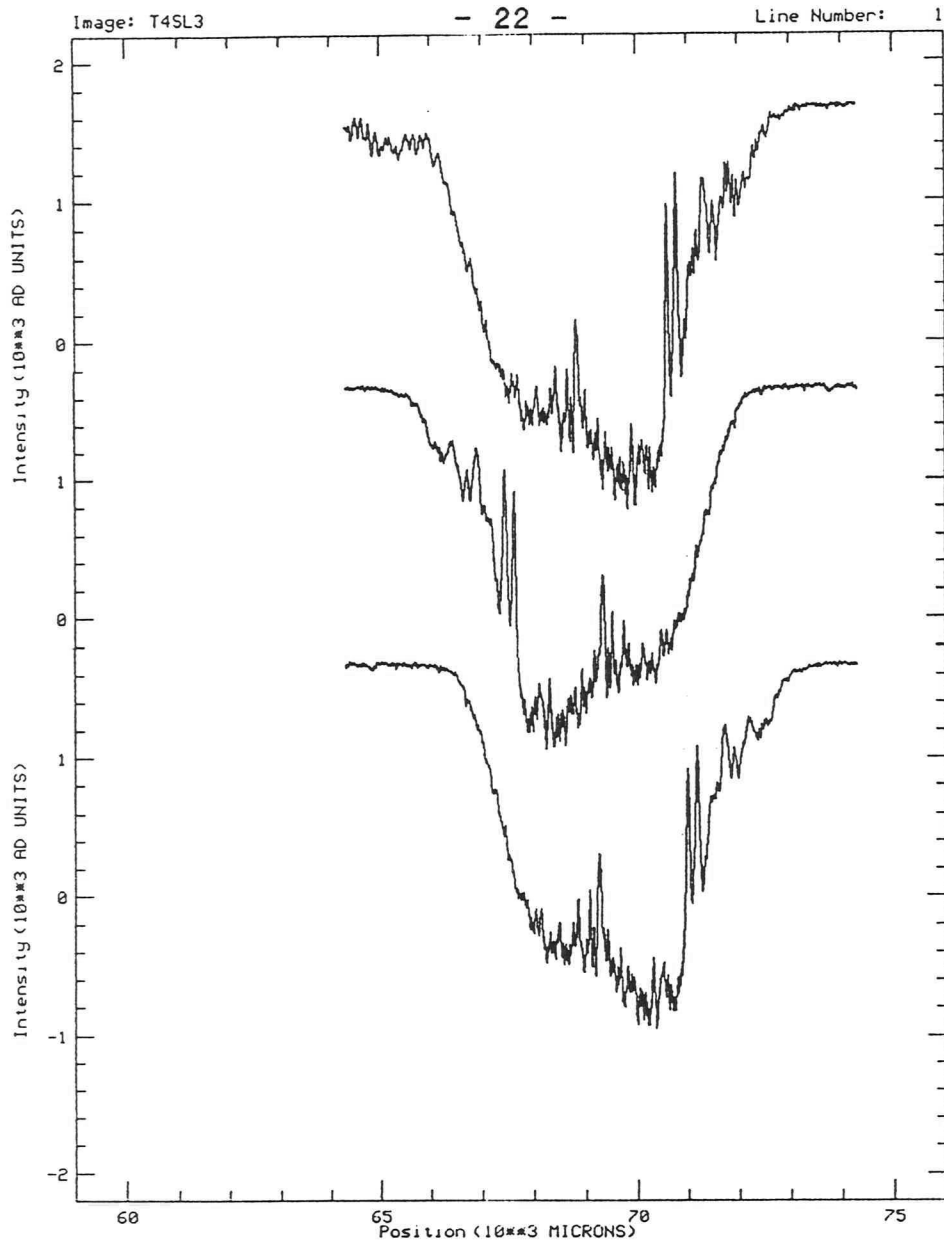


Abb 10: Normales, revertiertes und geflipptes revertiertes Spektrum untereinander dargestellt

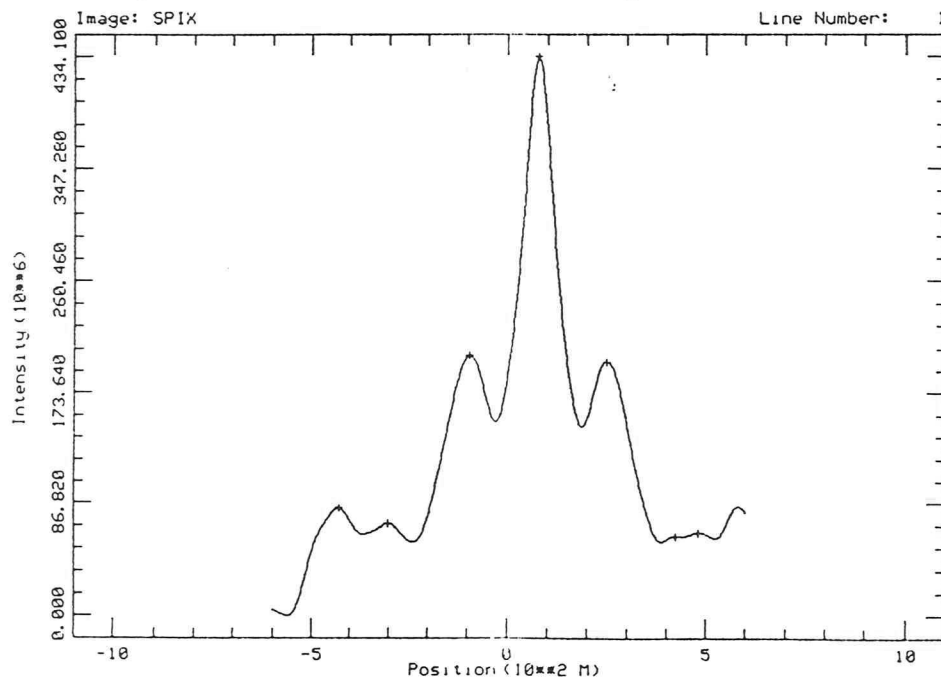


Abb 11: Übereinstimmungsfunktion mit rel. Maxima

Diese Abfrage ermöglicht es dem Benutzer, unbrauchbare, wie den Plattenrand überragende oder durch Plattenfehler zerstörte Teile des Spektrums abzuschneiden und mit dem Rest des Spektrum weiterzuarbeiten. Weiterhin kann der Benutzer, den aus Gründen der bequemeren Einstellung an der PDS zu lang gewählten Scan, auf das richtige Maß (durch die Empfindlichkeitsgrenze der verwendeten Platte vorgegeben) reduzieren.

Nach Beschränkung der Spektren auf den gewählten Bereich wird die Unterroutine KOINZ ein zweitesmal aufgerufen, um die endgültige Koinzidenzposition zu bestimmen. Danach werden die Spektren auf diese Position skaliert und die Koinzidenzposition in der Tabelle DIPL zu eventuellen Kontrollzwecken abgelegt.

In der Unterroutine KOINZ wird um die Verschiebung bei der die größte Übereinstimmung vorliegt zu bestimmen, zuerst eine Berechnung der Gaußschen Fehlerquadrate für die einzelnen Verschiebungen durchgeführt (LSQUARE/IMA modifiziert aus XCORR/IMA, s.C.2.1). Dabei wird eine Tabelle angelegt, die neben der Verschiebung dx auch das Fehlerquadrat CHI2(dx) enthält. Dabei gilt:

$$\text{CHI2}(dx) = \sum_{x=m}^{n-m} (\text{SP}_N(x) - \text{SP}_R(x+dx))^2 \quad (5)$$

mit  $n$  = Anzahl der Pixel der Spektren  
 $m$  = Halbe Breite des Least-Square-Fit in Pixel  
 $dx$  = Verschiebung im Bereich von  $-m \leq dx \leq m$

Um die Verschiebung bei maximaler Übereinstimmung, d.h. das Minimum der Fehlerquadrate genauer als auf Pixelbreite (s.Bem.3.1) zu bestimmen, wird nun die Liniensuch-Routine SEARCH/LINE benutzt. Hierzu muß allerdings vorher die Tabelle mit dem Befehl CONVERT/TAB in ein Bild umgewandelt werden. Da die Liniensuch-Routine einen Fehler enthält, so daß nur für Emissionslinien, d.h. bei der Suche nach relativen Maxima, diese Genauigkeit größer als Pixelbreite erreicht wird, werden vor der Liniensuche die Fehlerquadrate nach folgender Formel in eine Übereinstimmungsfunktion umgerechnet:

$$\text{Ueb}(dx) = -\text{CHI2}(dx) + \max(\text{CHI2}) \quad (6)$$

Hierzu werden direkt nach dem Least-Square-Fit Minimum und Maximum der Fehlerquadrate mit der Tabellenhistogramm-Routine READ/HIST bestimmt.

Sollten mit der Liniensuch-Routine mehr als ein relatives Übereinstimmungs-Maximum gefunden werden, wird dem Benutzer die Übereinstimmungs-Funktion sowie die gefundenen Maxima graphisch dargestellt (Abb 11) und um Eingabe des korrekten Übereinstimmungs-Maximums gebeten. Die Position dieses Maximums wird im Keyword DELTA(1) zurückgeliefert.

Ursprünglich wurde in der Routine KOINZ anstatt der Fehlerquadrat-Funktion folgende Korrelationsfunktion (XCORR/IMA) benutzt:

$$\text{CORR}(dx) = \sum_{x=m}^{n-m} \text{SP}_N(x) * \text{SP}_R(x+dx) \quad (7)$$

mit  $n$  = Anzahl der Pixel der Spektren  
 $m$  = Halbe Breite der Korrelation in Pixel  
 $dx$  = Verschiebung im Bereich von  $-m \leq dx \leq m$

Da jedoch die Abweichung der mittels Korrelationsfunktion bestimmten Koinzidenzpositionen teilweise mehr als  $4\mu\text{m}$  von der wahren Koinzidenzposition betrug, wurden die Korrelationsfunktion sowie als Alternative die Fehlerquadrat-Funktion genauer getestet.

Hierzu wurden mit verschiedenen Verschiebungsbereichen ein linear gemessenes IIA-0 Tageslichtspektrum (Abb 8a, T4CS1), ein logarithmisch gemessenes IIIA-J Tageslichtspektrum (Abb 8d, T4AL1), ein ESO He/Ar-Glimmlampen-Spektrum (Abb 12a, A4SL1) sowie ein Hohlkathoden Fe-Spektrum (Abb 12b, E4SL1) sowohl vollständig als auch auf einen "gültigen" Bereich eingeschränkt, mit sich selbst verglichen. Während bei der Fehlerquadrat-Funktion erwartungsgemäß die beste Übereinstimmung immer bei einer Verschiebung von  $\pm 0.00 \mu\text{m}$  gefunden wurde ergaben sich für die Korrelationsfunktion bei folgenden Verschiebungen dx die beste Übereinstimmung:

Korrelationsfunktion:

| Bereich | T4CS1 | T4CS1e | T4SL1 | T4SL1e |
|---------|-------|--------|-------|--------|
| 10      | 0.00  | 0.00   | 0.00  | 0.00   |
| 25      | 0.00  | -2.00  | 0.00  | 0.00   |
| 50      | 0.84  | 1.86   | -0.01 | -0.03  |
| 100     | 0.98  | 0.93   | 0.02  | -0.19  |
| 250     | 0.80  | 0.97   | -0.06 | 0.26   |
| 500     | -0.53 | -0.91  | -0.07 | 0.37   |
| 1000    | -3.07 | falsch | -0.67 | 0.47   |

| Bereich | A4SL1 | A4SL1e | E4SL1 | E4SL1e |
|---------|-------|--------|-------|--------|
| 10      | 0.00  | 0.00   | 0.00  | 0.00   |
| 25      | 0.00  | 0.00   | 0.00  | 0.00   |
| 50      | 0.04  | 0.03   | 0.00  | -0.05  |
| 100     | -0.04 | 0.01   | 0.01  | -0.01  |
| 250     | 0.10  | -0.09  | 0.00  | 0.06   |
| 500     | 0.13  | 0.19   | -0.03 | -0.11  |
| 1000    | 0.16  | -2.95m | -0.07 | -0.74m |

Das Ergebnis dieses Testes zeigte, daß die Korrelationsfunktion bei starken Schwärzungsgradienten im Randbereich der Spektren, wie sie an der Kante der Plattenempfindlichkeit bei Spektren, die nicht auf die Kontinuumsstrahlung normalisiert wurden, auftreten und auch bei normalisierten Spektren durch starke Linien auftreten können, größere Fehler aufweist, von denen der Least-Square-Fit verschont bleibt.

Um die Genauigkeit der Koinzidenzpositions-Bestimmung mit der Fehlerquadrat-Funktion zu überprüfen wurde in einem zweiten Test für verschiedene Tageslicht- bzw Referenz-Spektren die Abhängigkeit der Koinzidenzposition  $x_c$  von verschiedenen starken Glättungen (FILTER/SMOOTH) sowie von vereinfachten Normalisierungen (Abb 13a) der Form, daß ein sehr stark geglättetes Spektrum (Abb 13b) vom eigentlichen, ev. schwach geglätteten Spektrum abgezogen wird, überprüft.

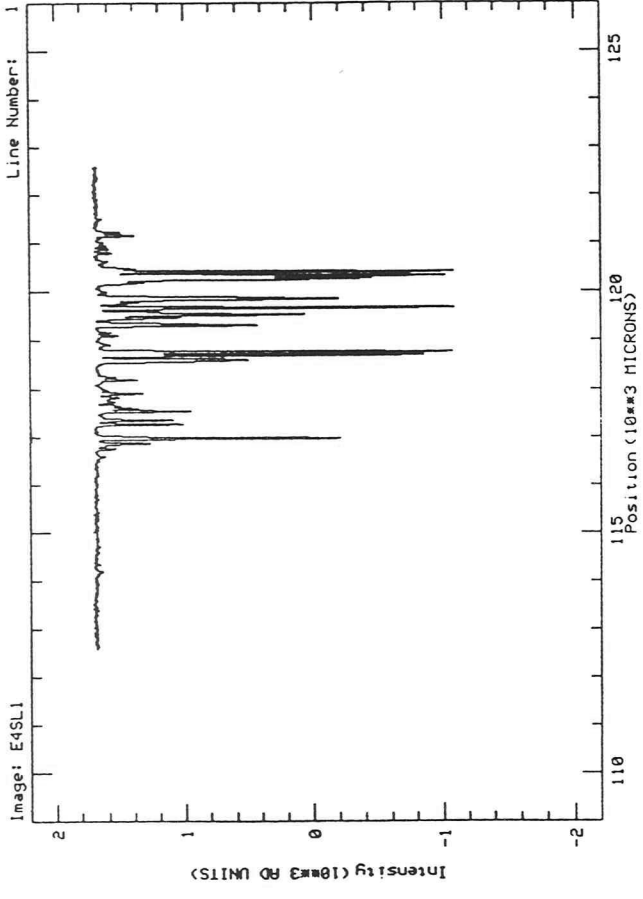


Abb 12ab: He/Ar-Glimmlampe (A4SL1) und Fe-Hohlkathode (E4SL1)

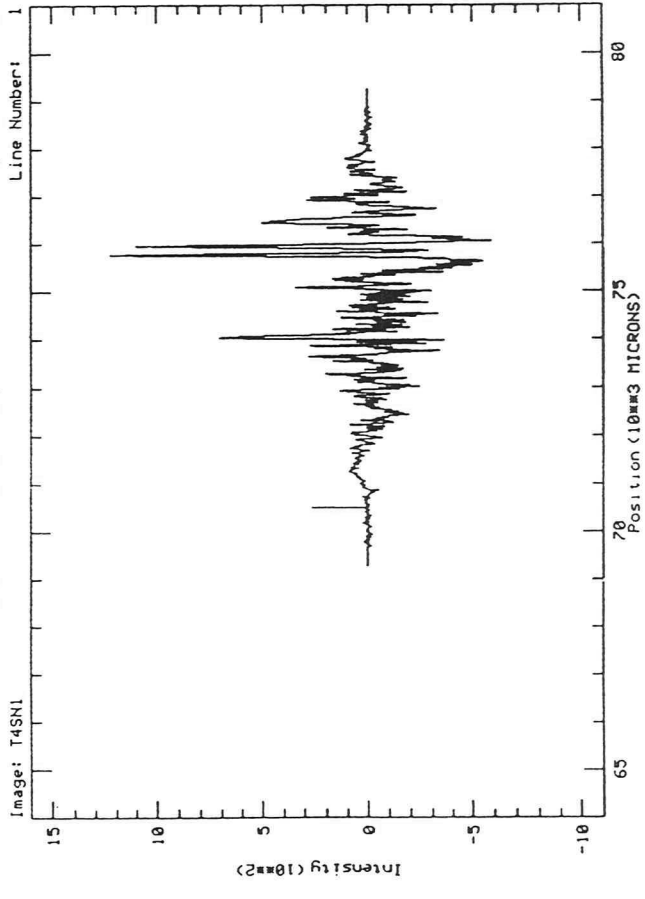
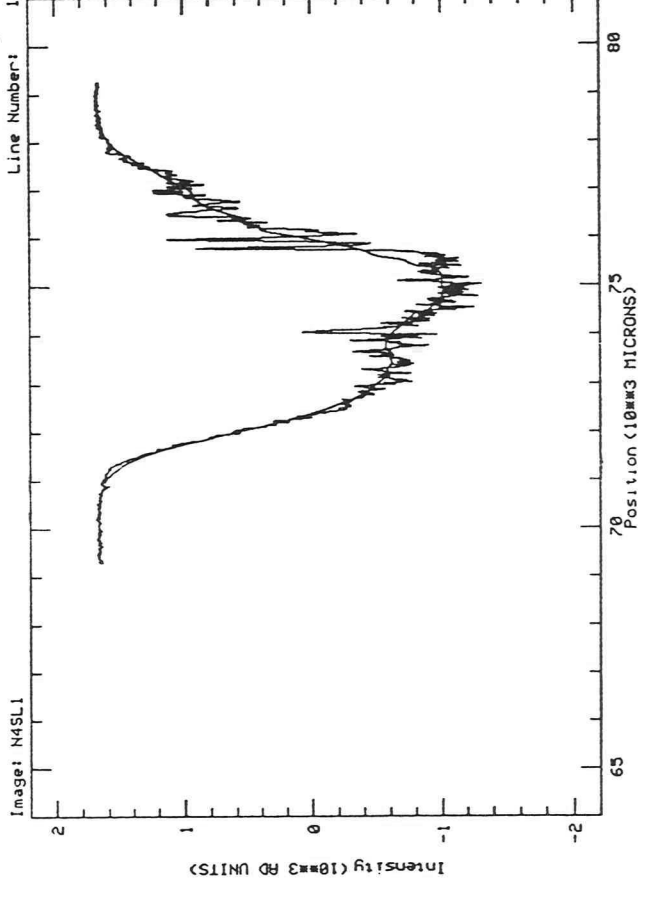
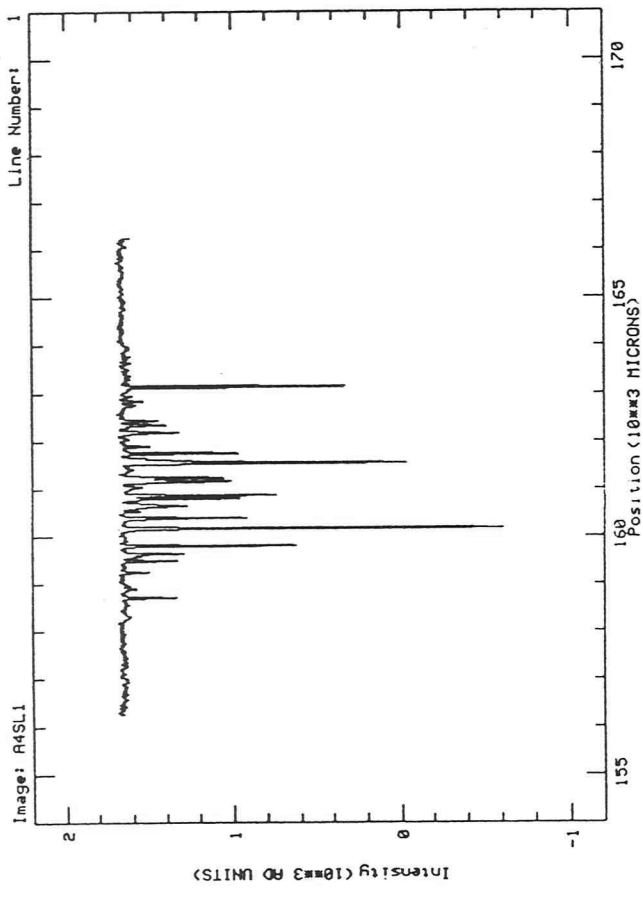


Abb 13ab: Normalisiertes Tageslichtspektrum (T4SN1) und dazu gehörendes Tageslichtspektrum (N4SL1) mit starker Glättung

Koinzidenzposition:

| Glättung  | T4CS1 | T4CS2 | T4SL1 | T4SL2 | T4SL4 |
|-----------|-------|-------|-------|-------|-------|
| ohne      | 5.517 | 3.454 | 8.524 | 4.442 | 9.485 |
| Smooth 1  | 5.512 | 3.459 | 8.536 | 4.435 | 9.422 |
| Smooth 2  | 5.511 | 3.478 | 8.548 | 4.440 | 9.402 |
| Smooth 3  | 5.513 | 3.495 | 8.557 | 4.451 | 9.371 |
| Smooth 5  | 5.530 | 3.541 | 8.576 | 4.472 | 9.273 |
| Norm 200  | 5.325 | 3.550 | 8.547 | 4.310 | 8.475 |
| N 200 S 2 | 5.254 | 3.563 | 8.577 | 4.324 | 8.471 |
| N 200 S 5 | 5.224 | 3.636 | 8.618 | 4.377 | 8.381 |
| Norm 500  | 5.421 | 3.695 | 8.526 | 4.310 | 8.542 |

| Glättung | A4SL1 | E4SL1 |
|----------|-------|-------|
| ohne     | 3.469 | 2.588 |
| Smooth 1 | 3.467 | 2.608 |
| Smooth 2 | 3.466 | 2.620 |
| Smooth 3 | 3.462 | 2.630 |
| Smooth 5 | 3.430 | 2.652 |

Als Ergebnis dieses Testes kann man feststellen, daß der durch Glättung verursachte Fehler in der Koinzidenzposition  $\Delta x_{CG1}$  in der Größenordnung von  $0,08 \mu\text{m}$  liegt. Dies ergibt umgerechnet einen Fehler in der Radialgeschwindigkeit  $\Delta v_{RG1}$  von  $\pm 1,1 \text{ km/s}$ .

Gravierender ist dagegen der Unterschied, ob eine Normalisierung durchgeführt wurde oder nicht. Hier können insbesondere am Feldrand Fehler von bis zu  $1,0 \mu\text{m}$  oder umgerechnet  $\pm 13,9 \text{ km/s}$  auftreten. Ein Vergleich der gewonnenen Radialgeschwindigkeiten bei einer früheren Aufnahmeserie ergab bei den normalisierten Spektren genauere Radialgeschwindigkeiten, d.h. eine bessere Übereinstimmung mit dem theoretischen Wert sowie eine leichtere Linienidentifikation, so daß bei der weiteren Spektrenauswertung von normalisierten Spektren ausgegangen wurde.

Da jedoch der Fehler in der Koinzidenzposition  $\Delta x_{CN0}$  für die verschiedenen Normalisierungsverfahren von der exakten Normalisierung bis zu obigen Vereinfachungen im Bereich von  $0,2 \mu\text{m}$  oder  $\pm 2,8 \text{ km/s}$  liegt wurde der Einfachheit wegen die Normalisierung mittels geglättetem Spektrum (Norm 200) gewählt. Diese ist jedoch nicht Bestandteil des Programmes und muß vom Benutzer vorher durchgeführt werden.

### 3.3 Die Wellenlängen-Bestimmung

Wenn die Spektren auf die Koinzidenzposition skaliert sind, wird von der Routine DGPMAIN für die normalen und für die revertierten Spektren getrennt die neue Unterroutine WELEICH aufgerufen, in der die Wellenlängenbestimmung der Spektren und die Bestimmung der Radialgeschwindigkeiten durchgeführt wird.

In dieser Routine WELEICH werden zuerst die Anzahl und die Positionen der Linien im Referenz-Spektrum mit der Linien-Suchroutine SEARCH/LINE bestimmt. Da im allgemeinen die gleiche Spektrallampe zur Aufnahme der Referenz-Spektren benutzt wird, wird dem Benutzer zur Beschleunigung der Identifikation der Referenz-Spektrallinien bei Übereinstimmung der Linienanzahl mit der, des vorher bearbeiteten Referenz-Spektrums, die Linienidentifikation dieses vorherigen Referenz-Spektrums zur Übernahme angeboten. Sollte diese Identifikation nicht übernommen werden können weil die Anzahl der Linien nicht übereinstimmt oder eine andere Spektrallampe gewählt wurde, sind die Linien vom Benutzer mit dem Graphik-Cursor zu identifizieren (IDENT/GCURSOR).

Diese Identifikationen werden zusammen mit den Positionen in der MIDAS-Routine CALIBRATE/LINE benutzt um durch Ausgleichung in einem Polynomansatz die Koeffizienten der Dispersionskurve am Ort des Spektrums zu bestimmen und im Keyword DISP abzulegen.

$$x(\lambda) = \text{DISP}(1) + \text{DISP}(2)*\lambda + \text{DISP}(3)*\lambda^2 \quad (8)$$

Zur Kontrolle dieser Koeffizienten werden in der MIDAS-Routine CALIBRATE/LINE zusätzlich die Abweichungen von einem linearen Ansatz für die Dispersionskurve für die gemessenen und die aus diesen Koeffizienten errechneten Linien gegen die Wellenlänge graphisch dargestellt (Abb 14). Für eventuelle spätere Kontrollen werden anschließend die Koeffizienten der Dispersionskurve zusammen mit der Koinzidenzwellenlänge in der Tabelle DIPL abgespeichert.

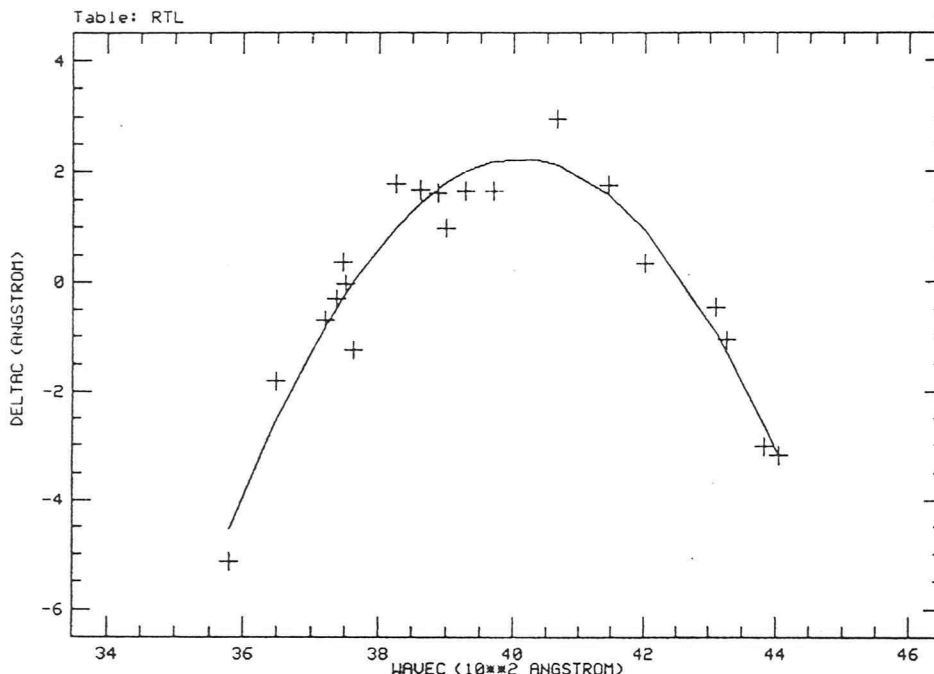


Abb 14: Graphische Darstellung der Abweichung von einem linearem Ansatz für gemessene und berechnete Linien

Im weiteren Verlauf der Unterroutine WELEICH wird nun mit Hilfe der Koeffizienten der Dispersionskurve ein Rebinning (MIDAS-Routine REBIN/WAVE) der Sternspektren vom skalierten Weltkoordinatensystem ins Labor-Wellenlängen-System durchgeführt:

$$SPL(\lambda) = SPW(x(\lambda)) \quad (9)$$

Im Wellenlängen-System werden nun mittels Linien-Suchroutine (SEARCH/LINE) die Spektrallinien im Sternspektrum gesucht und für die Bestimmung der Radialgeschwindigkeit geeignet erscheinende Linien (d.h. Linien mit ausreichender spektraler Reinheit) vom Benutzer mittels Graphik-Cursor (IDENT/GCURSOR) identifiziert. Von jeder der ausgewählten Spektrallinien liegt jetzt sowohl über die Position der Linie ihre Wellenlänge  $\lambda_{i0}$  im ruhenden Laborsystem als auch über die Identifikation die Wellenlänge  $\lambda_i$  im bewegten Stern-System vor. Nach dem Dopplergesetz wird nun für jede Linie aus ihrer Wellenlänge im Laborsystem und im Stern-System die Radialgeschwindigkeit des Sterns bestimmt und in der Tabelle mit den Linien-Positionen und -Identifikationen abgespeichert.

$$RGES(i) = \frac{(IDENT(i) - POS(i))}{POS(i)} * c \quad (10)$$

mit  $RGES(i)$  = Radialgeschwindigkeit für die i-te Linie ( $v_{Ri}$ )  
 $IDENT(i)$  = Wellenlängen-Identifikation der i-ten Linie ( $\lambda_i$ )  
 $POS(i)$  = Wellenlängen-Position der i-ten Linie ( $\lambda_{i0}$ )

Schließlich wird mit der Tabellen-Histogramm-Routine READ/HIST Mittelwert und Standardabweichung der Radialgeschwindigkeit ermittelt und in der Tabelle ERGB abgespeichert.

Um diesen Teil des Programms zu testen wurden Bestimmungen der Radialgeschwindigkeit mit verschiedenen Spektrallampen durchgeführt. Hierbei zeigte sich deutlich, wie problematisch es ist, ein geeignetes Referenz-Spektrum zu gewinnen. Zum einen muß das Referenz-Spektrum eine ausreichend hohe Anzahl an Linien in dem betreffenden Spektralbereich aufweisen, zum anderen muß die für die Aufnahme des Referenz-Spektrum benötigte Zeit in einem sinnvollem Verhältnis zur Aufnahmezeit der Stern-Spektren stehen.

In folgender Tabelle werden für die bisher getesteten Spektrallampen (vergl. Abb 15) die Anzahl der gefundenen Linien sowie die Belichtungsdauer auf IIIa-J Platten angegeben

| Spektrallampe | Linien | Belichtungsdauer |
|---------------|--------|------------------|
| Hg            | 5-6    | 30 s             |
| He            | 5-7    | 30 min           |
| Cd            | 4      | 30 min           |
| He/Ar (ESO)   | >16    | 240 min          |
| Fe-Hohlkath.  | >18    | 240 min          |
| H2O-Balmerl.  | >8     | 480 min          |

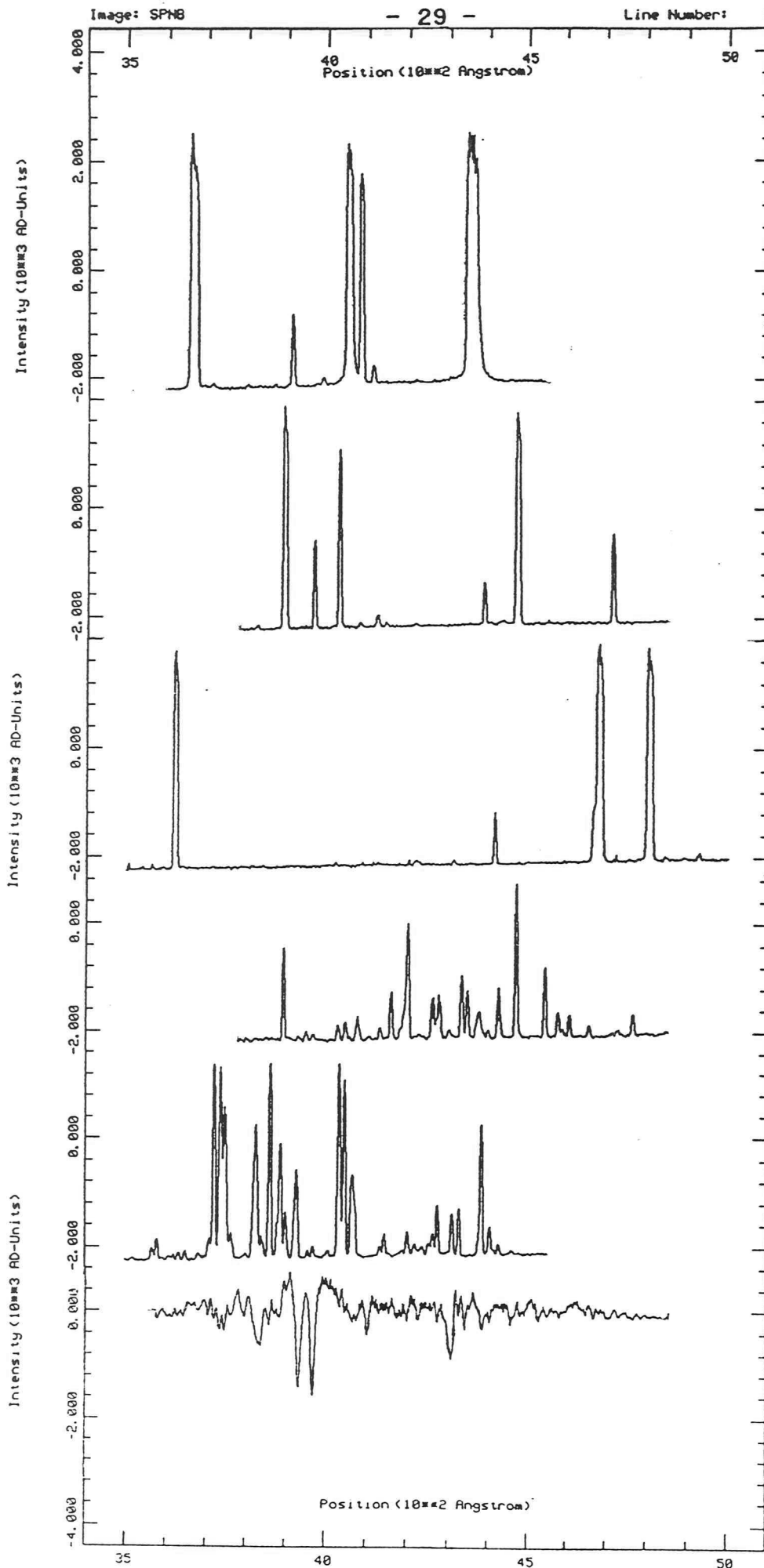


Abb 15: Vergleich der Spektrallampen (von oben: Hg, He, Cd, He/Ar, Fe, Tageslicht)



In der Anfangsphase wurden aus Gründen der Belichtungsdauer mit Hg-Spektren gearbeitet. Da dieses jedoch mit 5-6 meßbaren Linien zu wenig Linien besitzt und außerdem die beiden Randlinien (3650 Å bzw. 4358 Å) als geblendete Linien vorliegen, und damit sehr ungenaue Dispersionskurven lieferte, wurde versuchsweise mit zwei verschiedenen Referenz-Spektren auf getrennten Platten gearbeitet, die mathematisch zusammengefügt wurden. Auch diese Kombination zweier Spektren (vorzugsweise Hg & He) führte zu keinen nennenswerten Verbesserungen, war aber in der Aufnahmetechnik, dem Ausmessen und Auswerten aufwendiger, so daß für die 4. Aufnahmeserie trotz der langen Belichtungszeit mit einer ESO He/Ar Glimmlampe bzw. mit einer Fe-Hohlkathoden-Lampe gearbeitet wurde. Auf den Test der Balmer-Lampe wurde verzichtet, da eine Belichtungsdauer von 8 Stunden für ein Referenz-Spektrum in keinem Verhältnis zu einer Stern-Aufnahme mehr steht.

Aber auch mit der He/Ar-Glimmlampe bzw. der Fe-Hohlkathode ist das Problem der Auswahl der richtigen Referenzlicht-Lampe nicht endgültig gelöst, da weder die Belichtungszeiten noch die Genauigkeit den Ansprüchen, die gestellt werden müssen entsprechen. Zwar wird bei einer Stern-Aufnahme der für die Tageslichthimmel notwendige BG 37-Filter wegfallen, was einen großen Zeitgewinn bedeutet, jedoch wird durch die notwendige Verbreiterung des Referenz-Spektrums die Rest-Belichtungszeit verzehnfacht, was effektiv eine Verdoppelung der Belichtungszeit bedeutet.

Zur Angabe der Genauigkeit der Dispersionskurve kann einerseits die von MIDAS berechnete Standardabweichung dienen, welche sowohl für die He/Ar- als auch für die Fe-Spektren bei 3 Å liegt, andererseits können auch die Residuen der einzelnen Linien herangezogen werden, die zwischen 0,2 und 1 Å liegen. Dieser Fehler wirkt sich natürlich in der Bestimmung der Radialgeschwindigkeit aus, insbesondere wenn nur wenige Linien auf Grund von Blendwirkung zur Bestimmung herangezogen werden können.

Um diese Problematik in ihren Auswirkungen zu untersuchen wurden deshalb bei der Untersuchung der Tageslichtspektren kein Mittelwert der errechneten Radialgeschwindigkeiten der verschiedenen Linien aus einem Spektrum gebildet, sondern eine Mittelung der jeweiligen Linien aus den 2\*4 verschiedenen Spektren.

Wie man aus der folgenden Tabelle deutlich sieht, welche neben der Radialgeschwindigkeit getrennt für die beiden Referenz-Spektren auch die Häufigkeit enthält, daß eine Linie meßbar war, ergibt die Wahl des Referenz-Spektrums stark unterschiedliche Werte für die einzelnen Radialgeschwindigkeiten. Dies liegt einerseits an der Linienarmut des He/Ar-Spektrums im Bereich von 3700-4000 Å (teilweise Extrapolation notwendig) andererseits an den oben beschriebenen fehlerhaften Dispersionskurven. Im Vergleich der inneren Fehler liegt allerdings das Eisenspektrum mit typischen Fehlern von 50 km/s schlechter als das He/Ar-Spektrum welches bei 40 km/s liegt.

| Linie [Å] | Ident.     | $v_R$ (Eisen) | n  | $v_R$ (He/Ar) | n  |
|-----------|------------|---------------|----|---------------|----|
| 3705.57 ? | Fe I       |               | 3x |               | 2x |
| 3719.95   | Fe I       |               | 2x |               | 2x |
| 3727.63   | Fe I       |               | 2x |               | 2x |
| 3734.87   | Fe I       |               | 3x |               | 3x |
| 3745.59 ? | Fe I       |               | 2x |               | -  |
| 3749.50   | Fe I       | -86.538       | 4x | -376.288      | 5x |
| 3767.20 ? | Fe I       |               | 3x |               | 3x |
| 3797.90 ? | H10        | -90.569       | 8x | -208.433      | 8x |
| 3835.39 ? | H 9        |               | 3x |               | 4x |
| 3859.92   | Fe I       | -28.770       | 7x | -174.052      | 6x |
| 3889.03 ? | H 8        |               | 3x |               | 2x |
| 3905.53   | Fe I       |               | 2x |               | 1x |
| 3933.68   | Ca II      | 12.098        | 8x | -51.719       | 8x |
| 3968.49   | Ca II      | 8.379         | 8x | -34.723       | 8x |
| 4030.70 ? | Mn I       |               | 4x |               | 3x |
| 4045.83   | Fe I       | 36.610        | 5x | -24.680       | 5x |
| 4063.61   | Fe I       |               | 2x |               | 2x |
| 4071.75   | Fe I       |               | 2x |               | 2x |
| 4077.72   | Fe I       |               | 3x |               | 3x |
| 4101.75   | H $\delta$ | -6.854        | 8x | -17.770       | 8x |
| 4132.07 ? | Fe I       |               | 4x |               | 4x |
| 4199.11 ? | Fe I       | -16.815       | 8x | 19.821        | 8x |
| 4226.74   | Fe I       | -15.236       | 8x | 11.765        | 8x |
| 4271.77   | Fe I       | 24.826        | 8x | 38.929        | 8x |
| 4307.91 ? | Fe I       | -58.423       | 8x | -45.114       | 8x |
| 4325.78   | Fe I       | -36.272       | 5x | -22.504       | 5x |
| 4340.48   | H $\gamma$ | -12.793       | 7x | 22.136        | 8x |
| 4383.56   | Fe I       | 39.778        | 8x | 75.409        | 8x |
| 4404.76   | Fe I       | -15.624       | 8x | 6.375         | 8x |

Linien, deren Identifikation auf Grund von Blends unsicher sind, wurden mit einem Fragezeichen gekennzeichnet und für Linien mit einer Häufigkeit kleiner als fünf keine Radialgeschwindigkeit angegeben.

Beschränkt man sich allerdings beim Fe-Spektrum auf die wirklich gut meßbaren Linien Ca H u. Ca K sowie H $\delta$  und H $\gamma$ , dann erhält man mit einer Radialgeschwindigkeit von  $v_R = 0,20 \pm 11,93$  km/s (theor.  $v_R = -0,04$  km/s), ein Ergebnis welches trotz aller Probleme und Fehlerquellen für einen Stern späteren Spektraltyps, wie es die Sonne ist, brauchbar ist.

#### 4. Ergebnisse und Ausblick

Als Ergebnis der Diplomarbeit ist festzuhalten, daß mit dem außeroxialen Schmidtkamerasystem ein vielseitiges, für Aufnahmen mit geteilter Pupille gut geeignetes, Kamerasystem geschaffen worden ist. Leider konnte diese Vielseitigkeit aus Zeitmangel bisher nur ansatzweise mit ein paar gefilterten Direkt- bzw. Doppelkeilplatten-Aufnahmen getestet werden, deren oberflächliche Auswertung auch gute Photometrische Eigenschaften versprechen zu scheinen.

Die Möglichkeit mittels Umlenkspiegel im Adapterstück, ein Objektiv als Sucher zusätzlich anschließen zu können, hat sich, obwohl bisher ohne Lochplatten gearbeitet wurde, als sehr nützlich erwiesen, wenn es darum ging, das Bildfeld für die Objektivprismen-Aufnahme und den dazugehörenden Leitstern festzulegen.

Auch die Verbreiterungen mit der Kassette der Schmidtkamera hat sich bewährt, wenn auch die Rasterung des Verbreiterungsmechanismus etwas Fingerspitzengefühl verlangt, damit die Spektren nicht streifig werden.

Auch im zweiten Teil wurden die nötigen Untersuchungen so umfangreich, daß durch sie die zeitlichen Grenzen einer Diplomarbeit überschritten wurden. Aus diesen Gründen konnten leider weder die Prüfung der Handhabung von Lochplatten mit Lochdurchmessern von ca 0,2 mm durchgeführt werden (s.3.0) noch die Suche nach einer geeigneten Lichtquelle für die Referenzaufnahmen (s.3.3) abgeschlossen werden. Ebenfalls unterbleiben mußten Stabilitätsuntersuchungen, in wie weit Referenzaufnahmen, die auf getrennter Photoplatte aufgenommen werden, zusätzliche Fehler verursachen.

Trotz der fehlenden abschließenden Untersuchungen kann schon jetzt festgehalten werden, daß das neue Auswerte-Verfahren gute Ergebnisse liefern wird. Der Fehler in der Koinzidenzposition kann mit etwas Aufwand (korrekte Normalisierung) auf umgerechnet ca. 2 km/s herabgedrückt werden, so daß die Hauptschwierigkeit in der Bestimmung der Dispersionskurve aus dem Referenz-Spektrum liegt. Es besteht daher die Berechtigung anzunehmen, daß der Fehler in der Radialgeschwindigkeit, bei Lösung dieses Problems nur um eben diese 2 km/s größer sein wird, als der eines Spaltspektrographen vergleichbarer Auflösung. Da jedoch die Objektivprismen-Spektroskopie gegenüber der Spalt-Spektroskopie den Geschwindigkeitsvorteil hat, daß die Radialgeschwindigkeiten mehrerer Objekte im Feld gleichzeitig bestimmt werden, kann dieser etwas größere Fehler durch die Gewinnung der Radialgeschwindigkeit aus mehreren Aufnahmen kompensiert werden.

Eine mögliche Lösung für das Problem der geeigneten Referenz-Lichtquelle, könnte die Verwendung der Eisen-Bogenlampe vom Universal-Astro-Gitterspektrographen sein. Diese liefert das Eisen-Spektrum mit hoher Lichtintensität, so daß sowohl die Linienanzahl genügend hoch als auch die Belichtungszeit ausreichend kurz sein wird.

Eine weitere Verbesserung der Genauigkeit der Dispersionskurve wäre durch eine Verbesserung der spektralen Reinheit der Referenz-Spektren zu erreichen. Im Gegensatz zu einer Sternaufnahme wird hier nämlich der Spalt bzw. das Loch der Maskenplatte abgebildet, was bei einem Spalt von 0,2 mm eine Verschmierung einer Linie über 40  $\mu\text{m}$  oder umgerechnet für das Doppelgitterprisma II eine Linienbreite von mehr als 7,9  $\text{\AA}$  bewirkt.

Eine Halbierung des Lochdurchmessers bzw. der Spaltbreite könnte eine deutliche Verbesserung bringen, jedoch ist es dazu nötig den Verbreiterungsmechanismus auf 20 Schritte a 15  $\mu\text{m}$  umzubauen oder mit Spaltplatten mit Spalten von 1,5x0,1 mm zu arbeiten. Die Frage die dafür geklärt werden müßte ist jedoch ob Maskenplatten mit solch feinen Öffnungen exakt genug hergestellt werden und nachher für die Aufnahme einjustiert werden können.

Gleichzeitig zeigen diese Überlegungen, daß, um die neue Methode sinnvoll anwenden zu können, keine bedeutend größeren Lochdurchmesser gewählt werden dürfen, da sonst die Bestimmung der Dispersionskurve durch Blendlinien verschlechtert wird.

## A. Literatur

- Becker, H.J.: 1979, Diplomarbeit, Bonn  
Becker, M.: 1978, Diplomarbeit, Bonn  
Cochran, W.D.: 1984, in "Proceedings of the Workshop on  
improvements to Photometry"  
NASA Conf.Public 2350, 33
- Dettmar, R.-J.: 1981, Diplomarbeit, Bonn  
ESO Image Processing Group:  
1988, MIDAS Users Guide Volume A: System  
MIDAS Release 88JAN
- Fehrenbach, Ch.: 1947, Ann. Astrophys. 10, 257, 306  
Fehrenbach, Ch.: 1964, Mitt. Astron. Ges.  
Geyer, E.H.: 1981, Mitt. Astron. Ges. 54, 156  
Geyer, E.H., Hoffmann, M., Nelles, B.:  
1979, Astron. Astrophys. 80, 24  
Geyer, E.H., Nelles, B.:  
1984, in "Very Large Telescopes, their  
Instrumentation and Programms"  
IAU Coll. 79, 575
- Geyer, E.H., Nelles, B.:  
1985, Astron. Astrophys. 148, 312
- Geyer, E.H., Schmidt, H.:  
1976, Mitt. Astron. Ges. 40, 125
- Giesecking, F.: 1976, Astron. Astrophys. 47, 43  
Hodam, F.: 1967, "Technische Optik", Berlin  
Kurucz, R.L., Furenlid, I., Brault, J., Testermann, L.:  
1984, "Solar Flux Atlas from 296 to 1300 nm",  
National Solar Obs. Atlas No 1
- Lentes, F.T.: 1980, Diplomarbeit, Bonn  
Moore, Ch.E., Minnaert, M.G.J., Houtgast, J.:  
1966, "The Solar Spectrum 2935 Å to 8770 Å",  
National Bureau of Standards, Monogr. 61
- Nelles, B.: 1978, Diplomarbeit, Bonn  
Nelles, B.: 1981, Mitt. Astron. Ges. 54, 186  
Nelles, B.: 1983, Dissertation, Bonn  
Nelles, B.: 1985, Optik 69, 153
- Orbinski, A.: 1895, Astron. Nachr. 138, 9  
Pickering, E.C.: 1896, Astron. Nachr. 142, 105  
Schwarzschild, K.: 1913, Publ. Astrophys. Obs. Potsdam, Nr 69  
Secci, A.: 1878, in "Die Sterne", Brockhaus, Leipzig  
Seitter, W.C.: 1970, "Atlas für Objektiv Prismen Spektren"  
Bonner Spektralatlas I, Dümmler, Bonn
- Waland, R.L.: 1964, in "Amateur Telescope Making III",  
Kingsport, S.342ff
- Wells, D.C., Greisen, E.W., Harten, R.H.:  
1981, Astron. Astrophys. Suppl. Ser. 44, 363
- Willmarth, D.: 1987, "Radial Velocities from CCD Spectra"  
Zaidel', A.N., Prokof'ev, V.K., Raiskii, S.M. et al.:  
1970, "Tables of Spectral Lines", IFI/Plenum,  
New York, London

B. Tabelle der in der Arbeit verwendeten Spektren

In der Diplomarbeit werden für Abbildungen und Ergebnistabellen folgende Doppelgitterprismen-Spektren der 4. Aufnahmeserie (aufgenommen 28.12.88, ausgemessen 03.02.89) unter ihrem Dateinamen benutzt:

- T4SP1-4: Tageslicht auf IIIa-J Platte lin. gemessen nach Opt.
- T4SV1-4: Tageslicht auf IIIa-J Platte lin. gemessen 2x verst.
- T4SL1-4: Tageslicht auf IIIa-J Platte log. gemessen nach Opt.
- T4CS1-4: Tageslicht auf IIa-O Platte lin. gemessen nach Opt.
- A4SL1-4: He/Ar-Glimmlampe auf IIIa-J Platte log. gem. nach Opt.
- C4SL1-4: Cd-Spektrallampe auf IIIa-J Platte log. gem. nach Opt.
- E4SL1-4: Fe-Hohlkathode auf IIIa-J Platte log. gem. nach Opt.
- H4SL1-4: He-Spektrallampe auf IIIa-J Platte log. gem. nach Opt.
- R4SL1-4: Hg-Spektrallampe auf IIIa-J Platte log. gem. nach Opt.
  
- N4SL1-4: s.T4SL1-4, geglättet mit Smooth-Filter 200,0,0
- T4SN1-4: s.T4SL1-4, normalisiert durch Abzug von N4SL1-4
- T4SL4a : s.T4SL1-4, 2. Messung direkt nach T4SL4
- T4SL4b : s.T4SL1-4, 3. Messung nach Anfahren einer anderen Pos.

Ein an einen Namen angehängtes kleines-e weist darauf hin, daß dieses Spektrum auf einen "gültigen" Bereich eingeschränkt wurde. Die gewählten Positionen 1-4 entnehme man dem Schema der Vielfachspaltplatte (Abb 16).

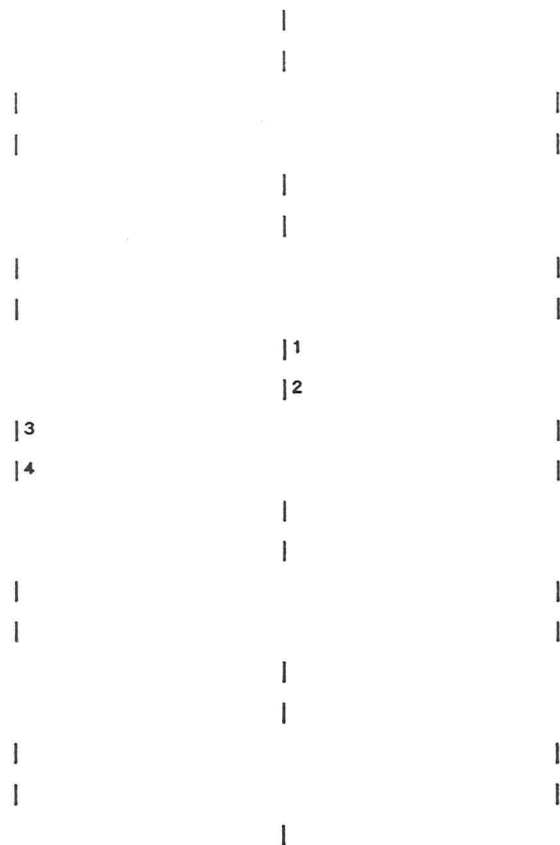


Abb 16: Zuordnung der Doppelgitterprismen-Spektren zu den Spalten der Vielfachspaltplatte

## C. Programmbeschreibung

### C.1 Auswertungsroutinen in MIDAS

#### C.1.1 Routine DGPMAIN.PRG

```
!++
!   Prozedur: DGPMAIN
!
!   Zweck   : Steuerung der Bestimmung der Koinzidenzpositionen
!             von Stern- und Referenz-Spektrum sowie Aufruf der
!             Wellenlängeneichung mit Radialgeschwindigkeits-
!             bestimmung
!
!   Version : 1.0   Nov.88   (T. Mainka)
!
!   Param.  : P1 - Name des Sternspektrums
!
!             P2 - Name des Referenz-Spektrums
!
!             P3 - Namenspräfix für Linien-Tabellen
!
!             P4 - Flagge für Reduzierte Ausgabe
!
!             P5 - Ausgabezeile des Ergebnisses in den Tabellen
!                   DIPL und ERGB
!
!   Keys    : CO (C/1/1)
!             - Flagge zur Kontrolle der reduzierten Ausgabe
!
!             DELTA (R/1/1)
!             - Verschiebung der Koinzidenzposition von der
!               Mittenposition
!
!             SGRENZE (R/1/2)
!             - Start- und Endposition des gültigen Bereichs im
!               Sternspektrum
!
!             RGRENZE (R/1/2)
!             - Start- und Endposition des gültigen Bereichs im
!               Referenz-Spektrum
!
!             KPOS (R/1/3)
!             - Koinzidenzposition des Sternspektrums
!
!             RPOS (R/1/3)
!             - Koinzidenzposition des Referenz-Spektrums
!++

!+
!   Abfrage und Definition der Parameter und Initialisierung des
!   Ausgabeformats
!-
DEFINE/PAR P1 ? I "Spektrum:"
DEFINE/PAR P2 ? I "Referenz:"
DEFINE/PAR P3 STL ? "Output-Table:"
DEFINE/PAR P4 N C "Red. Output:"
DEFINE/PAR P5 ? C "Output-Row"
SET/FORMAT I8 G18.8
SET/PLOT XAXIS:AUTO YAXIS:AUTO
WRITE/KEY CO/C/1/1 'P4'
```

```
!+
! Aufteilung des Doppelgitterprismen-Sternspektrums in normales
! und revertiertes Spektrum sowie Spiegelung des revertierten
! Spektrums um seine Mittenposition
!-
EXTRACT/LINE SPN = 'P1'[@1,@1:@5000,@1]
EXTRACT/LINE SPR = 'P1'[@1,@2:@5000,@2]
FLIP/IMA SPR X
IF CO .EQ. "N" THEN
  PLOT/IMA SPN
  OVERPLOT/IMA SPR
ENDIF
!+
! Grobbestimmung der Koinzidenzposition
!-
@@ KOINZ SP 250 20 1.E04
!+
! Definition der Keys für die Positionsverschiebung des normalen
! Sternspektrums
!-
WRITE/KEY H3/R/1/1 -5000.
WRITE/KEY H2/R/1/1 'SPN,STEP(2)'
WRITE/KEY H1/R/1/1 'SPN,STEP(1)'
COMPUTE/KEY DELTA = M$NINT('DELTA'/(2*'H1'))*'H1'
COMPUTE/KEY H3 = 'H3'-'SPN,START(1)''-DELTA'
!+
! Verschiebung des normalen Stern-Spektrums auf die
! Koinzidenz-Grobposition
!-
REBIN/LIN SPN SPNA 'H1','H2' 0.,0. 'H3',0.
!+
! Definition der Keys für die Positionsverschiebung des
! revertierten Sternspektrums
!-
WRITE/KEY H3/R/1/1 -5000.
COMPUTE/KEY H3 = 'H3'-'SPR,START(1)'+DELTA'
WRITE/KEY KPOS/R/1/3
'SPN,START(1)',{'{'P1'}',START(2)'},{{'P1'},STEP(2)'}
KPOS(1) = KPOS(1)+5000+'DELTA'
!+
! Verschiebung des revertierten Stern-Spektrums auf die
! Koinzidenz-Grobposition
!-
REBIN/LIN SPR SPRA 'H1','H2' 0.,0. 'H3',0.
!+
! Abfrage des Gültigkeitsbereiches des Sternspektrums
!-
IF M$EXISTK("SGRENZE") .EQ. 0 THEN
  PLOT/IMA SPNA
  OVERPLOT/IMA SPRA
  WRITE/KEY SGRENZE/R/1/2 -1000.,1000.
  INQUIRE SGRENZE "Bitte linke und rechte Grenze eingeben : "
ENDIF
!+
! Beschränkung des normalen und revertierten Stern-Spektrums auf
! den Gültigkeitsbereich und Feinbestimmung der
! Koinzidenzposition
!-
EXTRACT/IMA SPN = SPNA['SGRENZE(1)':'SGRENZE(2)']
EXTRACT/IMA SPR = SPRA['SGRENZE(1)':'SGRENZE(2)']
@@ KOINZ SP 50 20 1.E04
```



```
!+
! Bestimmung der Keys für die Positionsverschiebung und
! Verschiebung der Sternspektren auf Feinposition
!-
WRITE/KEY H2/R/1/1 'SPN,STEP(2)'
WRITE/KEY H1/R/1/1 'SPN,STEP(1)'
COMPUTE/KEY H3 = -'DELTA'/2
REBIN/LIN SPN SPNA 'H1','H2' 0.,0. 'H3',0.
COMPUTE/KEY H3 = +'DELTA'/2
KPOS(1) = KPOS(1)+'DELTA'/2
REBIN/LIN SPR SPRA 'H1','H2' 0.,0. 'H3',0.
!+
! Aufteilung des Doppelgitterprismen-Referenz-Spektrums in
! normales und revertiertes Spektrum sowie Spiegelung des
! revertierten Spektrums um seine Mittenposition
!-
EXTRACT/LINE REN = 'P2'[@1,@1:@5000,@1]
EXTRACT/LINE RER = 'P2'[@1,@2:@5000,@2]
FLIP/IMA RER X
IF CO .EQ. "N" THEN
    PLOT/IMA REN
    OVERPLOT/IMA RER
ENDIF
!+
! Grobbestimmung der Koinzidenzposition
!-
@@ KOINZ RE 200 20 1.E06
!+
! Definition der Keys für die Positionsverschiebung des normalen
! Sternspektrums
!-
WRITE/KEY H3/R/1/1 -5000.
WRITE/KEY H2/R/1/1 'REN,STEP(2)'
WRITE/KEY H1/R/1/1 'REN,STEP(1)'
COMPUTE/KEY DELTA = M$NINT('DELTA'/(2*'H1'))*'H1'
COMPUTE/KEY H3 = 'H3'-'REN,START(1)''-'DELTA'
!+
! Verschiebung des normalen Referenz-Spektrums auf die
! Koinzidenz-Grobposition
!-
REBIN/LIN REN RENA 'H1','H2' 0.,0. 'H3',0.
!+
! Definition der Keys für die Positionsverschiebung des
! revertierten Referenz-Spektrums
!-
WRITE/KEY H3/R/1/1 -5000.
COMPUTE/KEY H3 = 'H3'-'RER,START(1)'+ 'DELTA'
WRITE/KEY RPOS/R/1/3
'RER,START(1)',{'{'P2'},START(2)'},{{'P2'},STEP(2)'}
RPOS(1) = RPOS(1)+5000+'DELTA'
!+
! Verschiebung des revertierten Referenz-Spektrums auf die
! Koinzidenz-Grobposition
!-
REBIN/LIN RER RERA 'H1','H2' 0.,0. 'H3',0.
```

```
!+
! Abfrage des Gültigkeitsbereiches des Referenz-spektrums
!-
IF M$EXISTK("RGRENZE") .EQ. 0 THEN
  PLOT/IMA RENA
  OVERPLOT/IMA RERA
  WRITE/KEY RGRENZE/R/1/2 -1000.,1000.
  INQUIRE RGRENZE "Bitte linke und rechte Grenze eingeben : "
ENDIF
!+
! Beschränkung des normalen und revertierten Referenz-Spektrums
! auf den Gültigkeitsbereich und Feinbestimmung der
! Koinzidenzposition
!-
EXTRACT/IMA REN = RENA['RGRENZE(1)':'RGRENZE(2)']
EXTRACT/IMA RER = RERA['RGRENZE(1)':'RGRENZE(2)']
@@ KOINZ RE 50 20 1.E06
!+
! Bestimmung der Keys für die Positionsverschiebung und
! Verschiebung der Sternspektren auf Feinposition
!-
WRITE/KEY H2/R/1/1 'REN,STEP(2)'
WRITE/KEY H1/R/1/1 'REN,STEP(1)'
COMPUTE/KEY H3 = -'DELTA'/2
REBIN/LIN REN RENA 5.,'H2' 0.,0. 'H3',0.
COMPUTE/KEY H3 = +'DELTA'/2
RPOS(1) = RPOS(1)+'DELTA'/2
REBIN/LIN RER RERA 5.,'H2' 0.,0. 'H3',0.
!+
! Start der Wellenlängeneichung für die normalen Spektren
!-
@@ WELEICH REN SPN 'P3'N 'P5'
!+
! Start der Wellenlängeneichung für die revertierten Spektren
!-
KPOS(2) = KPOS(2)+KPOS(3)
RPOS(2) = RPOS(2)+RPOS(3)
@@ WELEICH RER SPR 'P3'R 1'P5'
!++
! Ende von DGPMAIN
!++
```

C.1.2 Routine KOINZ.PRG

```
!++
!   Prozedur: KOINZ
!
!   Zweck   : Bestimmung der Koinzidenz-Position von normalen und
!             revertiertem Spektrum
!
!   Version : 1.1   Nov.88   (T. Mainka)
!
!   Param.  : P1 - Name des zu bearbeitenden DGP-Spektrums
!             (Benutzt als Spektren-Typ-Identifikation:
!             SP=Stern-Spektrum, RE=Referenz-Spektrum)
!
!             P2 - Halbe Bandbreite des Least-Square-Fit in
!             Pixelkoordinaten
!
!             P3 - Größe des Suchfensters zur Bestimmung der
!             Fehler-Minima
!
!             P4 - Größe des Schwellenwerts zur Bestimmung der
!             Fehler-Minima
!
!   Keys    : CO (C/1/1)
!             - Flagge zur Kontrolle der reduzierten Ausgabe
!
!             NUMMER (I/1/1)
!             - Variable zur Auswahl des richtigen Fehler-
!             Minimums
!
!             DELTA (R/1/1)
!             - Variable zur Ausgabe der Verschiebung der
!             Koinzidenzposition von der Mittenposition
!
!++

!+
!   Abfrage und Definition der Parameter
!-
DEFINE/PAR P1 ? C "Enter Type-letters of frames/tables:"
DEFINE/PAR P2 ? N "1/2 bandwith = shift of frames:"
DEFINE/PAR P3 ? N "window = size of search line"
DEFINE/PAR P4 ? N "thresholt = detecting thresholt"
!+
!   Definition der Keys für das Programm LSQUARE
!-
WRITE/KEY IN_A 'P1'N           ! normales Spektrum
WRITE/KEY IN_B 'P1'R           ! revertiertes Spektrum
WRITE/KEY OUT_A 'P1'TX        ! LS-Fit Tabelle
WRITE/KEY INPUTI/I/1/1 'P2'    ! Bandbreite des Fit's
WRITE/KEY HISTORY "LSQUARE/IMA 'P1'N 'P1'R 'P1'TX 'P2'"
!+
!   Start von LSQUARE
!-
RUN DISK$USER:[MAINKA]LSQUARE
!+
!   Erstellung eines Tabellen-Histogramms zur Bestimmung des
!   Maximal- und Minimalwertes
!-
READ/HIST 'P1'TX #2
```

```
!+
! Berechnung der Y-Plotachse aus Minimal- und Maximalwert
!-
INPUTR(2) = (OUTPUTR(4)-OUTPUTR(3))/5
INPUTR(3) = (OUTPUTR(4)-OUTPUTR(3))
SET/PLOT YAXIS=0.,'INPUTR(3)', 'INPUTR(2)'
!+
! Definition der Keys zur Umwandlung der Tabelle in ein Bild
!-
WRITE/KEY DELTA/R/1/1 1           ! Definition von Delta
WRITE/KEY IN_A 'P1'N             ! normales Spektrum
WRITE/KEY IN_B 'P1'R             ! revertiertes Spektrum
WRITE/KEY OUT_A 'P1'TX           ! LS-Fit Tabelle
WRITE/KEY OUT_B 'P1'IX           ! LS-Fit Bild
DELTA(1) = {'{'IN_A'},STEP(1)'} ! Abstand der Pixel
WRITE/KEY INPUTI/I/1/1 'P2'
INPUTR(1) = -INPUTI(1)*DELTA(1) ! Startwert des Bildes
INPUTI(1) = (INPUTI(1)*2)+1      ! Anzahl der Pixel
!+
! Erzeugung eines Referenzbildes REF und Umwandlung der
! Tabelle in ein Bild
!-
CREATE/IMA REF 1,'INPUTI(1)' 'INPUTR(1)', 'DELTA'
CONVERT/TAB 'OUT_B' = 'OUT_A' #1 #2 REF PLOT
!+
! Löschung überflüssiger Referenzbilder
!-
$ PURGE REF.BDF
!+
! Umrechnung des LS-Fit Bildes in ein Übereinstimmungs Bild um
! Maximums Linien-Suche zu ermöglichen (ev. Ausgabe des Bildes)
!-
COMPUTE 'OUT_B' = -'OUT_B'+ 'OUTPUTR(4)'
IF CO .EQ. "N" THEN
  PLOT/IMA 'OUT_B'
ENDIF
!+
! Suche nach Übereinstimmungsmaxima
!-
SEARCH/LINE 'OUT_B' 'P3','P4' XTM
!+
! Überprüfung der Anzahl der gefundenen Übereinstimmungsmaxima
! und Abfrage des richtigen Maxima bei mehr als einem Maximum
!-
WRITE/KEY NUMMER/I/1/1 1
IF 'XTM.TBL,TBLCONTR(4)' .GT. 1 THEN
  IF CO .NE. "N" THEN
    PLOT/IMA 'OUT_B'
  ENDIF
  OVERPLOT/TAB XTM #1 #3
  READ/TAB XTM #1 #3
  NUMMER = 0
EING:
  INQUIRE/KEY NUMMER "Wo ist die Koinzidenz:"
  IF NUMMER .EQ. 0 GOTO EING
ENDIF
!+
! Übergabe der Abweichung von Koinzidenzposition von der
! Mittenposition an DELTA und Rücksetzung der Y-Plotachse
!-
DELTA(1) = {'XTM,:X,{'NUMMER'}}'
SET/PLOT YAXIS:AUTO
```

C.1.3 Routine WELEICH.PRG

```
!++
!   Prozedur: WELEICH
!
!   Zweck   : Wellenlängeneichung der auf die Koinzidenzposition
!             verschobenen Sternspektren aus Referenz-Spektren
!
!   Version : 1.1   Jan.89   (T. Mainka)
!
!   Param.  : P1 - Name des Referenzspektrums
!             P2 - Name des Sternspektrums
!             P3 - Name der Linien-Tabelle
!             P4 - Ausgabezeile des Ergebnisses in den Tabellen
!                 DIPL und ERGB
!
!   Keys    : CO (C/1/1)
!             - Flagge zur Kontrolle der reduzierten Ausgabe
!
!             DISPCOE (R/1/8)
!             - Dispersionskoeffizienten aus Referenzspektrum
!
!             KPOS (R/1/3)
!             - Koinzidenzposition des Sternspektrums
!
!             RPOS (R/1/3)
!             - Koinzidenzposition des Referenz-Spektrums
!++

!+
!   Liniensuche im Referenz-Spektrums und Ausgabe der Linien
!-
COMPUTE/IMA 'P1'B = -'P1'A
SEARCH/LINE 'P1'B 32,100 RTL GAUSSIAN EMISSION
IF CO .EQ. "N" READ/TAB RTL #1 #3
PLOT 'P1'B
OVERPLOT/TAB RTL #1 #3
!+
!   Prüfung auf vorhandene korrekte Linien-Identifikation
!
IF M$EXIST("ID.TBL") .EQ. 1 THEN
  WRITE/KEY HI/I/1/2 'ID.TBL,TBLCONTR(4)', 'RTL.TBL,TBLCONTR(4)'
  IF HI(1) .EQ. HI(2) THEN
    COPY/TT ID :IDENT RTL :IDENT
    OVERPLOT/IDENT RTL :X :IDENT
    WRITE/KEY C1/C/1/1 N
    INQUIRE C1 "In Ordnung [J/N]"
    IF C1 .EQ. "J" GOTO CAL
  ENDIF
ENDIF
!+
!   Linien-identifikation mit dem Graphik-Cursor und Abspeicherung
!   Identifikation für spätere Spektren
!-
IDENTIFY/GCURSOR RTL :IDENT :X 30
CREATE/TAB ID 1 800 NULL
COPY/TT RTL :IDENT ID :IDENT
CAL:
```

```
!+
! Errechnung der Dispersionskoeffizienten aus den Linien des
! Referenz-Spektrums und Abspeicherung in der Tabelle DIPL
!-
CALIBRATE/LINE 10,2 RTL EICH IDENT
SET/FORMAT I4 G20.10
DIPL,:D0,'P4' = 'DISPCOE(1)'
DIPL,:D1,'P4' = 'DISPCOE(2)'
DIPL,:D2,'P4' = 'DISPCOE(3)'
!+
! Errechnung der Koinzidenzwellenlänge des Referenzspektrums
! zu Prüfzwecken und Abspeicherung in den Tabellen DIPL und ERGB
!-
H1 = -'DISPCOE(2)'/((2*'DISPCOE(3)')
H2 = -'DISPCOE(1)'/ 'DISPCOE(3)'
H3 = H1-M$EXP(M$LN((H1*H1)+H2))/2
DIPL,:RKWL,'P4' = 'H3'
ERGB,:RKWL,'P4' = 'H3'
!+
! Abspeicherung der Koinzidenzpositionen in der Tabelle DIPL
!-
DIPL,:SKPX,'P4' = 'KPOS(1)'
DIPL,:SKPY,'P4' = 'KPOS(2)'
DIPL,:RKPX,'P4' = 'RPOS(1)'
DIPL,:RKPY,'P4' = 'RPOS(2)'
!+
! Wellenlängenkalibration des Sternspektrums und Suche der
! Linien im Sternspektrum
!-
SEARCH/LINE 'P2'A 32,100 'P3'A GAUSSIAN EMISSION
REBIN/WAVE 'P2'A 'P2'B REF350B
PLOT 'P2'B
SEARCH/LINE 'P2'B 30,100 'P3' GAUSSIAN EMISSION
IF CO .EQ. "N" READ/TAB 'P3' #1 #3
OVERPLOT/TAB 'P3' #1 #3
!+
! Löschung der :Y Spalte in der Linien-Tabelle des Sternspektrums
!-
WRITE/KEY N/I/1/1 0
WRITE/KEY M/I/1/1 {'{'P3'}.TBL,TBLCONTR(4)'}
DO N = 1 'M'
    'P1',:Y,'N' = *
ENDDO
!+
! Identifikation der Spektrallinien
!-
IDENT/GCOURSE 'P3' :Y :X 20
WRITE/KEY C1/C/1/1 N
INQUIRE C1 "In Ordnung [J/N]"
IF C1 .EQ. "J" GOTO VEL
    OVERPLOT/IDENT 'P3' :X :Y
    IDENT/GCOURSE 'P3' :Y :X 20
VEL:
!+
! Berechnung der Radialgeschwindigkeit aus der Linienverschiebung
! und Speicherung des Mittelwertes sowie Standardabweichung in
! der Tabelle ERGB
!-
COMP/TAB 'P3' :GE = (:X-:Y)/:Y*299792.458
READ/HIST 'P3' :GE
ERGB,:RGES,'P4' = 'OUTPUTR(1)'
ERGB,:STAB,'P4' = 'OUTPUTR(2)'
```

C.2 Fortran-Programme für MIDAS

C.2.1 Programm LSQUARE.FOR

PROGRAM LSQUARE

C

C-----

C

C.IDENTIFICATION

C

C program LSQUARE (ehemals XCORR)

C

version 1.0 870602

C M. Rosa

ESO - Garching

C

C.KEYWORDS

C

C least-square-fit, spectral reduction

C

C.INPUT/OUTPUT

C

C Input : IN\_A        TEMPLATE

C        IN\_B        SPECTRUM

C        OUT\_A       CORRELATION TABLE

C        INPUTI      SHIFT IN PIXELS

C

C.VERSIONS

C 1.10                    clean up a little bit... K.Banse (ESO)

C 1.20                    History-Descriptor        T.Mainka (Bonn)

C 2.00                    Least-Square-Fit         T.Mainka (Bonn)

C-----

C

IMPLICIT NONE

C

CHARACTER\*72            CUNITA,CUNITB,IDENTA,IDENTB  
CHARACTER\*16            LABEL(2),UNIT(2),TABFULL  
CHARACTER                TEMPLA\*60,SPECTR\*60,CTAB\*60,HISTORY\*80  
CHARACTER                IDENT\*60  
CHARACTER\*6              TYPE,FMT

C

INTEGER\*4                NAXISA,NAXISB,IAC,NPOI,ISHI,I  
INTEGER\*4                NPIX(3),NPIXB(3),PNTRA,PNTRB,IST  
INTEGER\*4                NCOL,NROW  
INTEGER\*4                ICOL(2),INDEX

C

REAL\*4                   STEPA(3),STARTA(3),STEPB(3),STARTB(3)

C

DATA LABEL(1)/'XPOS'/, LABEL(2)/'CORRVAL'/  
DATA TYPE/'R\*4'/, FMT/'E12.4'/

C

C ... INITIALIZE PARAMETERS

C

CALL PROLOG\_ST('XCORR')  
CALL GETKEY\_ST('IN\_A', 'C', 1, 60, IAC, TEMPLA, IST)  
CALL GETKEY\_ST('IN\_B', 'C', 1, 60, IAC, SPECTR, IST)  
CALL GETKEY\_ST('OUT\_A', 'C', 1, 60, IAC, CTAB, IST)  
CALL GETKEY\_ST('INPUTI', 'I', 1, 2, IAC, ISHI, IST)

C

CALL GETIMAG\_ST(TEMPLA,'I', 2, NAXISA, NPIX(3), STARTA, STEPA,  
IDENTA, CUNITA, PNTRA, IST)  
CALL GETIMAG\_ST(SPECTR,'I', 2, NAXISB, NPIXB(3), STARTB, STEPB,  
IDENTB, CUNITB, PNTRB, IST)

C

```
      IF( (NAXISA.NE.NAXISB) .OR. (NPIX(1).NE.NPIXB(1)) )
+     CALL ERRTERM_ST(1,'Frames not matching...')
C
C ... How many points will we have in overlap region
C
      NPOI = NPIX(1)-ISHI-ISHI
      IF(NPOI.LT.3)
+     CALL ERRTERM_ST(2,'Too little overlap '//
+     '- shift too large or frames too short')
C
      I = INDEX(CTAB,' ') - 1
      TABFULL = CTAB(1:I)//'.TBL'
      NCOL = 2
      NROW = ISHI+ISHI+1
      UNIT(1) = CUNITA(1:16)
      UNIT(2) = '          ',
      CALL TBL_INIT(CTAB,NCOL,NROW,IST)
      DO I=1,2
        CALL TBL_DEFCOL(CTAB,TYPE,FMT,UNIT(I),LABEL(I),ICOL(I),IST)
      ENDDO
C
C ... Call SXCHI2relation routine
C
      CALL SX_CHI2(NPIXA,%VAL(PNTRA),NPIXB,%VAL(PNTRB),STPA(1),
+     CTAB,NROW,ICOL,TYPE,ISHI)
C1.20
      I = INDEX(TEMPLA,' ') + 1
      IDENT = 'Least-Square-Fit of '//TEMPLA(1:I)//'and '//SPECTR
      CALL PUTDSCR_ST(TABFULL,'IDENT','C',IDENT,1,60,IST)
      CALL GETKEY_ST('HISTORY','C',1,80,I,HISTORY,IST)
      CALL PUTDSCR_ST(TABFULL,'HISTORY','C',HISTORY,1,80,IST)
C1.20
      CALL TBL_FREE(CTAB,IST)
      CALL EPILOG_ST
C
      CALL EXIT
      END
```



```
      SUBROUTINE SX_CHI2 (NA,TEMP,NB,SPEC,STEP,  
      CTAB,NT,ICOL,TYPE,NSH)
```

```
C  
C This program computes the least-square-fit between a template  
C and a spectrum over a bandwidth of 2*NSH pixels. Identity of  
C start and step are assumed. We will have 2*NSH + 1 data.  
C  
C      IMPLICIT NONE  
C  
C      CHARACTER*(*)    CTAB  
C      CHARACTER*6      TYPE  
C  
C      REAL*4           TEMP(1),SPEC(1),STEP  
C      REAL*4           VALROW(2)  
C      REAL*4           MEAN(2)  
C  
C      INTEGER*4       NA,NB,NT  
C      INTEGER*4       ICOL(2),NSH,MPS,MPE,NCT,IST  
C      INTEGER*4       IROW,I,J  
C  
C      NCT = 2  
C      MPS = 1+NSH  
C      MPE = NA-NSH  
C      IROW = 0  
C      DO I = -NSH,NSH  
C          IROW = IROW+1  
C          VALROW(1) = FLOAT(I)*STEP  
C          VALROW(2) = 0.0  
C          DO J = MPS,MPE  
C              VALROW(2) = VALROW(2)+(SPEC(J)*TEMP(J+I))  
C              VALROW(2) = VALROW(2)+(SPEC(J)-TEMP(J+I))*(SPEC(J)-TEMP(J+I))  
C          ENDDO  
C          CALL TBL_WROW(CTAB,TYPE,IROW,NCT,ICOL,VALROW,IST)  
C      ENDDO  
C  
C      RETURN  
C      END
```

C.2.2 Programm PDSMID.FOR

```
C+++++
C
C
C          *****
C          * Program PDStoMIDAS *
C          *****
C
C
C Author :      P.Mueller                      1.0
C
C Version:      1.1 Korrektur der Descriptoren  T.Mainka
C               1.2 richtige Anzahl der Bloecke T.Mainka
C
C-----
```

```
PROGRAM PDS_TO_MIDAS
C
C  implicit integer (a-z)
C
C  parameter (n_b=2880, n_h=1440, n_w=720, nci=80, na=16)
C  parameter (nr=360)
C
C  CHARACTER*60 PDSFILE/'PDSDATA.DAT'/
C  CHARACTER*16 FILES
C  CHARACTER*80 DESCR,ATEXT(16384)
C  LOGICAL      SIMPLE,EDIT,MORE
C
C  CHARACTER*48 TEXT
C  character*60 INFILE,outfile
C
C  INTEGER*2 REPZAL,REPDMA
C  INTEGER*4 JDIM(2),ISTATUS,IZEIL
C  INTEGER*4 LIMPOINTS/14000/,NPOINTS
C  INTEGER*4 DIMBUF,DIMBYTBUF
C  PARAMETER (DIMBYTBUF=2*16383)
C  PARAMETER (DIMBUF=16383)
C  INTEGER*2 WORDBUF(DIMBUF) ! buffer for data
C  BYTE      BYTEBUF(DIMBYTBUF)
C  BYTE      B
C  REAL*8 XWERT
C  CHARACTER*1 C
C
C  REAL*4      FMIN, FMAX, CUTS(6), CRPIX(NA)
C  REAL*4      BSCALE, BZERO, STEP(NA), CRVAL(NA)
C  INTEGER*4   JREC(N_W), NPIX(NA)
C  INTEGER*2   IREC(N_H)
C  CHARACTER*(NCI)      CREC(NR)
C  BYTE        BREC(N_B)
C  INTEGER*4   ACTVAL, BITPIX, CHANL, COUNT, ENDINDX
C  INTEGER*4   FEND, FHERE, FSTART, GCOUNT, I, IERR, IEXT
C  INTEGER*4   II, INTINDX, IRECORD,ISTAT,L, LENF, LENINT
C  INTEGER*4   LENL, LFID, N, NAXIS, NBITS, NBL, NCOM
C  INTEGER*4   NDIV, NHIS, NREC, NRECORD, NUMDONE, OFFSET
C  INTEGER*4   PCOUNT, REMAIN, SCAN_FLAG, SIZE
C  INTEGER*4   SKIP, SLEN, SPECEND, SPECOFF, SS, STATUS
C  INTEGER*4   SUM, TFIELDS, WASTE, NCOUNT, NBYTES
C  INTEGER*4   IRECBL,LRECBL,COUNT1,AXIS(4)
C  integer*4   amap
C  CHARACTER*(N_B) AREC
C  CHARACTER   CUNIT*192, MSG01*50, MSG02*50, OUTPUT*80
C  CHARACTER   FILE_SPECS*60, INTVAL*10, FFILE*4,LFILE*4
```

```
CHARACTER          FLAG*1, DEVICE*60, FILE1*12
CHARACTER          CBUF*8, FILE*12, NOFIL*5, TAPE*4, FID*4
CHARACTER          SCRATCH*2,MIDENT*30
CHARACTER*8        TNULL(128)
CHARACTER*80       TEXT0,TEXT1,TEXT2,TEXT3,TEXT4,NEWPAGE
CHARACTER*80       TEXT5
LOGICAL            ITABLE,IMAGE,PROCESS_IMAGE, PROCESS_TABLE
LOGICAL            IESO,IFORM,TAPE_FLAG,MTEOF,MTEOT
LOGICAL*1          KREC(2880)
REAL*4             TSCAL(128),TZERO(128)

C
common/fits01/ size,bitpix,bscale,bzero,fmin,fmax,tscal,tzero
common/fits02/ naxis,npix,step,crval,crpix,ncom,nhis,nbl,iext,
&              pcount,gcount,tfields,ieso,iform
common/fits03/ file1,mident,cunit,tnull
common/fits04/ irecb1,lrecb1
COMMON/FITS99/ INFILE,OUTFILE

C
EQUIVALENCE(WORDBUF(1997),REPZAL)
EQUIVALENCE(WORDBUF(2006),REPDMA)
equivalence (irec, jrec, brec)
equivalence (brec, crec)
equivalence (arec, crec)
equivalence (arec, krec)

C
include 'babmessage:babmessag.inc'

C
CALL PROLOG_ST('PDS_TO_MIDAS')

C
C -----
C
C prompt user for input file:
C
CALL GETKEY_ST('IN_A', 'C', 1, 60, ACTVAL, PDSFILE, STATUS)
call default_file_name(infile,'.DAT',infile)
STATUS=LIB$GET_LUN(DATALUN)
CALL BAB_CHECK(STATUS)
OPEN(UNIT=DATALUN,FILE=PDSFILE,STATUS='OLD',READONLY,
1FORM='UNFORMATTED',IOSTAT=IO_STAT)
CALL BAB_CHECK(IOSTAT_STATUS(IO_STAT))

C
READ(DATALUN,IOSTAT=IO_STAT)(WORDBUF(I),I=1,DIMBUF)
CALL BAB_CHECK(IOSTAT_STATUS(IO_STAT))
REWIND DATALUN
NDMAS=REPDMA
CALL PDS_HAD(CREC,DATALUN,JDIM)

C
C -----
C
CALL GETKEY_ST('OUT_A', 'C', 1, 60, ACTVAL, OUTFILE, STATUS)
call default_file_name(outfile,'.BDF',outfile)

C
C ... set up MIDAS environment+enable automatic error abort
C
MIDENT = ' '
OUTPUT = '6'
NCOUNT = 0
NBYTES = N_B

C
```

```
C ... initialize constants
C
      BSCALE = 1.
      BZERO  = 0.
      FMIN   = +999999.
      FMAX   = -999999.
      IERR   = 0
      NREC   = 0
      SIZE   = 0
      BITPIX = 0
      OFFSET = 0
      SUM    = 0
      NHIS   = 0
      NCOM   = 0
      NBL    = 0
      CUNIT  = ' '
      IRECBL = 0
      LRECBL = 0
      DO I=1, NA
          STEP(I) = 1.0
          CRVAL(I) = 1.0
          CRPIX(I) = 1.0
      ENDDO
C
C ... process main header
C
      SCAN_FLAG=0
      FLAG='S'
      DO WHILE (IERR.EQ.0)
          NREC = NREC + 1
          IF (NREC.EQ.1) CALL FNDBL(CREC,ITABLE)
          CALL DCFITS(outfile, CREC, NREC, IERR, Amap,SCAN_FLAG, FLAG)
      ENDDO
      IF (IERR.GT.0) THEN
          WRITE(SCRATCH, 10010) IERR
          MSG02(30:31) = SCRATCH
          CALL PUTTEXT_ST(MSG02,STATUS)
      ENDIF
      NDIV      = BITPIX/8
      NBITS     = BITPIX*GCOUNT*(PCOUNT+SIZE)
      NRECORD   = INT((NBITS+23039)/23040)
      PROCESS_IMAGE = IERR.LE.0 .AND. SCAN_FLAG.NE.1
C
C ... process BDF data
C
      WRITE IMAGE
      ISTART = 1
      CALL FILLARRAY(%VAL(AMAP),JDIM,WORDBUF,DIMBUF,REPZAL,ISTART)
C now do the same for the rest:
      DO J=1,NDMAS-1
          READ(DATALUN,IOSTAT=IO_STAT)(WORDBUF(I),I=1,DIMBUF)
          CALL BAB_CHECK(IOSTAT_STATUS(IO_STAT))
          CALL FILLARRAY(%VAL(AMAP),JDIM,WORDBUF,DIMBUF,REPZAL,ISTART)
      ENDDO
C
```

```
C ... end of image data
C
5000   IF (PROCESS_IMAGE.AND.SIZE.GT.0) THEN
        IF (SUM.LT.SIZE) CALL PUTTEXT_ST(MSG01,STATUS)
        CUTS(3) = FMIN
        CUTS(4) = FMAX
        CALL PUTDSCR_ST(outFILE,'LHCUTS','R*4',CUTS(3),3,2,STATUS)
        IF (FLAG.EQ.'S') THEN
            WRITE(OUTPUT,20200) outFILE,MIDENT,NAXIS,(NPIX(I),I=1,3)
        ELSE
            WRITE(OUTPUT,20000) outFILE
        ENDIF
        MIDENT = ' '
        CALL PUTTEXT_ST(OUTPUT,STATUS)
        CALL CLOSE_ST( outFILE, STATUS)
        NCOUNT = NCOUNT + 1
    ENDIF
C
        CALL EPILOG_ST
        CALL EXIT
C
C ... normal end
C
8000   CONTINUE
        WRITE(OUTPUT,40000) NCOUNT
        CALL PUTTEXT_ST(OUTPUT,STATUS)
C
C ... abnormal end
C
9000   CALL PUTTEXT_ST('Error in file read',STATUS)
        CALL EXIT
C
C formats -----
C
10000  FORMAT(I4.4)
10001  FORMAT(I<SLEN>)
10010  FORMAT(I2)
20000  FORMAT(' image ',A,' loaded')
20200  FORMAT(1X,A12,1X,A30,4(I6,1X))
30000  FORMAT(' table ',A,' loaded')
30300  FORMAT(1X,A12,1X,A30,4(I6,1X))
40000  FORMAT(' Total number of files: ',I7)
C
        END
C
```

C\*\*\*\*\*

C

    SUBROUTINE DCFITS(FILE,BREC,NREC,IERR,PNTR,LIST\_FLAG,FLAG)

C

    Decode FITS header.

C

    Input parameters :

C

|           |         |                                                                       |
|-----------|---------|-----------------------------------------------------------------------|
| FILE      | CHAR*12 | Filename                                                              |
| BREC(36)  | CHAR*80 | Character array with header<br>in FITS format.                        |
| NREC      | INTG*4  | Record sequential no.                                                 |
| LIST_FLAG | INTG*4  | LIST FLAG : 1-ONLY LIST                                               |
| FLAG      | CHAR*1  | DISPLAY FLAG: 'S' SHORT DISPLAY<br>'F' FULL DISPLAY<br>'N' NO DISPLAY |

C

    Output parameters :

C

|      |        |                                                                |
|------|--------|----------------------------------------------------------------|
| IERR | INTG*4 | Return code as<br>> 0 Error code<br>= 0 No error<br>< 0 'END'. |
| PNTR | INTG*4 | Pointer to data set                                            |

C

    External references :

C

    MIDAS library  
    OPFITS - Operation on FITS descriptor.

C

C

C

C

```
IMPLICIT NONE
INTEGER*4 NLABEL, NA
PARAMETER (NLABEL=360, NA=16)
REAL*4 BSCALE, BZERO, FMIN, FMAX, TSCAL(128)
REAL*4 STEP(NA), CRVAL(NA), CRPIX(NA), TZERO(128)
INTEGER*4 NPIX(NA), AXIS, BITPIX, GCOUNT, IERR
INTEGER*4 IEXT, IPP, IREC, LEN, LIST_FLAG, NAXIS
INTEGER*4 NBL, NCOM, NHIS, NREC, PCOUNT, PNTR
INTEGER*4 POINTER, SIZE, STATUS, TFIELDS
LOGICAL IDIS, IESO,IFORM
CHARACTER*80 BREC(NLABEL)
CHARACTER MSG01*33, OUTPUT*80, FLAG*1
CHARACTER*(*) FILE
```

C

    INCLUDE 'dra0:[midas.88jan.inc1]KEYFITS.INC/NOLIST'

C

    DATA

```
- MSG01 /' ^ Illegal keyword. '/
COMMON/FITS01/ SIZE, BITPIX, BSCALE, BZERO, FMIN, FMAX,
. TSCAL, TZERO
COMMON/FITS02/ NAXIS, NPIX,STEP,CRVAL,CRPIX,NCOM,NHIS,NBL,
. IEXT,PCOUNT,GCOUNT,TFIELDS,IESO,IFORM
IDIS = FLAG .EQ. 'F'
```

C

C ... Loop on fixed keywords : SIMPLE (or XTENSION), BITPIX, NAXIS  
C

```
    POINTER = 0
    IF(NREC.EQ.1) THEN
        IF(IERR.NE.0) GO TO 500
        PCOUNT = 0
        GCOUNT = 1
        POINTER = POINTER + 1
        IF (IDIS) THEN
            OUTPUT = BREC(POINTER)
            CALL PUTTEXT_ST(OUTPUT,STATUS)
        ENDIF
        IPP = 1
        IF(BREC(POINTER)(1:8) .EQ. 'SIMPLE ' .OR.
           BREC(POINTER)(1:8) .EQ. 'XTENSION' ) THEN
            IF(BREC(POINTER)(1:8) .EQ. 'XTENSION' ) IPP = 49
            CALL OPFITS(FILE, BREC(POINTER), FITSKEY(IPP)(14:15),
                       FITSKEY(IPP)(11:11), IERR, PNTR,LIST_FLAG,FLAG)
        ELSE
            IERR = 1
            IF (.NOT.IDIS) THEN
                OUTPUT = BREC(POINTER)
                CALL PUTTEXT_ST(OUTPUT,STATUS)
            ENDIF
            CALL PUTTEXT_ST(MSG01,STATUS)
        ENDIF
        DO IREC = 2,3
            IF(IERR.NE.0) GO TO 500
            POINTER = POINTER + 1
            IF (IDIS) THEN
                OUTPUT = BREC(POINTER)
                CALL PUTTEXT_ST(OUTPUT,STATUS)
            ENDIF
            IF(BREC(POINTER)(1:8) .NE. FITSKEY(POINTER)(1:8)) THEN
                IERR = 1
                IF (.NOT.IDIS) THEN
                    OUTPUT = BREC(POINTER)
                    CALL PUTTEXT_ST(OUTPUT,STATUS)
                ENDIF
                CALL PUTTEXT_ST(MSG01,STATUS)
            ELSE
                CALL OPFITS(FILE, BREC(POINTER), FITSKEY(POINTER)(14:15),
                           FITSKEY(POINTER)(11:11), IERR, PNTR,LIST_FLAG,FLAG)
            ENDIF
        ENDDO
    ENDIF
```

C

```
C ... Loop on keywords NAXISn
C
  DO IREC = 1,NAXIS
  IF(IERR.NE.0) GO TO 500
  POINTER = POINTER + 1
  IF (IDIS) THEN
    OUTPUT = BREC(POINTER)
    CALL PUTTEXT_ST(OUTPUT,STATUS)
  ENDIF
  LEN = INDEX(BREC(POINTER), ' ') - 6
  IF(LEN.LT.0 .OR. LEN.GE.3) THEN
    IERR = 1
    IF (.NOT.IDIS) THEN
      OUTPUT = BREC(POINTER)
      CALL PUTTEXT_ST(OUTPUT,STATUS)
    ENDIF
    CALL PUTTEXT_ST(MSG01,STATUS)
    GO TO 500
  ENDIF
  READ(BREC(POINTER), 10010) AXIS
  IF(BREC(POINTER)(1:5) .NE. FITSKEY(4)(1:5) .OR.
  .   AXIS .NE. IREC) THEN
    IERR = 1
    IF (.NOT.IDIS) THEN
      OUTPUT = BREC(POINTER)
      CALL PUTTEXT_ST(OUTPUT,STATUS)
    ENDIF
    CALL PUTTEXT_ST(MSG01,STATUS)
  ELSE
    CALL OPFITS(FILE, BREC(POINTER), FITSKEY(4)(14:15),
  .   FITSKEY(4)(11:11), IERR, PNTR,LIST_FLAG,FLAG)
  ENDIF
  ENDDO
ENDIF
C
C ... Loop on remeining keywords
C
  DO WHILE (POINTER.LT.NLABEL .AND. IERR.EQ.0)
  POINTER = POINTER + 1
  IF (IDIS) THEN
    OUTPUT = BREC(POINTER)
    CALL PUTTEXT_ST(OUTPUT,STATUS)
  ENDIF
  DO IREC = NKEYS1, NKEYS2
    IF(BREC(POINTER)(1:5) .EQ. FITSKEY(IREC)(1:5)) GO TO 300
  ENDDO
  DO IREC = NKEYS3, NKEYS4
    IF(BREC(POINTER)(1:8) .EQ. FITSKEY(IREC)(1:8)) GO TO 400
  ENDDO
C
C ... modified to allow NAXISn with n > NAXIS
C
C   IF(BREC(POINTER)(1:5) .EQ. 'NAXIS') GO TO 450
C
```



C Error - OTHER DESCRIPTORS

```
C
200     CONTINUE
        CALL OPFITS(FILE, BREC(POINTER), '28','0', IERR, PNTR,
        .      LIST_FLAG, FLAG)
        GO TO 450
300     CONTINUE
        LEN = INDEX(BREC(POINTER), ' ') - 6
        LEN = MIN0(Len,3)
        IF(Len.LT.0) GO TO 200
        READ(BREC(POINTER), 10010) AXIS
        IF(AXIS.LT.0) GO TO 200
400     CONTINUE
        CALL OPFITS(FILE, BREC(POINTER), FITSKEY(IREC)(14:15),
        .      FITSKEY(IREC)(11:11), IERR, PNTR,LIST_FLAG,FLAG)
450     CONTINUE
        ENDDO
C
C ... list header after error
C
500     CONTINUE
        IDIS = .TRUE.
        DO WHILE (POINTER.LT.NLABEL .AND. IERR.NE.-1 .AND. IDIS)
            POINTER = POINTER + 1
            OUTPUT = BREC(POINTER)
            CALL PUTTEXT_ST(OUTPUT,STATUS)
        ENDDO
        RETURN
C10000  FORMAT(A,':',I2,'/',I2)
10010   FORMAT(5X, I<LEN>)
        END
C
```

C\*\*\*\*\*

C

SUBROUTINE OPFITS(FILE, RECORD, OPCODE, CFMT, IERR, PNTR,  
LIST\_FLAG,FLAG)

C

Operation on FITS descriptors

C

Input params :

C

|           |         |                                                                       |
|-----------|---------|-----------------------------------------------------------------------|
| FILE      | CHAR*12 | filename                                                              |
| RECORD    | CHAR*80 | FITS header line                                                      |
| OPCODE    | CHAR*2  | Operation code                                                        |
| CFMT      | CHAR*1  | Value format                                                          |
| LIST_FLAG | INTG*4  | LIST FLAG : 1-ONLY LIST                                               |
| FLAG      | CHAR*1  | DISPLAY FLAG: 'S' SHORT DISPLAY<br>'F' FULL DISPLAY<br>'N' NO DISPLAY |

C

Output params :

C

|      |        |                                                            |
|------|--------|------------------------------------------------------------|
| IERR | INTG*4 | Return code as<br>>0 Error code<br>=0 No error<br><0 'END' |
| PNTR | INTG*4 | Pointer to data set                                        |

C

External references :

C

MIDAS library.

C

C

C

---

|               |                                                                                                          |
|---------------|----------------------------------------------------------------------------------------------------------|
| IMPLICIT      | NONE                                                                                                     |
| INTEGER*4     | NA, NKEY, I1, I2                                                                                         |
| PARAMETER     | (NA = 16, NKEY = 17, I1 = 10, I2 = 30)                                                                   |
| REAL*4        | BSCALE, BZERO, STEP(NA), CRVAL(NA),<br>CRPIX(NA),START(NA), RVALUE, FMIN, FMAX,<br>TSCAL(128),TZERO(128) |
| INTEGER*4     | NPIX(NA)                                                                                                 |
| CHARACTER*(*) | FILE                                                                                                     |
| LOGICAL       | LVALUE,IESO,IFORM                                                                                        |
| INTEGER*4     | BITPIX, CODE, GCOUNT, I, ICOL, IERR                                                                      |
| INTEGER*4     | IEXT, IFROM, II, II1, INDX, INN                                                                          |
| INTEGER*4     | ITO, IVALUE, L, L1, LEN, IVAL                                                                            |
| INTEGER*4     | LIST_FLAG, NAXIS, NBL, NBYTES, NCOM                                                                      |
| INTEGER*4     | NHIS, NST, NTT, NWCOL, PCOUNT, PNTR                                                                      |
| INTEGER*4     | SIZE, STATUS, TFIELDS, NVAL, NPR                                                                         |
| CHARACTER*8   | TNULL(128)                                                                                               |
| CHARACTER     | RECORD*80, BLANK*21, ESOST*21, ESOEND*21                                                                 |
| CHARACTER     | INSTRUM*72, ZVALUE*72, OUTPUT*80                                                                         |
| CHARACTER     | KEYUNIT*12, FILE1*12, CUNIT*192                                                                          |
| CHARACTER     | CVALUE*80, NEWLAB*20, WORK*16, OPCODE*2                                                                  |
| CHARACTER     | CFMT*1, IFMT*1, T*1, FLAG*1                                                                              |
| CHARACTER     | NAME*10, TYPE*8, FORM*20,MIDENT*30                                                                       |
| CHARACTER*30  | MSG02/' SIMPLE must be .TRUE.'/                                                                          |
| CHARACTER*30  | MSG03/' BITPIX not allowed '/                                                                            |
| CHARACTER*30  | MSG04/' NAXIS overflow '/                                                                                |
| CHARACTER*30  | MSG05/' NAXISn > NAXIS '/                                                                                |
| CHARACTER*30  | MSG06/' CTYPE not allowed '/                                                                             |
| CHARACTER*30  | MSG07/' No binary image '/                                                                               |
| CHARACTER*30  | MSG08/' XTENSION is not TABLE'/                                                                          |
| CHARACTER     | HRECORD*80                                                                                               |

```
COMMON/FITS01/      SIZE, BITPIX, BSCALE, BZERO, FMIN, FMAX,
                    TSCAL, TZERO
COMMON/FITS02/      NAXIS, NPIX,STEP,CRVAL,CRPIX,NCOM,NHIS,NBL,
                    IEXT,PCOUNT,GCOUNT,TFIELDS,IESO,IFORM
COMMON/FITS03/      FILE1,MIDENT,CUNIT,TNULL
EQUIVALENCE         (STEP, START)
INCLUDE             'DRA0:[MIDAS.88JAN.INCL]TABLEAPP.INC/NOLIST'
DATA                BLANK/'          '/
DATA                ESOST/'ESO-DESCRIPTORS START'/
DATA                ESOEND/'ESO-DESCRIPTORS END  '/

C
C ... Decode operation
C
      READ(OPCODE,10000) CODE
      CODE = CODE - 9
      IFMT = CFMT

C
C ... Decode axis number
C
      IF( CODE.EQ.4.OR.(CODE.GE.11.AND.CODE.LE.15).OR.
          (CODE.GE.26.AND.CODE.LE.32) ) THEN
        LEN = INDEX(RECORD, ' ') - 6
        LEN = MIN0(Len,3)
        READ(RECORD,10010,ERR=10) INDX
        GOTO 20
10      INDX = 1
        CALL PUTTEXT_ST('*** WARNING: error in FITS keyword ***',
                        STATUS)
        CALL PUTTEXT_ST('***          assigned to first axis ***',
                        STATUS)
20      CONTINUE
        IF(INDX.LE.0) THEN
          IERR = 5
          CALL PUTTEXT_ST(MSG05,STATUS)
          RETURN
        ENDIF
      ENDIF

C
C ... Decode value (IVALUE, RVALUE, CVALUE)
C
      L = 10
      DO WHILE (RECORD(L:L).EQ.' ' .AND. L.LT.80)
        L = L+1
      ENDDO
      LEN = 30 - L + 1

C
C ... Modified to allow blanks in data field.
C
      IF(IFMT.NE.'0' .AND. LEN.LE.0) THEN
        IFMT = '0'
        CODE = 16
      ENDIF

C
      IF(IFMT.EQ.'0') ZVALUE = RECORD(9:80)
      IF(IFMT.EQ.'1') READ(RECORD(L:L),10100) LVALUE
      IF(IFMT.EQ.'2') READ(RECORD(L:30),10200) IVALUE
      IF(IFMT.EQ.'3') READ(RECORD(L:30),*,ERR=1) RVALUE
C [3.5]
      GOTO 2
```

```
1      RVALUE = 0
      PRINT*,RECORD
      CALL PUTTEXT_ST('*** WARNING: data conversion error ***',
                     STATUS)
2      CONTINUE
C [3.5]
      IF(IFMT.EQ.'4') THEN
          L = INDEX(RECORD,'''')
          CVALUE = ' '
          IF (L.EQ.0) THEN
              PRINT*,RECORD
              CALL PUTTEXT_ST('*** WARNING: missing '' ',STATUS)
              GOTO 4
          ENDIF
          L1 = INDEX(RECORD(L+1:80),'''')-1
          IF (L1.EQ.0) THEN
              GOTO 4
          ENDIF
          CVALUE(1:L1) = RECORD(L+1:L+L1)
4      CONTINUE
          L = L1
      ENDIF
C
C ... if LIST return
C
      IF (LIST_FLAG.EQ.1) THEN
          IF (CODE.EQ.17) THEN
              IERR=-1
          ELSE
              IERR= 0
          ENDIF
          RETURN
      ENDIF
C
C ... determine action
C
      GO TO (100,110,120,130,140,150,160,170,180,190,
            .   200,210,220,230,240,250,260,270,280,290,
            .   300,310,320,330,340,350,360,370,380,390,
            .   400,410,420), CODE
C ** SIMPLE
100     CONTINUE
          IF(.NOT.LVALUE) THEN
              IERR = 2
              CALL PUTTEXT_ST(MSG02,STATUS)
          ENDIF
          IEXT = 0
          FILE1 = FILE
          RETURN
C ** Decode no. of bits per pixel
110     CONTINUE
          BITPIX = IVALUE
          IF(BITPIX.NE.8 .AND. BITPIX.NE.16 .AND. BITPIX.NE.32) THEN
              IERR = 3
              CALL PUTTEXT_ST(MSG03, STATUS)
          ENDIF
          RETURN
```

```
C ** Decode number of axis
120  CONTINUE
      NAXIS = IVALUE
C     IF(NAXIS.LE.0 .OR. NAXIS.GT.NA) THEN
      IF(NAXIS.LT.0 .OR. NAXIS.GT.16) THEN
          IERR = 4
          CALL PUTTEXT_ST(MSG04,STATUS)
      ENDIF
      RETURN
C ** Find no. of pixels
130  CONTINUE
      NPIX(INDX) = IVALUE
      IF(INDX.EQ.NAXIS) THEN
          SIZE = 1
          DO INDX =1, NAXIS
              SIZE = SIZE * NPIX(INDX)
          ENDDO
          IF(SIZE.LT.0) THEN
              IERR = 4
              CALL PUTTEXT_ST(MSG07,STATUS)
              RETURN
          ENDIF
          IF (SIZE.GT.0.AND.IEXT.EQ.0) GO TO 1000
      ENDIF
      RETURN
C ** Assign BSCALE
140  CONTINUE
C
C ... Just check crazy scales
C
      IF (RVALUE.GT.0.1E30) THEN
          CALL PUTTEXT_ST('*****',STATUS)
          CALL PUTTEXT_ST('      BSCALE OVERFLOW      ',STATUS)
          CALL PUTTEXT_ST('      PIXEL VALUES WILL      ',STATUS)
          CALL PUTTEXT_ST('      BE RE-SCALED          ',STATUS)
          CALL PUTTEXT_ST('*****',STATUS)
          RVALUE = 0.1E15
      ENDIF
      BSCALE = RVALUE
      RETURN
C ** Assign BZERO
150  CONTINUE
      BZERO = RVALUE
      RETURN
C ** Include as descriptor
160  CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      LEN = MIN(INDEX(RECORD,' '),INDEX(RECORD,'='))-1
      IF (CFMT.EQ.'0')
          . CALL PUTDSCR_ST(FILE1,RECORD(1:LEN),'C',ZVALUE,-1,72,STATUS)
            IF (CFMT.EQ.'2')
          . CALL PUTDSCR_ST(FILE1,RECORD(1:LEN),'I*4',IVALUE,1,1,STATUS)
            IF (CFMT.EQ.'3')
          . CALL PUTDSCR_ST(FILE1,RECORD(1:LEN),'R*4',RVALUE,1,1,STATUS)
            IF (CFMT.EQ.'4')
          . CALL PUTDSCR_ST(FILE1,RECORD(1:LEN),'C',CVALUE,1,L,STATUS)
      RETURN
C ** Include in INSTRUME
170  CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      CALL PUTDSCR_ST(FILE1,'INSTRUME','C',CVALUE,1,L,STATUS)
      RETURN
```

```
C ** Modify to HISTORY + Include as descriptor
180   CONTINUE
      RECORD(1:8) = 'HISTORY '
      GO TO 160
C ** Include in IDENT
190   CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      INSTRUM = CVALUE
      MIDENT = CVALUE
C     IF (FLAG.EQ.'S') CALL PUTTEXT_ST(RECORD, STATUS)
      CALL PUTDSCR_ST(FILE1,'IDENT','C',INSTRUM,1,72,STATUS)
      RETURN
C ** CRVAL
200   CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      CRVAL(INDX) = RVALUE
      CALL PUTDSCR_ST(FILE1,'START','R*4',CRVAL(INDX),INDX,1,STATUS)
      RETURN
C ** CRPIX
210   CONTINUE
      CRPIX(INDX) = RVALUE
      RETURN
C ** CDELT
220   CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      STEP(INDX) = RVALUE
      CALL PUTDSCR_ST(FILE1,'STEP','R*4',STEP(INDX),INDX,1,STATUS)
      RETURN
C ** Assign CUNIT
230   CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      IFROM = (INDX-1)*16 + 1
      ITO = IFROM + 7
      CUNIT(IFROM:ITO) = CVALUE(1:8)
      RETURN
C ** CROTA
240   CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      CALL PUTDSCR_ST(FILE1,'CROTA','R*4',RVALUE,INDX,1,STATUS)
C ** Skip
250   RETURN
C ** End of process. Check format in the current implementation
260   CONTINUE
      IERR = -1
      CALL PUTTEXT_ST(' ',STATUS)
      IF (IEXT.EQ.1) THEN
          FMT_NCOL = TFIELDS
          NWCOL = 0
C
C ... ALLOCATE TABLE
C
      DO I = 1, FMT_NCOL
          CALL TBL_TYPCHK(FMT_TYPE(I),T,NBYTES)
          NWCOL = NWCOL + (ABS(NBYTES)+3)/4
      ENDDO
      CALL TBL_INIT(FILE,NWCOL,NPIX(2),STATUS)
      FMT_LLEN = NPIX(1)
C
```

```
C ... DEFINE COLUMNS
C
      DO I = 1, FMT_NCOL
        CALL TBL_DEFCOL(FILE,FMT_TYPE(I),FMT_FORMAT(I),
          FMT_UNIT(I),FMT_LABEL(I),ICOL,STATUS)
        CALL TBL_RDFMT (FILE,ICOL,FMT_FORMAT(I),II1,FMT_TYPE(I),
          STATUS)
        WORK          = FMT_FORMAT(I)
        II            = INDEX(WORK,' ') -1
        FMT_FORMAT(I) = '('//WORK(1:II)//')'
        FMT_LAST(I)   = FMT_FIRST(I) + II1 - 1
      ENDDO
    ELSE
      IF (SIZE.NE.0) THEN
        DO I = 1, NAXIS
          CRVAL(I) = CRVAL(I) - (CRPIX(I)-1.)*STEP(I)
        ENDDO
        CALL PUTDSCR_ST(FILE1,'START','R*4',CRVAL,1,NAXIS,STATUS)
        NTT = MIN(16*(NAXIS+1),192)
        CALL PUTDSCR_ST(FILE1,'CUNIT','C',CUNIT,1,NTT,STATUS)
      ENDIF
    ENDIF
  RETURN
C ** Modify to COMMENT + include as descriptor
270  CONTINUE
      RECORD(1:8) = 'COMMENT '
      GO TO 160
C ** Include full record
280  CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      CALL PUTDSCR_ST(FILE1,RECORD(1:8),'C',RECORD(9:80),1,72,STATUS)
      RETURN
C ** append record
290  CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      IF (RECORD(1:7).EQ.'COMMENT') THEN
        NST = NCOM
        NCOM = NCOM + 1
      ELSE
        IF (RECORD(1:7).EQ.'HISTORY') THEN
          NST = NHIS
          NHIS = NHIS + 1
        ELSE
          NST = NBL
          NBL = NBL + 1
        ENDIF
      ENDIF
    ENDIF
C Alt
      CALL PUTDSCR_ST(FILE1,RECORD(1:8),'C',
        RECORD(9:80),NST*72+1,72,STATUS)
C Alt
C Neu
C      HRECORD = RECORD(9:80)//'          '
C      CALL PUTDSCR_ST(FILE1,RECORD(1:8),'C',
C        HRECORD(1:80),NST*80+1,80,STATUS)
C Neu
      RETURN
```

```
C ** BUNIT
300   CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      IFROM = NAXIS*16 +1
      ITO   = IFROM + 7
      CUNIT(IFROM:ITO) = CVALUE(1:8)
      RETURN

C ** XTENSION
310   CONTINUE
      IF (CVALUE(1:5) .NE. 'TABLE') THEN
        IERR = 8
        CALL PUTTEXT_ST(MSG08,STATUS)
      ELSE
        IEXT = 1
        INN  = INDEX(FILE,' ')-1
        FILE1= FILE(1:INN)//'.TBL'
        FMT_NCOL = 0
        FMT_NROW = 0
        FMT_LLEN = 0
        DO I = 1, 128
          TSCAL(I)   = 1.0
          TZERO(I)   = 0.0
          TNULL(I)   = ' '
          FMT_FIRST(I) = 0
          FMT_LAST(I)  = 0
          FMT_FORMAT(I) = ' '
          FMT_LABEL(I) = ' '
          FMT_UNIT(I)  = ' '
          FMT_TYPE(I)  = ' '
        ENDDO
      ENDIF
      RETURN

C ** PCOUNT
320   CONTINUE
      PCOUNT = IVALUE
      RETURN

C ** GCOUNT
330   CONTINUE
      GCOUNT = IVALUE
      RETURN

C ** TFIELDS
340   CONTINUE
      TFIELDS = IVALUE
      RETURN

C ** TTYPE
350   CONTINUE
      CALL LABTR(CVALUE,NEVLAB)
      FMT_LABEL(INDX) = NEVLAB
      RETURN

C ** TBCOL
360   CONTINUE
      FMT_FIRST(INDX) = IVALUE
      RETURN

C ** TFORM
370   CONTINUE
      FMT_FORMAT(INDX) = CVALUE
      T                 = CVALUE(1:1)
      IF (T.EQ.'A')      FMT_TYPE(INDX) = 'C* '//CVALUE(2:4)
      IF (T.EQ.'D')      FMT_TYPE(INDX) = 'R*8'
      IF (T.NE.'A'.AND.T.NE.'D') FMT_TYPE(INDX) = 'R*4'
      RETURN
```



```
C ** TUNIT
380   CONTINUE
      FMT_UNIT(INDX) = CVALUE
      RETURN

C ** TSCAL
390   CONTINUE
      TSCAL(INDX) = RVALUE
      RETURN

C ** TZERO
400   CONTINUE
      TZERO(INDX) = RVALUE
      RETURN

C ** TNULL
410   CONTINUE
      TNULL(INDX) = CVALUE
      RETURN

C ** HISTORY + ESO DESCRIPTORS
420   CONTINUE
      IF (SIZE.EQ.0 .OR. IEXT.NE.0) RETURN
      IF (RECORD(I1:I2).EQ.ESOST) THEN !CHECK IF START ESO DESCRIPTORS
        IESO = .TRUE.
        IFORM = .TRUE.
        RETURN
      ENDIF
      IF (IESO) THEN ! ESO DESCRIPTORS
        IF (IFORM) THEN
          IF (RECORD(I1:I2).EQ.BLANK) RETURN ! SKIP BLANKS
          IF (RECORD(I1:I2).EQ.ESOEND) THEN
            IESO = .FALSE.
            RETURN
          ENDIF
          CALL ESODCF(FILE1,RECORD,NAME,TYPE,NVAL,NPR,FORM) !DEC.FORMAT
          IFORM = .FALSE.
          IVAL = 1
        ELSE
          CALL ESODCW(FILE1,RECORD,NAME,TYPE,IVAL,
                     NVAL,NPR,FORM) ! WRITE DESCRIPTOR
        ENDIF
        IF (IVAL.GT.NVAL) IFORM = .TRUE.
      ELSE ! INCLUDE HISTORY
        IF (RECORD(I1:I2).EQ.BLANK) RETURN ! SKIP BLANKS
      ENDIF
      CALL PUTDSCR_ST(FILE1,RECORD(1:8),'C',
                    RECORD(9:80),1,72, STATUS)
C Alt
C
C
C Alt
C Neu
      L = INDEX(RECORD,'''')
      CVALUE = '
      CVALUE(51:80) = '
      IF (L.EQ.0) THEN
        PRINT*,RECORD
        CALL PUTTEXT_ST('*** WARNING: missing .',
                      STATUS)
        GOTO 424
      ENDIF
      L1 = INDEX(RECORD(L+1:80),'''')-1
      IF (L1.EQ.0) THEN
CV      PRINT*,RECORD
CV      CALL PUTTEXT_ST('*** WARNING: MISSING ..',
CV      STATUS)
        GOTO 424
      ENDIF
```

```

CVALUE(1:L1) = RECORD(L+1:L+L1)
424 CONTINUE
L = L1
C HRECORD = RECORD(9:80)//' '
CALL PUTDSCR_ST(FILE1,RECORD(1:8),'C',
                CVALUE(1:80),1,80,STATUS)
C Neu
    ENDIF
    RETURN
C
C Create + define standard defaults
C
1000 CONTINUE
CALL PUTIMAG_ST(FILE,'O',NAXIS,NPIX,START,STEP,INSTRUM,CUNIT,
                PNTR, STATUS)
ZVALUE = ' '
CALL PUTDSCR_ST(FILE,'COMMENT','C',ZVALUE,1,72,STATUS)
RETURN
C
10000 FORMAT(I2)
10010 FORMAT(5X,I<LEN>)
10020 FORMAT(I1)
10100 FORMAT(L1)
10200 FORMAT(I<LEN>)
10300 FORMAT(F<LEN>.0)
END
C
C*****
C
SUBROUTINE FNDBL(BREC,ITABLE)
C
    IMPLICIT NONE
    INTEGER*4 NLABEL, NA
    PARAMETER (NLABEL=36, NA=16)
    INTEGER*4 IREC
    CHARACTER*80 BREC(NLABEL)
    LOGICAL ITABLE
C
    ITABLE = .FALSE.
    DO IREC = 1, NA
        IF(BREC(IREC)(1:6).EQ.'EXTEND'.AND.
            BREC(IREC)(30:30).EQ.'T') THEN
            ITABLE = .TRUE.
        RETURN
    ENDIF
ENDDO
RETURN
END
C
```

```
C*****
C
C      SUBROUTINE LABTR(OLDLAB,NEWLAB)
C
C      CHANGE SPECIAL CHARACTERS TO '_'
C
C      IMPLICIT      NONE
C      INTEGER*4     I, I1, II, ISTAT
C      CHARACTER*20  OLDLAB,NEWLAB,WORK
C      CHARACTER*1   C
C      CHARACTER*80  OUTPUT
C      INTEGER       STR$UPCASE
C
C      I1      = INDEX(OLDLAB,' ') -1
C      WORK    = ' '
C      DO I = 1, I1
C         C = OLDLAB(I:I)
C         IF (INDEX('!@#$$%^&*()-+=<>/[{}]:;,.',C).NE.0) THEN
C            WORK(I:I) = '_'
C         ELSE
C            WORK(I:I) = C
C         ENDIF
C      ENDDO
C      II = STR$UPCASE(NEWLAB,WORK)
C      IF (OLDLAB.NE.NEWLAB) THEN
C         OUTPUT = ' *** '//OLDLAB//' modified to '//NEWLAB//' ***'
C         CALL PUTTEXT_ST(OUTPUT,ISTAT)
C      ENDIF
C      RETURN
C      END
C
```

```
C*****
C
C      SUBROUTINE ESODCF(FRAME1,RECORD,NAME,TYPE,NVAL,NPR,FORM)
C
C      DECODE ESO DESCRIPTOR FORMAT IN HISTORY RECORD
C
C      INPUT ARGUMENTS
C      FRAME1  CHAR(*)      FILE NAME
C      RECORD  CHAR*80     HISTORY RECORD
C
C      OUTPUT ARGUMENTS
C      NAME    CHAR(*)      DESCRIPTOR NAME
C      TYPE    CHAR(*)      DESCRIPTOR TYPE
C      NVAL    INTEGER*4    NUMBER OF VALUES
C      NPR     INTEGER*4    NUMBER OF VALUES PER RECORD
C      FORM    CHAR(*)      DECODING FORMAT
C
C      IMPLICIT NONE
C      CHARACTER*(*)      FRAME1
C      CHARACTER*(*)      RECORD, NAME, TYPE, FORM
C      INTEGER*4          NVAL, NPR, I1, INDEX, IVAL(1000)
C      INTEGER*4          STATUS
C      REAL*8             DVAL(500)
C      REAL*4             RVAL(1000)
C      CHARACTER*80       WORK
C      CHARACTER*1000     CVAL/' '/
C      EQUIVALENCE        (DVAL, RVAL)
C      EQUIVALENCE        (IVAL, RVAL)
C
C      ... decode information
C
C      WORK = RECORD(11:26)
C      I1   = INDEX(WORK, ' ') - 1
C      NAME = WORK(1:I1)
C      WORK = RECORD(29:36)
C      I1   = INDEX(WORK, ' ') - 1
C      TYPE = WORK(1:I1)
C      WORK = RECORD(44:48)
C      READ(WORK,100) NVAL
C      WORK = RECORD(51:80)
C      I1   = INDEX(WORK, ' ') - 1
C      FORM = '('//WORK(1:I1)//')'
C
C      ... write descriptor
C
C      IF (TYPE(1:1).EQ.'I') THEN
C        CALL PUTDSCR_ST(FRAME1, NAME, TYPE, IVAL, 1, NVAL, STATUS)
C        NPR = 7
C      ELSEIF (TYPE(1:1).EQ.'R') THEN
C        CALL PUTDSCR_ST(FRAME1, NAME, TYPE, RVAL, 1, NVAL, STATUS)
C        NPR = 5
C      ELSEIF (TYPE(1:1).EQ.'C') THEN
C        CALL PUTDSCR_ST(FRAME1, NAME, TYPE, CVAL, 1, NVAL, STATUS)
C        NPR = 70
C      ENDIF
C      RETURN
100  FORMAT(I5)
C      END
C
```

```
C*****
C
C      SUBROUTINE ESODCW(FRAME1,RECORD,NAME,TYPE,IP,
C          NVAL,NPR,FORM)
C
C      DECODE ESO DESCRIPTOR VALUES IN HISTORY RECORD
C
C      INPUT ARGUMENTS
C      FRAME1  CHAR*(*)      FILE NAME
C      RECORD  CHAR*80      HISTORY RECORD
C
C      OUTPUT ARGUMENTS
C      NAME    CHAR*(*)      DESCRIPTOR NAME
C      TYPE    CHAR*(*)      DESCRIPTOR TYPE
C      IP      INTEGER*4     POINTER TO ACTUAL VALUE
C      NVAL    INTEGER*4     NUMBER OF VALUES
C      NPR     INTEGER*4     NUMBER OF VALUES PER RECORD
C      FORM    CHAR*(*)      DECODING FORMAT
C
C      IMPLICIT NONE
C      CHARACTER*(*)      FRAME1
C      CHARACTER*(*)      RECORD, NAME, TYPE, FORM
C      INTEGER*4          NVAL, NPR, IP, MIN, N1, II
C      INTEGER*4          IVAL(7), STATUS
C      REAL*8            DVAL(5)
C      REAL*4            RVAL(5)
C      CHARACTER*80      CVAL
C
C      ... decode information + write descriptor
C
C      N1 = MIN(NPR, NVAL - IP + 1)
C      IF (TYPE(1:1).EQ.'I') THEN
C          READ(RECORD(10:79),FORM) IVAL
C          CALL PUTDSCR_ST(FRAME1, NAME, TYPE, IVAL, IP, N1, STATUS)
C      ELSEIF (TYPE(1:3).EQ.'R*4') THEN
C
C      ... special handling due to overflow !!!!
C
C      II = INDEX(RECORD,'E+39')
C      IF (II.GT.0) THEN
C          RECORD(II:II+3) = 'E+38'
C          CALL PUTTEXT_ST(' Overflow in CUTS',STATUS)
C      ENDIF
C      READ(RECORD(10:79),FORM) RVAL
C      CALL PUTDSCR_ST(FRAME1, NAME, 'R*4', RVAL, IP, N1, STATUS)
C      ELSEIF (TYPE(1:3).EQ.'R*8') THEN
C          READ(RECORD(10:79),FORM) DVAL
C          CALL PUTDSCR_ST(FRAME1, NAME, 'R*8', DVAL, IP, N1, STATUS)
C      ELSEIF (TYPE(1:1).EQ.'C') THEN
C          CVAL = RECORD(10:80)
C          CALL PUTDSCR_ST(FRAME1, NAME, TYPE, CVAL, IP, N1, STATUS)
C      ENDIF
C      IP = IP + NPR
C      RETURN
C      END
C
```

```
C*****
C
SUBROUTINE PDS_HAD(HEADER,DATALUN,JDIM)
C
INTEGER*2 ISTART,L1,L2,L3
INTEGER*4 JDIM(2),ISTATUS,IZEIL
INTEGER*4 IP,POSCHAR
INTEGER*4 K,ICOUNT
!
INTEGER*4 DIMBUF,DIMBYTBUF
PARAMETER (DIMBYTBUF=2*16383)
PARAMETER (DIMBUF=16383)

INTEGER*2 WORDBUF(DIMBUF) ! buffer for data
BYTE      BYTEBUF(DIMBYTBUF)
BYTE      B
REAL*8    XWERT
CHARACTER*128 L
CHARACTER*1 C
CHARACTER*9  DATUM,COMM4
CHARACTER*80 BSCALE,BZERO,BUNIT,OBJECT,CRVAL1,CRPIX1,CRDEL1
CHARACTER*20 CRVALX,CRVALY,CRDELTX,CRDELTY
CHARACTER*80 CTYPE1,CRVAL2,CRPIX2,CRDEL2,CTYPE2,DATES
CHARACTER*80 ORIGIN,INSTRUME,COMMENT,HISTORY
CHARACTER*80 TITLE,BLANK,INVAL,COMMENT1,COMMENT2,COMMENT3
CHARACTER*80 COMMENT4,COMMENT5,COMM2,COMM3,USER
C
INTEGER*4 NCOLS,NROWS,MSTART,I,J,NDMAS,LBLOCK,ISTAT,NRET
INTEGER*4 IDIFKO

INTEGER*4 STATUS,READ_STATUS,IO_STAT,IOSTAT_STATUS
INTEGER*4 LIB$GET_LUN,NODLUN,MAPLUN,DATALUN,BAB_NCHAR
INTEGER*4 LIMPOINTS/14000/,NPOINTS,ACTVAL
C
CHARACTER*128 PDSFILE/'PDSDATA.DAT'/
CHARACTER*60  INFILE,OUTFILE
C
COMMON/FITS99/ INFILE,OUTFILE
LOGICAL*4 ERROR
C
INTEGER*4 KOORSX,KOORSY,KLAENG,KPIXEL,KPIXNR,KOORNX,KOORNY
INTEGER*4 KOORDX,KOORDY,KOORIX,KOORIY

INTEGER*2 VELOCX,VELOCITY,DIFKOX,DIFKOY,TAKTF1,TAKTF2
INTEGER*2 POINTN,LISTNR,REPZAL,NOPFEH,OPTVEL,GRAZAL
INTEGER*2 GAIN,ADJUST,DENMOD,SCNZAL,TYPZAL,REPDMA,DMANUM,DIRTRN
INTEGER*2 LISNUM,PRGNUM,SYNPRG,SYNTKT
INTEGER*2 INCTIM,MEAFLG,ISONUM,TABNUM
INTEGER*2 MAXAKT,MINAKT,INTENS,ILIGHT,VFOTOM,STATUS_PDS
INTEGER*2 MXWERT,MAXPOS,MIWERT,MINPOS,KRWERT,KORPOS
C

EQUIVALENCE(BYTEBUF,WORDBUF)
```

|                                       |         |
|---------------------------------------|---------|
| EQUIVALENCE(WORDBUF(1985),            | KOORSX) |
| EQUIVALENCE(WORDBUF(1987),            | KOORSY) |
| EQUIVALENCE(WORDBUF(1989),VELOCX)     |         |
| EQUIVALENCE(WORDBUF(1990),VELOCY)     |         |
| EQUIVALENCE(WORDBUF(1991),DIFKOX)     |         |
| EQUIVALENCE(WORDBUF(1992),DIFKOY)     |         |
| EQUIVALENCE(WORDBUF(1993),TAKTF1)     |         |
| EQUIVALENCE(WORDBUF(1994),TAKTF2)     |         |
| EQUIVALENCE(WORDBUF(1995),POINTN)     |         |
| EQUIVALENCE(WORDBUF(1996),LISTNR)     |         |
| EQUIVALENCE(WORDBUF(1997),REPZAL)     |         |
| EQUIVALENCE(WORDBUF(1998),NOPFEH)     |         |
| EQUIVALENCE(WORDBUF(1999),OPTVEL)     |         |
| EQUIVALENCE(WORDBUF(2000),GRAZAL)     |         |
| EQUIVALENCE(WORDBUF(2001),GAIN)       |         |
| EQUIVALENCE(WORDBUF(2002),ADJUST)     |         |
| EQUIVALENCE(WORDBUF(2003),DENMOD)     |         |
| EQUIVALENCE(WORDBUF(2004),SCNZAL)     |         |
| EQUIVALENCE(WORDBUF(2005),TYPZAL)     |         |
| EQUIVALENCE(WORDBUF(2006),REPDMA)     |         |
| EQUIVALENCE(WORDBUF(2007),DMANUM)     |         |
| EQUIVALENCE(WORDBUF(2008),DIRTRN)     |         |
| EQUIVALENCE(WORDBUF(2009),LISNUM)     |         |
| EQUIVALENCE(WORDBUF(2010),PRGNUM)     |         |
| EQUIVALENCE(WORDBUF(2011),            | KLAENG) |
| EQUIVALENCE(WORDBUF(2013),            | KPIXEL) |
| EQUIVALENCE(WORDBUF(2015),            | KPIXNR) |
| EQUIVALENCE(WORDBUF(2017),            | KOORNX) |
| EQUIVALENCE(WORDBUF(2019),            | KOORNY) |
| EQUIVALENCE(WORDBUF(2021),            | KOORDX) |
| EQUIVALENCE(WORDBUF(2023),            | KOORDY) |
| EQUIVALENCE(WORDBUF(2025),SYNPRG)     |         |
| EQUIVALENCE(WORDBUF(2026),SYNTKT)     |         |
| EQUIVALENCE(WORDBUF(2027),INCTIM)     |         |
| EQUIVALENCE(WORDBUF(2028),MEAFLG)     |         |
| EQUIVALENCE(WORDBUF(2029),ISONUM)     |         |
| EQUIVALENCE(WORDBUF(2030),TABNUM)     |         |
| EQUIVALENCE(WORDBUF(2031),MAXAKT)     |         |
| EQUIVALENCE(WORDBUF(2032),MINAKT)     |         |
| EQUIVALENCE(WORDBUF(2033),            | KOORIX) |
| EQUIVALENCE(WORDBUF(2035),            | KOORIY) |
| EQUIVALENCE(WORDBUF(2037),INTENS)     |         |
| EQUIVALENCE(WORDBUF(2038),ILIGHT)     |         |
| EQUIVALENCE(WORDBUF(2039),VFOTOM)     |         |
| EQUIVALENCE(WORDBUF(2040),STATUS_PDS) |         |
| EQUIVALENCE(WORDBUF(2041),MXWERT)     |         |
| EQUIVALENCE(WORDBUF(2042),MAXPOS)     |         |
| EQUIVALENCE(WORDBUF(2043),MIWERT)     |         |
| EQUIVALENCE(WORDBUF(2044),MINPOS)     |         |
| EQUIVALENCE(WORDBUF(2045),KRWERT)     |         |
| EQUIVALENCE(WORDBUF(2046),KORPOS)     |         |

C  
C  
C  
C  
C

read first DMA to get parameters for map:

```
READ(DATALUN,IOSTAT=IO_STAT)(WORDBUF(I),I=1,DIMBUF)
CALL BAB_CHECK(IOSTAT_STATUS(IO_STAT))
NDMAS=REPDMA
JDIM(1)=KPIXNR
JDIM(2)=SCNZAL
```

C

```
C WRITE HEADER
C
  L= ' '
  J=1
  K=0
C
  IZEIL=0
  DO I=1,3000
    C=CHAR(BYTEBUF(I))
    XWERT=ICHAR(C)
    IF(XWERT.LE.128.0) B=XWERT+0.5
C
    K=K+1
C
    IF(B.GE.32)THEN                ! printable chars
      IF(.NOT.(J.EQ.1.AND.B.EQ.32))THEN
        L(J:J)=C
        J=J+1
      ENDIF
    ENDIF
C
    IF(K.GE.64)THEN
      IF(BAB_NCHAR(L).GT.0) THEN
        CALL PUTTEXT_ST(L(1:(BAB_NCHAR(L))),ISTATUS)
        IZEIL = IZEIL + 1
        IF (IZEIL .EQ. 1) THEN
C----- POSCHAR NUR EINMAL GESETZT
          POSCHAR= INDEX(L,':')
          USER=' ' //L(POSCHAR+4:POSCHAR+28)
          #           // ' ' / USER IDENTIFIER '
        ELSE IF ( IZEIL .EQ. 2) THEN
          OBJECT=' ' //L(POSCHAR+4:POSCHAR+28)
          #           // ' ' / PLATE IDENTIFIER '
        ELSE IF ( IZEIL .EQ. 3) THEN
          COMMENT1=L(POSCHAR+1:(BAB_NCHAR(L)))
        ELSE IF ( IZEIL .EQ. 4) THEN
          IF (BAB_NCHAR(L).LE.POSCHAR+4) THEN
            COMMENT2=' #'
          ELSE
            COMMENT2=L(POSCHAR+4:(BAB_NCHAR(L)))
          ENDIF
        ELSE IF ( IZEIL .EQ. 5) THEN
          TITLE = ' ' //L(POSCHAR+4:POSCHAR+28)
          #           // ' ' / OBJECT NAME '
        ELSE IF ( IZEIL .EQ. 6) THEN
          COMMENT3=L(POSCHAR+3:(BAB_NCHAR(L)))
        ELSE IF ( IZEIL .EQ. 7) THEN
          COMMENT4=L(POSCHAR+3:(BAB_NCHAR(L)))
        ELSE IF ( IZEIL .EQ. 8) THEN
          COMMENT5=L(POSCHAR+3:(BAB_NCHAR(L)))
        ELSE IF ( IZEIL .EQ. 9) THEN
          IF (BAB_NCHAR(L).LE.POSCHAR+4) THEN
            COMMENT = ' #'
          ELSE
            COMMENT = ' ' //L(POSCHAR+4:(BAB_NCHAR(L)))// ' '
          ENDIF
        ELSE IF ( IZEIL .EQ.19) THEN
          COMM4=' '
          COMM4(1:1)=L(57:57)
        ELSE IF ( IZEIL .EQ.21) THEN
          COMM4(2:2)=L(28:28)
        
```



```

ELSE IF ( IZEIL .EQ.27) THEN
  COMM4(3:3)=L(28:28)
  END IF
ENDIF
L= ' '
J=1
K=0
  ENDIF
C
  ENDDO
C
  BSCALE='1. / TRUE = TAPE*BSCALE +BZERO'
  CALL FITS_HAD(HEADER,'BSCALE',BSCALE,JDIM,ISTAT)

  BZERO='0.'
  CALL FITS_HAD(HEADER,'BZERO',BZERO,JDIM,ISTAT)

  BUNIT='''AD UNITS'' / DEPENDS ON MEASUREMENT MODE'
  CALL FITS_HAD(HEADER,'BUNIT',BUNIT,JDIM,ISTAT)

  BLANK='-32768 / INVALID PIXEL FLAG'
  CALL FITS_HAD(HEADER,'BLANK',BLANK,JDIM,ISTAT)

  INVAL='-32768 / INVALID PIXEL FLAG USED BY ASPIC'
  CALL FITS_HAD(HEADER,'INVAL',INVAL,JDIM,ISTAT)

  KOORSX = KOORSX -KOORNX -(KPIXEL*KPIXNR/2)
  KOORSY = KOORSY -KOORNY
  WRITE(CRVALX,*) KOORSX
  WRITE(CRVALY,*) KOORSY

  CRVAL1= CRVALX
#    //' / COORDN FIRST PIXEL IN ROW (USER COORD.) '
  CALL FITS_HAD(HEADER,'CRVAL1',CRVAL1,JDIM,ISTAT)

  CRPIX1=' 1. / CRVAL1 REFERS TO FIRST PIXEL IN ROW '
  CALL FITS_HAD(HEADER,'CRPIX1',CRPIX1,JDIM,ISTAT)

  WRITE(CRDELTX,*) KPIXEL
  CRDELTX=CRDELTX //' / INCREMENT ALONG ROW (X-STEP)'
  CALL FITS_HAD(HEADER,'CDELTX',CRDELTX,JDIM,ISTAT)

  CTYPE1='''MICRONS '' / 10**-6 METERS '
  CALL FITS_HAD(HEADER,'CTYPE1',CTYPE1,JDIM,ISTAT)

  CRVAL2= CRVALY //' / CONST. COORD OF FIRST ROW (Y-START)'
  CALL FITS_HAD(HEADER,'CRVAL2',CRVAL2,JDIM,ISTAT)

  CRPIX2=' 1. / CRVAL2 REFERS TO FIRST PIXEL IN COLUMN '
  CALL FITS_HAD(HEADER,'CRPIX2',CRPIX2,JDIM,ISTAT)

  IDIFKO=DIFKOY
  WRITE(CRDELTY,*) IDIFKO
  CRDELTY= CRDELTY //' / INCREMENT ALONG COLUMN (Y-STEP) '
  CALL FITS_HAD(HEADER,'CDELTY',CRDELTY,JDIM,ISTAT)

  CTYPE2='''MICRONS '' / 10**-6 METERS '
  CALL FITS_HAD(HEADER,'CTYPE2',CTYPE2,JDIM,ISTAT)

  CALL FITS_HAD(HEADER,'USER',USER,JDIM,ISTAT)
  CALL FITS_HAD(HEADER,'OBJECT',OBJECT,JDIM,ISTAT)
  CALL FITS_HAD(HEADER,'TITLE',TITLE,JDIM,ISTAT)
```

```
INSTRUME='''PDS 1010 MICRODENSITOMETER'''
CALL FITS_HAD(HEADER,'INSTRUME',INSTRUME,JDIM,ISTAT)

ORIGIN='''ASTRON. INST. UNIV. BONN (AIUB)'''
CALL FITS_HAD(HEADER,'ORIGIN',ORIGIN,JDIM,ISTAT)

CALL DATE(DATUM)
DATES='''//DATUM//''' / FILE CREATION DATE DD-MMM-YY'
CALL FITS_HAD(HEADER,'DATE',DATES,JDIM,ISTAT)

IF (COMMENT2(1:1) .NE. '#') THEN
  L1=BAB_NCHAR(COMMENT2)
  L2=BAB_NCHAR(COMMENT1)
  COMMENT2='''//COMMENT2(1:L1)//''' / TELESCOPE'
  CALL FITS_HAD(HEADER,'TELESCOP',COMMENT2,JDIM,ISTAT)
ENDIF

IF (COMM4(3:3).EQ.'1') THEN
  COMM3='''Scan = Meander'
ELSE
  COMM3='''Scan = Einfach'
ENDIF
IF (COMM4(2:2).EQ.'1') THEN
  COMM2=COMM3(1:15)//' Licht = Auflicht'
ELSE
  COMM2=COMM3(1:15)//' Licht = Transmis'
ENDIF
IF (COMM4(1:1).EQ.'1') THEN
  COMM3=COMM2(1:33)//' AD-Wandler = Log'' / PDS-STEUERSTATUS'
ELSE
  COMM3=COMM2(1:33)//' AD-Wandler = Lin'' / PDS-STEUERSTATUS'
ENDIF
CALL FITS_HAD(HEADER,'PDSMODE',COMM3,JDIM,ISTAT)

L1=BAB_NCHAR(COMMENT3)
L2=BAB_NCHAR(COMMENT4)
L3=BAB_NCHAR(COMMENT5)
IF (L1+L2+L3.LE.46) THEN
  COMM3='''Filter='//COMMENT3(1:L1)//
#      ' Spalt ='//COMMENT4(1:L2)//
#      ' Vergr ='//COMMENT5(1:L3)//''''
  CALL FITS_HAD(HEADER,'PDSOPTIK',COMM3,JDIM,ISTAT)
ENDIF

IF (COMMENT(1:1) .NE. '#') THEN
  CALL FITS_HAD(HEADER,'COMMENT',COMMENT,JDIM,ISTAT)
ENDIF

CALL GETKEY_ST('IN_A','C',1,60,ACTVAL,INFILE,STATUS)
CALL GETKEY_ST('OUT_A','C',1,60,ACTVAL,OUTFILE,STATUS)
CALL DEFAULT_FILE_NAME(INFILE,'.DAT',INFILE)
CALL DEFAULT_FILE_NAME(OUTFILE,'.BDF',OUTFILE)
L1=BAB_NCHAR(INFILE)
L2=BAB_NCHAR(OUTFILE)
COMM3='''MIDPROC:PDS '//INFILE(1:L1)//' '//OUTFILE(1:L2)//''''
CALL FITS_HAD(HEADER,'HISTORY',COMM3,JDIM,ISTAT)
CALL FITS_HAD(HEADER,'END',COMMENT,JDIM,ISTAT)
```

C

```
RETURN
END
```

C

```
C-----
SUBROUTINE FILLARRAY (ARRAY, JDIM, WORDBUF, DIMBUF, REPZAHL, ISTART)
IMPLICIT NONE

INTEGER*4 JDIM(2)
INTEGER*4 DIMBUF
INTEGER*2 WORDBUF(DIMBUF), REPZAHL
INTEGER*4 ISTART, IEND, IX, IY
REAL*4 ARRAY(JDIM(1), JDIM(2))

IEND=ISTART+REPZAHL
DO 10 IY=ISTART, IEND
  DO 10 IX=1, JDIM(1)
    ARRAY (IX, IY)=WORDBUF(2048+(IY-ISTART)*JDIM(1)+IX)
10 CONTINUE
  ISTART=IY
  RETURN
END

C
C*****
C
SUBROUTINE FITS_HAD(HEADER, KEY, TEXT, JDIM, ISTAT)
C
CHARACTER*(*) HEADER(*), KEY, TEXT
CHARACTER*8 KWORT
CHARACTER*64 VALUE
CHARACTER*80 RECORD
C
INTEGER JDIM(2)
INTEGER IND/5/
C
IF(IND.EQ.5) THEN
WRITE(RECORD,*) 'SIMPLE = T'
HEADER(1)=RECORD(2:80)
WRITE(RECORD,*) 'BITPIX = 32'
HEADER(2)=RECORD(2:80)
WRITE(RECORD,*) 'NAXIS = 2'
HEADER(3)=RECORD(2:80)
WRITE(RECORD,*) 'NAXIS1 = ,JDIM(1)'
HEADER(4)=RECORD(2:80)
WRITE(RECORD,*) 'NAXIS2 = ,JDIM(2)'
HEADER(5)=RECORD(2:80)
ENDIF
C
IND=IND+1
KWORT=' '
KWORT=KEY(1:LEN(KEY))
L=LEN(TEXT)
VALUE=' '
IF(L.GT.64) THEN
  VALUE=TEXT(1:64)
ELSE
  VALUE=TEXT
ENDIF
WRITE(HEADER(IND),100) KWORT, VALUE
100 FORMAT(A8,T9,' = ',A64)
C
RETURN
END
```

C.3 Bildschirm-Ausdrucke des PDS-Steuerrechners

```
HEADER INFORMATIONSTEIL : <HOME> MENUE-ENDE
#####
BENUTZER :
PLATTENNAME :
TYPE DER PLATTE :
INSTRUMENT :
OBJEKT :
FILTER :
SPALT :
VERGROESSERUNG :
KOMMENTAR :
*****
GEWUENSCHTE FUNKTION :
```

```
PDS - VERSION 1.1 (31.05.84) <HOME> ZURUECK NACH KOS
#####
A *** ADJUST PEGEL BUFFER-KOPIE > VAX
COPY HILFEN>EPSON DATA TERMINAL-VAX
ERNEUT INITIIERUNG FUNCTIONS BENUTZER
GRAPHIK EINZELSCAN HEADER INFORMATIVS
ISOPHOTEN GRAPHIK JUMP>PDS-INTERFACE
K >>> KOORDINATEN LISTEN - PROGRAMME
M *** MODE T=0 L=1 NULLPUNKT
OPTIMIERTE GESCHW. PIXEL-GROESSE
QITTIERT NULLPUNKT R >>> RELATIV-KOR.
SCAN DEFINIERUNGEN TRANSFER ALT-LISTE
U >>> IST>SOLL-KOR V *** VERSTAERKUNG
W >>> W-SCAN X >>> X-SCAN
Y >>> Y-SCAN ZUSTAND EINSTELLEN
: : : : : DAUER-FUNKTIONEN : : : : :
^!(<) PIXEL-VERSATZ # HARDCOPY > EPSON
<RUB> PDS-ZUSTAND ? HILFE AM MONITOR
```

```
*****
GEWUENSCHTE FUNKTION :
```

```
ZUSTAND EINSTELLEN                                <HOME> MENUE-ENDE
#####
PIXEL GROESSE      :          100        FELDGROESSE 1-4/5:          3
K >>> KOORDINATEN
X-KOORDINATE      :          4096       Y-KOORDINATE      :          4096
NULLPUNKT
X-NULLPUNKT       :          0          Y-NULLPUNKT       :          0
R >>> RELATIV-KOR.
VERSTAERKUNG **2  :          5          ZEIGT GRAPHIK-FELD
: : : : : : : : : : EINSTELLUNG      : : : : : : : : : :
GS-OPT 0=NO 1=OPT:          1          PHOTO-MULTIPLYER :          851
MINIMALER WERT    :          0          MAXIMALER WERT   :          0
MODE  0=TR 1=LOG :          0          JETZIGE DMA-NUM.:          0
BUFFER 0=NO 1=BUF:          1          LICHT 0=TR 1=AUF:          0
OPTIM. GAIN+ADJUST
: : : : : : : : : : LISTEN  BEFEHLE  : : : : : : : : : :
COPY ZUR PRG-LISTE
INPUT PROG LIST-NR
HOLT  PROG LIST-NR
TRANSFER ALT-LISTE
LAENGE PROG-LISTE:          0
LAENGE KOOR-LISTE:          0
*****
GEWUENSCHTE FUNKTION :
```

```
SCAN DEFINIERUNGEN                                <HOME> MENUE-ENDE
#####
PIXEL GROESSE      :          100        FELDGROESSE 1-4/5:          3
K >>> KOORDINATEN
X-KOORDINATE      :          4096       Y-KOORDINATE      :          4096
NULLPUNKT
X-NULLPUNKT       :          0          Y-NULLPUNKT       :          0
R >>> RELATIV-KOR.
VERSTAERKUNG **2  :          5          ZEIGT GRAPHIK-FELD
: : : : : : : : : : EINSTELLUNG      : : : : : : : : : :
GESCHWINDIGKEITEN
X-GESCHWINDIGKEIT:          10000       Y-GESCHWINDIGKEIT:          10000
DIFFERENZ IN X-KO:          0          DIFFERENZ IN Y-KO:          0
LAENGE DES SCANS  :          20000      ZAHL DER PIXEL    :          200
SCAN-ZAHL         :          1          ZAHL DER DUMPS    :          0
MEANDER SCAN-ART :          0          W >>> WINKEL SCAN
X >>> X-SCAN
: : : : : : : : : : LISTEN  BEFEHLE  : : : : : : : : : :
ENDE DER PRG-LISTE
COPY ZUR PRG-LISTE
TYPE DES SCANS    :          1
LAENGE PROG-LISTE:          0
*****
GEWUENSCHTE FUNKTION :
```

### Verzeichnis der Abbildungen:

|           |                                                                                                       |    |
|-----------|-------------------------------------------------------------------------------------------------------|----|
| Abb 1:    | Feldspektrograph auf Fokalreduktorbasis (Schema)                                                      | 3  |
| Abb 2:    | Gitterprisma (links) und Doppelgitterprisma (rechts)                                                  | 4  |
| Abb 3:    | Silhouettierung bei einer Schmidtkamera mit geteilter Eintrittspupille                                | 7  |
| Abb 4:    | Vermeidung der Silhouettierung durch ein außeraxiales Schmidtkamerasystems                            | 8  |
| Abb 5:    | Schemazeichnung des neuen außeraxialen Schmidtsystems                                                 | 9  |
| Abb 6:    | Optisches System der Schmidt Kamera nach Angaben der Herstellerfirma Jenaoptik                        | 12 |
| Abb 7:    | Aufsicht und seitlicher Schnitt durch den Kassetten-Einschub                                          | 13 |
| Abb 8a-d: | Tageslichtspektren auf IIa-O Platte (T4CS1) und IIIa-J Platte (T4SP1 lin., T4SV1 2x, T4SL1 log.)      | 20 |
| Abb 9:    | Schematischer Scan über ein Doppelgitterprismen-Spektrum                                              | 21 |
| Abb 10:   | Normales, revertiertes und geflipptes revertiertes Spektrum untereinander dargestellt                 | 22 |
| Abb 11:   | Übereinstimmungsfunktion mit rel. Maxima                                                              | 22 |
| Abb 12ab: | He/Ar-Glimmlampe (A4SL1) und Fe-Hohlkathode (E4SL1)                                                   | 25 |
| Abb 13ab: | Normalisiertes Tageslichtspektrum (T4SN1) und dazu gehörendes Tageslichtspektrum (N4SL1) mit Glättung | 25 |
| Abb 14:   | Graphische Darstellung der Abweichung von einem linearem Ansatz für gemessene und berechnete Linien   | 27 |
| Abb 15:   | Vergleich der Spektrallampen (von oben: Hg, He, Cd, He/Ar, Fe, Tageslicht)                            | 29 |
| Abb 16:   | Zuordnung der Doppelgitterprismen-Spektren zu den Spalten der Vielfachspaltplatte                     | 35 |

## Danksagungen:

An erster Stelle möchte ich meinem Referenten Prof.Dr. E.H.Geyer für seine gute Betreuung während der Diplomarbeit danken. Zahlreiche Diskussionen mit ihm und den anderen Mitarbeitern am Observatorium Hoher List sowie in den Astronomischen Instituten in Bonn haben, durch die dort erhaltenen Anregungen, im wesentlichen zum Gelingen der Arbeit beigetragen. Von diesen möchte ich vor allem Dr. M.Hoffmann und Dr. M.Geffert, die mich in die Beobachtungspraxis eingewiesen haben, und Dipl.Phys. F.Decker für seine Einführung in das Programmpaket MIDAS und seine Ratschläge bei der Modifikation der verschiedenen MIDAS-Routinen danken.

Weiter möchte ich auch dem technischen Personal des Hohen Listes für das gute Betriebsklima und die zahlreichen Hilfen während der Diplomarbeit danken. Vor allem ist hier die Werkstatt unter der Leitung von Herrn Klink zu nennen, die das neue außeraxiale Schmidtkamerasystem gebaut hat. Zusätzlich gilt Herrn Klink das Verdienst, als Konstruktors dieser Kamera tätig gewesen zu sein.

Ermöglicht wurde diese Arbeit durch zahlreiche Sachspenden der Deutschen Forschungsgemeinschaft. Neben der Beschaffung der Optik für das außeraxiale Schmidtkamerasystem (Ge 209/10-4) sollen hier auch frühere Sachmittelspenden wie der Fokalreduktor und die Doppelgitterprismen genannt werden.

Schließlich sollte die Image Processing Group des European Southern Observatory (ESO) nicht unerwähnt bleiben. Zu ihr bestand zwar während der Diplomarbeit keine direkte Verbindung, jedoch wäre ohne ihr Programmpaket MIDAS die Erstellung der Diplomarbeit viel schwieriger und zeitaufwendiger gewesen wäre.

Noch vielen Mitarbeitern, Freunden und Verwandten müßte hier für ihre Hilfe gedankt werden. Alle, die hier nicht namentlich aufgeführt werden können, sollen dennoch nicht vergessen werden. Ihnen gebührt vielleicht sogar der größte Dank denn: "Ein Steinchen kann die Bahn einer Lawine verändern oder ein Zweig die Flut aufhalten." (Zitat aus L.Alexander, Taran und das Zauberschwert)