Steuerung, Überwachung und Dokumentation im EDDA–Experiment

Diplomarbeit
von
Olaf Diehl

April 1994

Referent: Prof. Dr. J. Bisplinghoff
Koreferent: Prof. Dr. J. Ernst

Steuerung, Überwachung und Dokumentation im EDDA–Experiment

von

Olaf Diehl

Diplomarbeit in Physik

angefertigt im
Institut für Strahlen– und Kernphysik

vorgelegt der
Mathematisch–Naturwissentschaftlichen Fakultät
der
Rheinischen Friedrich–Wilhelms Universität
Bonn

im April 1994

Ich versichere, daß ich diese Arbeit selbstständig verfaßt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie die Zitate kenntlich gemacht habe.

Referent:	Prof. Dr. J. Bisplinghoff
Koreferent:	Prof. Dr. J. Ernst

Inhaltsverzeichnis

Kapitel 1

Einleitung

Den Rahmen für diese Arbeit bildet das Experiment EDDA. EDDA (**E**xcitation Function **D**ata Acquisition **D**esigned for **A**nalysis of Phase Shifts) ist ein Präzisionsexperiment am Teilchenbeschleuniger COSY (**CO**oler **SY**nchrotron) des Forschungszentrums Jülich [bis90, bis93]. Es untersucht elastische Proton–Proton Streuung bei einer Strahlenergie von 250 MeV bis 2,5 GeV. Der elastische, differentielle Wirkungsquerschnitt soll mit großer relativer Genauigkeit bei gleichzeitig hoher Winkel– und Energieauflösung gemessen werden. Dies geschieht in einer ersten Phase des Experimentes spinunabhängig unter Verwendung eines CH_2–Fädchentargets. Darüber hinaus sollen in einer zweiten Phase mit polarisiertem Strahl bzw. einem polarisierten Atomstrahltarget Polarisationsobservable präzise gemessen werden. Der Weltdatensatz soll damit in diesem Energiebereich erheblich erweitert werden. Die angestrebte Qualität und der Umfang der Meßdaten sollen Streuphasenanalysen erlauben, die eine wichtige Schnittstelle zur Theorie bilden. Verschiedene theoretische Modelle können damit am Experiment überprüft werden. Dies gilt insbesondere für vorhergesagte Resonanzen, z.B. „Dibaryonen“.

EDDA ist ein internes Experiment an COSY, d.h. das Experiment ist Teil des Beschleunigerringes. Dies erlaubt es, in der Beschleunigungsrampe von COSY zu messen und damit den gesamten zur Verfügung stehenden Energiebereich in einem Zug zu überstreichen. Außerdem wird die Verwendung sehr „dünner“ Targets, insbesondere des polarisierten Atomstrahltargets, ermöglicht. Da Protonen, die ohne Wechselwirkung das interne Target durchdringen, nach einem Umlauf erneut für eine Reaktion mit dem Target zur Verfügung stehen, erhält man auch mit den dünnen Targets eine hohe Luminosität. Andererseits erfordert die Verwendung von Feststofftargets eine extrem geringe Massenbelegeung, weil sonst die Luminosität zu hoch und die Lebensdauer des Strahls zu kurz wird.

In der ersten Phase kommt ein Fädchentarget aus Polypropylen zum Einsatz. Polypropylen ist ein CH_2–Polymer, dessen chemisch gebundener Wasserstoff das eigentliche Protonentarget darstellt. Der durch den Kohlenstoff erzeugte Untergrund soll durch Messungen mit einem reinen Kohlenstoff–Fädchentarget korrigiert werden. Die Fädchentargets dürfen nur einige μm dick sein, um die extrem geringe Massenbelegung zu erreichen. Diese geringen Abmessungen sorgen zusammen mit der kleinen Breite des COSY–Strahls dafür, daß der Wechselwirkungspunkt der elastischen Proton–Proton–Streuung ausreichend genau bekannt ist.

Die Protonen, die das Fädchentarget durchdringen, lösen Sekundärelektronen aus. Die Messung dieses Sekundärelektronenstroms ist ein gutes Luminositätssignal [schβ94]. Allerdings setzt es ein elektrisch leitendes Fädchentarget voraus. Deshalb werden alle Targets mit Aluminium bedampft. Dies verhindert gleichzeitig, daß die Targets durch elektrostatische Aufladung zerstört werden.

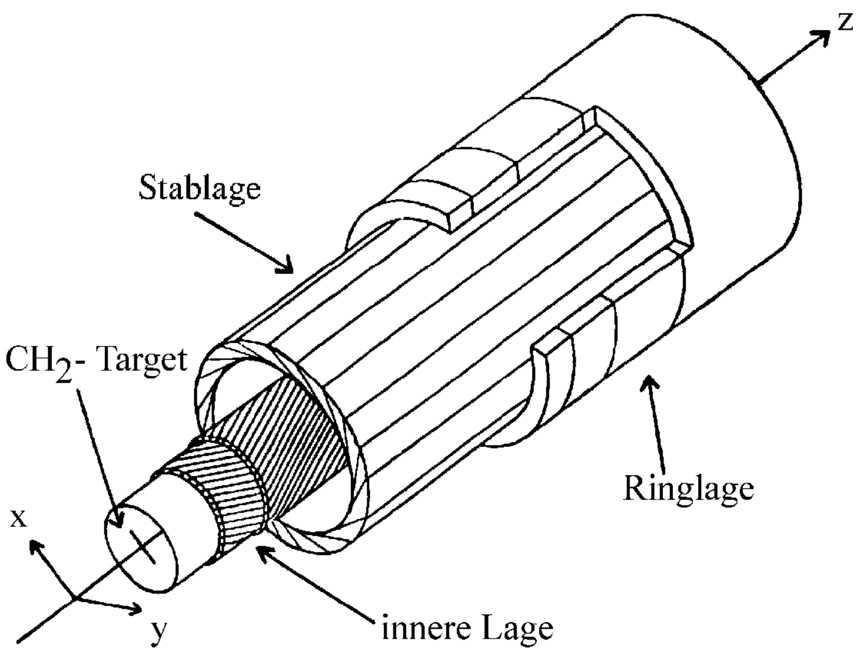

Abbildung 1.1: Der EDDA–Detektor

Abb. 1.1 zeigt eine Schemazeichnung des Detektors [lah92, gro93]. Auf der linken Seite, strahlaufwärts, befindet sich das Fädchentarget aus Polypropylen. Die äußere Schale des Detektors besteht aus Stablage und Ringlage. Die Stablage, bestehend aus 32 beidseitig auslesbaren Szintillatorstäben [ack93], dient der Winkelauflösung im Azimutwinkel ϕ. Die Winkelauflösung im Polarwinkel Θ wird durch die Ringlage erreicht, die aus 29 x 2 Halbringszintillatoren realisiert ist [sco93]. Die Signale der Außenschale werden auch zum Triggern auf elastische Proton–Proton–Streuung benutzt. Dazu nutzt man die Eigenschaften einer elastischen Zwei–Teilchen–Reaktion aus: Erstens gilt Koplanarität, d.h. die gestreuten Protonen müssen die Stablage an gegenüberliegenden Punkten durchstoßen (koplanarer Trigger). Zweitens liefert bei elastischen Reaktionen die Kinematik eine einfache Beziehung zwischen den beiden Streuwinkeln. Diese überträgt sich in eine einfache Beziehung zwischen getroffenen Ringszintillatoren. Eine Koinzidenz kinematisch assoziierter Ringe bildet dann den kinematischen Trigger. Koplanarer und kinematischer Trigger zusammen gewährleisten eine effektive Erkennung elastischer Ereignisse.

Die geometrische Akzeptanz, d.h. der Raumwinkelbereich, in dem Ereignisse nachgewiesen werden können, umfaßt 2π im Azimut sowie den Polarwinkelbereich von ca. $10°$ bis $72°$. Die Winkelauflösung innerhalb dieses Raumwinkels ist besser als die Granularität der Detektorelemente. Die Information, welcher der 32 Stabszintillatoren „gefeuert" hat, ergibt bereits eine azimutale Winkelauflösung von ca. $11°$. Die Ringszintillatoren lösen auf die gleiche Weise den polaren Winkel mit einer Genauigkeit von etwa $3°$ auf. Diese Auflösung wird noch folgendermaßen erhöht: Die Geometrie sowohl der Ringe als auch der Stäbe bedingt, daß immer zwei benachbarte Elemente gleichzeitig getroffen werden. Die Intensität der Signale („Amplitude") aus den Szintillatorelementen teilt sich je nach Durchstoßpunkt des Protons unterschiedlich auf die beiden Elemente auf. Umgekehrt kann aus dem Verhältnis der Amplituden auf den Durchstoßpunkt zurückgeschlossen werden. Dieses Verfahren der „Amplitudenteilung" erhöht die Winkelauflösung auf rund das Fünffache, das sind ca. $2°$ azimutal und ca. $0{,}6°$ im Polarwinkel.

Ein weiterer wichtiger Punkt ist die Auflösung der Strahlenergie. Durch die Verwendung des internen Targets ist es möglich, während der Beschleunigungsrampe von COSY zu messen. In diesem Modus erscheint eine Energieauflösung von ca. ein bis zwei MeV erreichbar. Dies bedeutet die Messung von rund 1000 Energieschritten innerhalb einer Beschleunigungsrampe. Die geringe Anzahl von Ereignissen pro Energieschritt und Beschleunigungsrampe wird durch die Messung vieler Rampen ausgeglichen („multi–pass–technique"). Dieses Verfahren schafft außerdem eine größere Unempfindlichkeit gegenüber Langzeitverschiebungen von Geräteeigenschaften und zufälligen Instabilitäten.

In einer zweiten Phase des EDDA–Experimentes wird zusätzlich in Abhängigkeit der Polarisation gemessen. Um möglichst viele Polarisationsobservablen messen zu können, sollen sowohl der COSY–Strahl als auch das Target polarisiert werden. Das Target soll dann ein polarisiertes Atomstrahltarget sein. Dieses bringt gegenüber dem Fädchentarget eine wesentlich größere Wechselwirkungszone mit sich. Um den Wechselwirkungspunkt rekonstruieren zu können, ist die innere Lage des Detektors entwickelt worden. Sie soll aus 640 szintillierenden Fasern bestehen, angeordnet in einer Doppelhelix (s. Abb. 1.1). Die feine Granularität dieser Doppelhelix erlaubt dann eine präzise Identifikation des Durchstoßpunktes. Zusammen mit dem Durchstoßpunkt durch die äußeren Lage sind dann die Trajektorien der Protonen bestimmt.

Die Reaktionsprodukte der Proton–Proton–Reaktionen werden im EDDA–Experiment nachgewiesen, indem sie beim Durchgang durch die Szintillatoren Licht erzeugen. Diese Lichtsignale werden dann mit Photomultipliern in elektrische Analogsignale umgewandelt. Für die Realisierung des Triggers müssen die Analogsignale diskriminiert, d.h. in logische Signale umgewandelt, sowie zur Einstellung von Koinzidenzen verzögert werden. Dazu müssen die Photomultiplier mit Hochspannung versorgt, die Schwellenspannung der Diskriminatoren sowie die Verzögerungszeit der Delays eingestellt werden. Berücksichtigt man die Verwendung von Vielkanaltechnik, ergibt sich im Falle des kompletten Detektors mit innerer und äußerer Lage folgender Anspruch an die Steuerung:

- Einstellung von ca. 160 Hochspannungskanälen
- Setzen von ca. 8 Diskriminatorschwellen, gültig für jeweils 16 Kanäle
- Einstellen von ca. 100 Verzögerungskanälen

Dieses durch Quantität dominierte Problem legt eine rechnergestützte Lösung nahe. Eine solche ermöglicht außerdem die Implementation von Überwachungs– und Dokumentationsfunktionen, die besonders für die Hochspannungsversorgung wichtig sind.

Aufgabe dieser Diplomarbeit war es, die entsprechenden Softwarelösungen zu erstellen, in das Datenaufnahmesystem des Experimentes zu intergrieren und im Einsatz zu testen (s. Kap. 3). EDDA benötigt desweiteren einen intensiven Datenaustausch mit der COSY–Leittechnik („COSY–Control“). Dieser dient insbesondere der Registrierung von Daten, die die Diagnoseelemente des Beschleunigers liefern. Sie sind zur Beurteilung der momentanen Strahleigenschaften zu jedem Zeitpunkt des Beschleunigungszyklus' wichtig, insbesondere für die Korrelation der Meßdaten mit Strahlenergie und Strahllage. Zur Realisierung des genannten Datentransfers konnte u.a. auf Softwarekonzepte, die ursprünglich für Steuerung und Überwachung von Hochspannungsversorgung etc. entwickelt wurden, zurückgegriffen werden.

Kapitel 2

Das Datenaufnahmesystem

Die Aufgabe des Datenaufnahmesystems ist es, die vom Detektor kommenden Signale zu selektieren, zu digitalisieren und über Rechner auf einem Massenspeichermedium abzulegen. Dazu soll im folgenden zunächst die verwandte Elektronik, danach das Datenaufnahmeprogramm auf dem Rechner betrachtet werden.

2.1 Elektronik

Der schematische Aufbau der Elektronik ist in Abb. 2.1 gezeigt. Die Analogimpulse vom Detektor werden zunächst mittels CAMAC–Elektronik weiterverarbeitet (CAMAC = **C**omputer **A**pplication to **M**easurement **a**nd **C**ontrol). Dies bedeutet zunächst eine Digita-

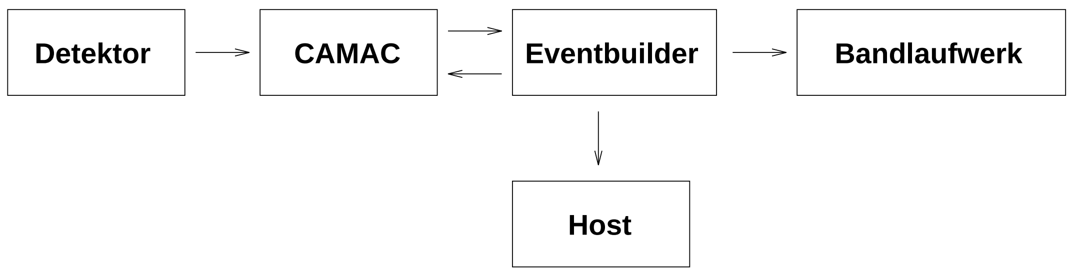

Abbildung 2.1: Hardware der Datenaufnahme

lisierung der Analogsignale. Diese digitalen Signale durchlaufen dann die Koinzidenzen der koplanaren und kinematischen Trigger. Ist die Triggerbedingung erfüllt, erfolgt eine „Alarmierung“ des Datenaufnahmerechners („Eventbuilder“). Der Eventbuilder ist ein ELTEC Eurocom – 6 Computer („E6“), der als VME–Modul (VME = **V**ersa **M**odule **E**urocard) realisiert ist. Er basiert auf einem Motorola 68030 Prozessor und arbeitet unter dem Echtzeitbetriebssystem OS–9. Auf diesem Eventbuilder läuft das Datenaufnahmeprogramm, das im nächsten Abschnitt beschrieben wird. In Reaktion auf die „Alarmierung“ durch die CAMAC–Elektronik bewerkstelligt es das Auslesen der digitalisierten Signale. Die so gewonnenen Daten werden formatiert und mit einem Exabyte Bandlaufwerk als Massenspeichermedium abgespeichert. Optional können die Daten auch zwecks Auswertung während der Messung („online Datenanalyse“) auf weitere Rechner übertragen werden. Das ist wichtig, um bereits während der Messung einen Überblick über die aufgenommenen Daten zu gewinnen.

2.2 Software

Das Datenaufnahmeprogramm geht auf die GSI Darmstadt (Gesellschaft für Schwerionenforschung) zurück. Dort wurde das als Provisorium gedachte Datenaufnahmesystem TDAS (**T**emporary **D**ata **A**cquisition **S**ystem) entwickelt. Es wurde 1990 vom EDDA–Experiment übernommen und ist seitdem immer wieder geändert, überarbeitet und ergänzt worden, nicht zuletzt im Rahmen dieser Arbeit.

Die jetzige Struktur des nach wie vor TDAS genannten Datenaufnahmeprogramms ist in Abb. 2.2 gezeigt. Ein Prozeß ist ein in Ausführung befindliches Programm unter einem Betriebssystem, das den parallelen Ablauf mehrerer Programme erlaubt; solche Prozesse sind als Ellipsen gezeichnet, alles andere ist durch eckige Kästen dargestellt. Der große gestrichelte Kasten umreißt den Eventbuilder.

Die zentrale Achse der Datenaufnahme ist die Linie CAMAC–TDASacquire–Ringbuffer–TDASarchive–Bandlaufwerk. Der Prozeß „TDASacquire“ liest die in CAMAC bereitstehenden Daten über den VDB Datenbus aus und legt sie formatiert zunächst in einem speziellen Hauptspeicherbereich ab. Dieser Speicherbereich ist mit Zugriffsrechten versehen, die aber von einem Prozeß zum nächsten weitergegeben werden können. Dieses Weiterreichen erfolgt zyklisch, weshalb diese Speicherbereiche auch „Ringbuffer“ genannt werden. Sie dienen als „Kreisverteiler“ für die Daten: Gefüllte Ringbuffer werden an den Prozeß „TDASarchive“ weitergereicht und auf Band geschrieben, worauf die Kontrolle wieder weitergereicht wird. Ein weiterer Nutzer der Ringbuffer ist der Prozeß „TDASaccessdc“. Seine Aufgabe ist es, bereits auf Band geschriebene Ringbuffer nochmals zu lesen und die Daten an den „Host“ Rechner weiterzuschicken. Auf dem „Host“ kann dann mit einem Datenauswerteprogramm überprüft werden, ob wirklich sinnvolle Daten aufgenommen werden. Gibt es mehr als einen „Host“, der Daten bekommen möchte, können mehrere Instanzen des Programms TDASaccessdc gleichzeitig laufen. Dann ist je ein Prozeß TDASaccessdc für jeweils einen „Host“ zuständig. Dieser Mechanismus zur Übertragung von Daten an „Hosts“ hat allerdings verminderte Priorität. Wird die Rechenzeit auf dem Eventbuilder knapp, hat das Schreiben auf Band Vorrang, und die „Hosts“ empfangen nur noch einen Ausschnitt der Datengesamtheit. Umgekehrt können aber auch Daten vom „Host“–Rechner in die Da-

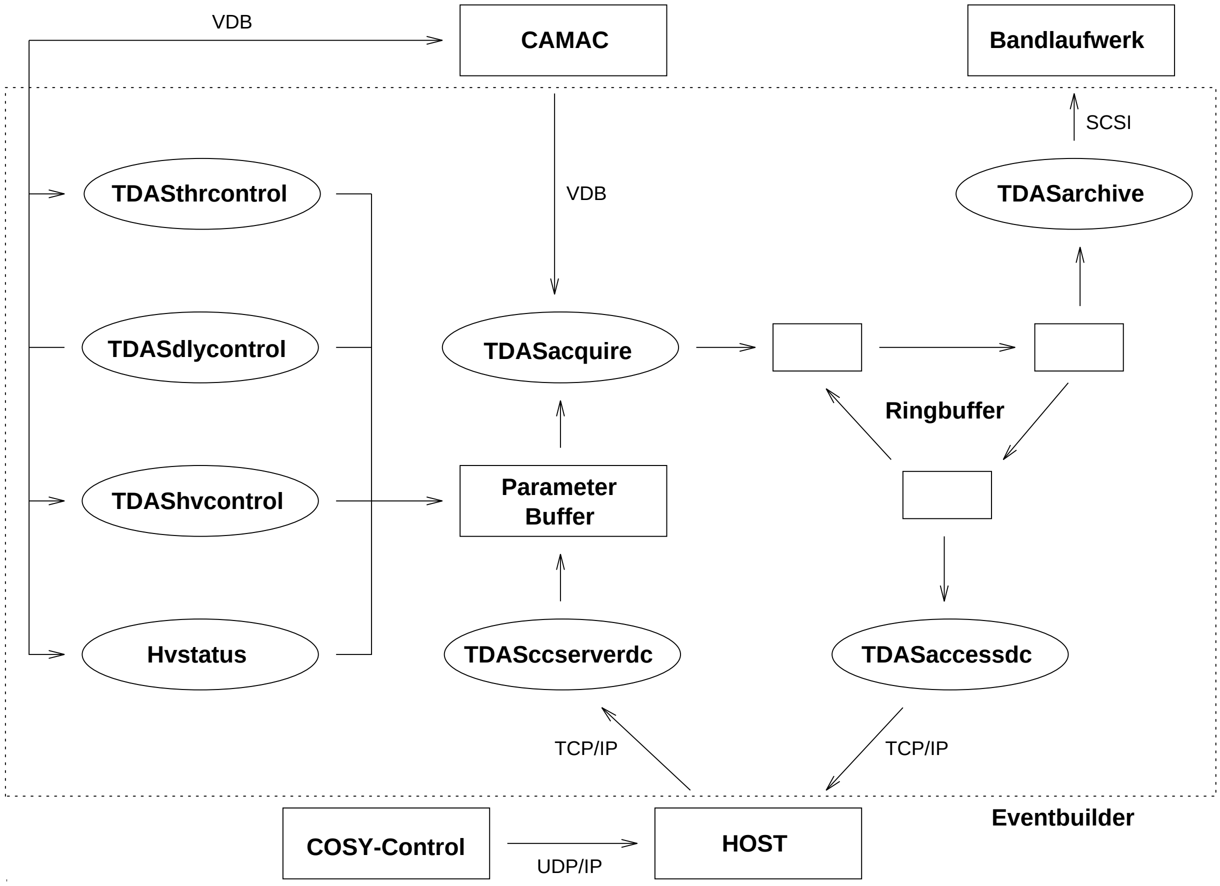

Abbildung 2.2: Struktur der Datenaufnahmesoftware

tenaufnahme auf dem Eventbuilder eingeschleust werden: Strahldiagnosedaten werden von den Rechnern der COSY–Leittechnik (COSY–Control) zunächst an den Host geschickt und gelangen über den Prozeß „TDASccserverdc“ zum „TDASacquire“, der diese Datensätze in den Rohdatenstrom integriert.

Daneben agieren die Kontrollprozesse TDASthrcontrol (Steuerung der Diskriminatorschwellen), TDASdlycontrol (Steuerung der Signalverzögerungen) und TDAShvcontrol („high voltage control“, Steuerung der Hochspannungsversorgung). Sie dienen der komfortablen Steuerung der entsprechenden Hardware über CAMAC. Gleichzeitig besteht die Möglichkeit, Information zur Dokumentation von Einstellungen an der Hardware mit auf Band zu schreiben. Im Falle der Hochspannungsversorgung findet eine unabhängige, fortlaufende Überwachung durch den Prozeß HVstatus statt.

Die Bedienung des Datenaufnahmesystems durch den Anwender erfolgt durch das „shell–script“ TDASmenu. Eine „shell“ ist ein Kommandointerpreter, der als Benutzerschnittstelle zum eigentlichen Betriebssystem fungiert, und ein „shell script“ ist eine in einer Datei abgelegte Liste solcher „shell“–Kommandos. TDASmenu ist ein menügesteuertes System und stellt generelle Funktionen wie Starten und Stoppen der Datenaufnahme, Steuern der Hardware sowie Einstellen von Optionen zur Verfügung. Außerdem kann das Programm TDASmonitor aufgerufen werden, welches dann periodisch einen Statusbericht über das gesamte System anzeigt. Dieser Statusbericht enthält z.B. die Anzahl der aufgezeichneten Ereignisse, die augenblickliche Totzeit und den letzten Eintrag in die Logdatei von TDAS.

Kapitel 3

Implementation von Steuerung und Überwachung

Die Stellung der neu entwickelten Steuerprozesse TDAShvcontrol, TDASthrcontrol und TDASdlycontrol sowie des Überwachungsprogramms HVstatus innerhalb des gesamten Datenaufnahmesystems ist in Abschnitt 2.2 dargestellt. Nun folgt eine detailierte Beschreibung der einzelnen Steuer– und Überwachungsprozesse.

3.1 Hochspannungsversorgung

Die Photomultiplier haben die wichtige Aufgabe, die Lichtsignale aus den Szintillatoren in elektrische Signale umzuwandeln und soweit zu verstärken, daß sie der nachfolgenden Elektronik zugänglich sind. Um diese Aufgabe zu erfüllen, müssen sie stets mit der richtigen Hochspannung versorgt werden. Dazu wird eine Hochspannungsversorgung eingesetzt, die rechnergestützt eingestellt und überwacht wird. Die Software, die dies leistet, wird im folgenden Abschnitt beschrieben. Aufgrund der Wichtigkeit einer kontinuierlichen Hochspannungsversorgung ist außerdem ein unabhängiges Überwachungsprogramm entwickelt worden, welches anschließend erläutert wird.

3.1.1 Steuerung

An eine leistungsfähige Steuerung der Hochspannungsversorgung (HV) im EDDA–Experiment werden folgende Anforderungen gestellt:

- sichere, schnelle und übersichtliche Einstellung einer großen Zahl von Hochspannungskanälen

- Einbindung in das bestehende Datenaufnahmesystem TDAS

- Dokumentation der Einstellung der Hochspannungsversorgung auf dem Datenband

Das Hardware–Konzept der Hochspannungsversorgung verdeutlicht Abb. 3.1. Herzstück der Konzeption sind die Hochspannungsversorgungen vom Typ SY 127 der Firma C.A.E.N. Jeder der Überrahmen HV 0, HV 1... nimmt bis zu 40 Hochspannungskanäle auf. Bis zu 100 solcher Überrahmen können mit einer seriellen Leitung verbunden und über eine einzige Verbindung zur CAMAC–Norm („HV–Controller") angesprochen werden. Dieser HV–Controller erhält seine Befehle wiederum über VDB (VME Differential Bus) vom Eventbuilder, der allgemein für die Bedienung der CAMAC–Elektronik verantwortlich ist. Der „CBV" in Abb. 3.1 ist ein CAMAC–Controller, der die Verbindung zwischen VDB und dem CAMAC–internen Datenbus herstellt.

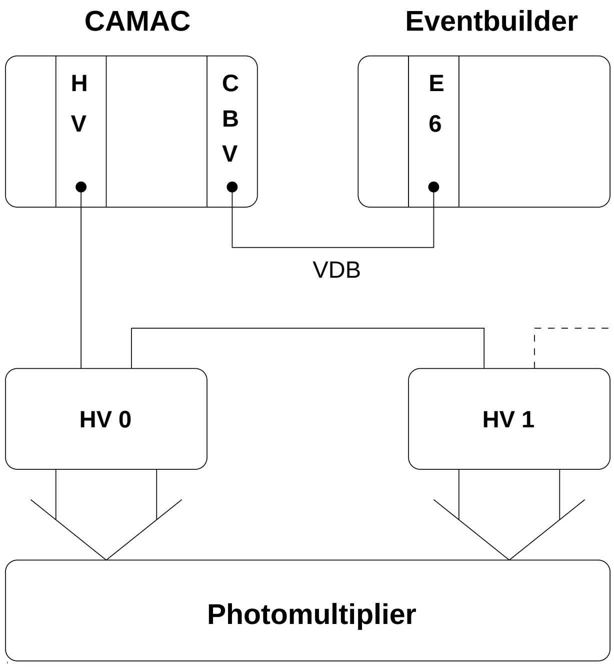

Abbildung 3.1: Hardwarekonzept der Hochspannungsversorgung

Das Steuerungsprogramm der Hochspannungsversorgung läuft auf dem Eventbuilder. Ihm liegt folgendes Prinzip zugrunde: Auf der einen Seite gibt es einen Satz von beschreibenden Textdateien. Das sind (s. Anhang A.2 bis A.4):

Konfigurationsdateien: Sie beschreiben in tabellarischer Form eine Konfiguration der Hochspannungsversorgung.

Ergebnisdateien: Ergebnisdateien beschreiben den tatsächlichen, durch Zurücklesen ermittelten Zustand der Hochspannungsversorgung.

Kanalauswahldateien: Sie geben an, welche Hochspannungskanäle zurückgelesen und in einer Ergebnisdatei abgelegt werden sollen. Konfigurationsdateien können als Kanalauswahldatei benutzt werden.

Auf der anderen Seite gibt es Funktionen, die die Steuerung der Hochspannungsversorgung unter Zuhilfenahme dieser Textdateien erlauben. Das sind z.B. Übertragen der Parameter einer Konfigurationsdatei in die Hardware der Hochspannungsversorgung und umgekehrt. Diese Funktionen sind in einem Menü zusammengefaßt, das die Benutzerschnittstelle zum Steuerprogramm der Hochspannungsversorgung darstellt. Dies ist das TDAShvcontrol–Untermenü in Abb. 3.2. Es ist ein Teil von TDASmenu, dem Hauptmenü zur Steuerung der Datenaufnahme. Die Menüfunktionen, welche Zugriff auf die Hardware der Hochspannungsversorgung erfordern, werden von dem Prozeß TDAShvcontrol ausgeführt. Alle benötigten Parameter werden über die Konfigurations– und Ergebnisdateien ausgetauscht. Über den Prozeß TDASacquire können Ergebnisdateien mit auf das Datenband geschrieben werden. Hierzu wird ein Beispiel in Anhang A.5 gegeben.

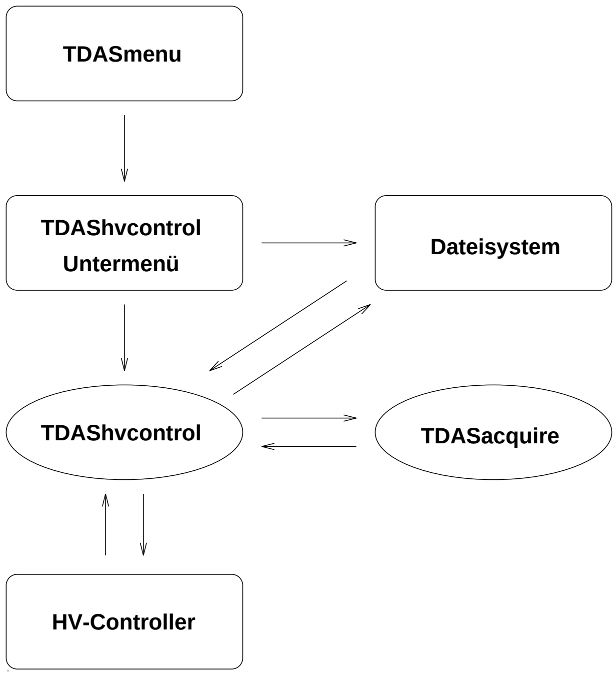

Abbildung 3.2: Aufbau des HV–Steuerprogramms

Diese Konzeption erfüllt die gestellten Anforderungen. Die Konfigurationsdateien lassen eine übersichtliche und sichere Einstellung der Hochspannungsversorgung zu. Die niedrige Geschwindigkeit (s. Tab. 3.1) wird durch die serielle Verbindung zwischen HV–Controller und den Hochspannungskanälen bestimmt. Die Steuerung ist in TDAS eingebunden, ohne davon abhängig zu sein, denn es wird nur die Menügestaltung sowie die Möglichkeit, auf Band zu schreiben, gemeinsam genutzt. Im einzelnen zeichnet sich diese Konzeption durch folgende Vorteile aus:

- Flexibilität: Zur Manipulation von Textdateien existieren bereits von vorne herein viele Werkzeuge, z.B. Editoren zum Modifizieren der Textdateien. Ebenfalls wird der Preprozessor der Programmiersprache C benutzt. Dieses Textersatzsystem ermöglicht es z.B., an mehreren Stellen benötigte Information in einer einzigen Datei zu zentralisieren (s. Anhang A.2).
- Modularität: Verwaltung und Bearbeitung der Textdateien einerseits und die Kommunikation mit dem HV–Controller andererseits sind voneinander unabhängig. Dies zahlt sich z.B. aus, wenn man eine komfortablere Benutzeroberfläche der HV–Steuerung auf dem Host erstellt, weil dann nur die Handhabung der Textdateien, aber nicht die Kommunikation mit dem HV–Controller erneuert werden muß.
- Einfachheit: Durch die Einbeziehung eines vorhandenen Editors und C–Preprozessors ist der Programmieraufwand und damit auch die Fehleranfälligkeit gering.

Diese Vorteile waren ausschlaggebend für die gewählte Konzeption. Nähere Informationen zur Implementation sind Anhang A zu entnehmen.

3.1.2 Überwachung

Eine einwandfreie Hochspannungsversorgung ist für den Betrieb der Photomultiplier und mithin des gesamten Detektors Voraussetzung. Eine ständige Überwachung der Hochspannungsversorgung ist daher angemessen. Ziel einer solchen Überwachung muß es sein, eine Fehlfunktion der HV rechtzeitig wahrzunehmen, auffällig anzuzeigen und ggf. auf dem Datenband zu dokumentieren. Zu diesem Zweck ist das Überwachungsprogramm HVstatus entwickelt worden (s. Anhang D).

HVstatus ist ein vom Datenaufnahmesystem TDAS unabhängiges Programm, das optional gestartet werden kann (s. auch Abb. 2.2). Seine Aufgabe wird durch die zur Verfügung stehende Überwachungsfunktionalität der HV–Hardware erleichtert: Tritt eine Fehlerbedingung (z.B. Überschreitung des maximal zulässigen Stromes) in irgendeinem Kanal auf, so wird diese automatisch in einem Statuswort im HV–Controller sichtbar. Die Aufgabe des Überwachungsprogramms besteht also vorwiegend darin, dieses Statuswort zu überwachen. Konkret werden folgende Fehlerbedingungen automatisch überwacht:

- Überschreitung des programmierten Stromlimits
- Abweichung von der Sollspannung

Die Ansprechschwelle für Abweichungen von der Sollspannung beträgt bei den verwendeten Modulen A133P/N 50V [caeα91]. Ist eine Fehlerbedingung erkannt worden, werden folgende Aktionen ausgeführt:

- Es erfolgt ein Eintrag in eine HV–Logdatei, die alle wichtigen Aktionen der HV–Steuerung enthält.
- Der Status der überprüften Hochspannungsversorgung wird mit einem auffälligen Hinweis auf den Bildschirm geschrieben.
- Eine Fehlermeldung wird in die Logdatei des Datenaufnahmesystems TDAS geschrieben und im TDASmonitor angezeigt.
- Optional kann eine Fehlermeldung mit auf das Datenband geschrieben werden.

Eine über das Statuswort hinausgehende Information über einzelne Kanäle kann nur durch Auslesen der gesamten HV–Parameter im Rahmen der Bedienung durch das Steuermenü erfolgen. Dem Programm HVstatus sind diese Informationen grundsätzlich nicht zugänglich, da die Hardware der CAMAC–Überrahmen keinen kontrollierten Zugriff gleichzeitig durch die Datenaufnahme und das Programm HVstatus zuläßt (s. Abschnitt 3.4).

3.2 Diskriminatoren

Die Diskriminatoren werden gebraucht, um die analogen Signale aus der äußeren Lage des Detektors in logische Signale umzuwandeln. Mit diesen logischen Signalen entscheidet die Triggerlogik, welche Ereignisse akzeptiert und abgespeichert werden. Verwandt werden CAMAC–Diskriminatoren vom Modell 4413 der Firma LeCroy. Einstellbarer Parameter dieser Diskriminatoren ist die Schwellenspannung, oberhalb der ein Analogsignal als Signal eines durchgegangenen Teilchens gewertet wird.

Anspruch an eine Diskriminatorsteuerung ist es, die Schwellenspannungen vieler Diskriminatoren sicher und einfach einzustellen. Nach einer Justierphase wird die Einstellung in der Regel einmalig am Beginn der Messung erfolgen, so daß eine automatische Einstellung am Anfang einer Messung besonders nützlich ist. Dies wird durch ein Konzept verwirklicht, das dem der HV–Steuerung ähnelt (s. Abb. 3.3). Unterhalb der zentralen Benutzerschnittstelle TDASmenu gibt es ein Untermenü für die Steuerung der Diskriminatoren (TDASthrcontrol submenu). Von hier aus werden alle Befehle zum Steuern und Dokumentieren der Diskriminatorschwellen abgesetzt. Ausgeführt werden sie vom Programm TDASthrcontrol, welches die notwendigen Einstellungen in den CAMAC–Modulen vornimmt. Die gewünschten Schwellen werden aus einer Tabelle entnommen. Das ist wie bei der HV–Steuerung eine Konfigurationsdatei, die mit einem Editor geschrieben und modifiziert wird (s. Anhang B.2 für eine detailierte Beschreibung der Menüfunktionen). Optional kann eine automatische

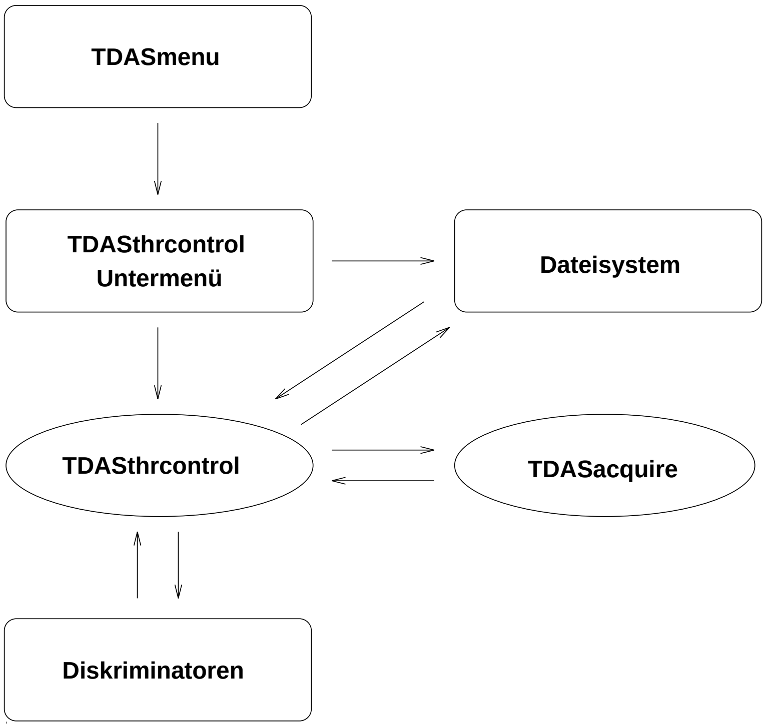

Abbildung 3.3: Aufbau der Diskriminatorsteuerung

Einstellung der Diskriminatorschwellen beim Starten der Datenaufnahme erreicht werden. In diesem Fall bekommt der Prozeß TDASthrcontrol ein Signal (Interrupt) vom Prozeß „TDASacquire", woraufhin eine fest vereinbarte Konfigurationsdatei ausgewertet und die entsprechenden Einstellungen vorgenommen werden.

3.3 Signal–Verzögerungen

Die Steuerung der Signal–Verzögerungen (Delays) ist ebenso wie die Steuerung der Diskriminatoren aus dem entsprechenden Programm für die Hochspannungsversorgung hervorgegangen. Dies mindert jedoch nicht die Leistungsfähigkeit, im Gegenteil wird die gesamte Leistungsstärke der komplexeren HV–Steuerung mit übernommen.

Gebraucht werden die Delays zu Synchronisationszwecken, z.B. innerhalb des Triggers. Der Trigger definiert physikalisch interessante Ereignisse und veranlaßt deren Auslese und Speicherung. Beim EDDA–Experiment überprüft er dazu die koplanare und kinematische Bedingung der interessierenden elastischen Proton–Proton Streuung. Die dazu notwendigen Koinzidenzen zwischen den Signalen der beiden Protonen erfordern einen genauen Abgleich der Signallaufzeiten, der durch Justieren der programmierbaren Delays erreicht wird.

Die Struktur des erstellten Steuerprogramms für die Delays entspricht derjenigen der

Diskriminatorsteuerung (s. Abschnitt 3.2). Einziger Unterschied ist, daß die Hardware der Delays kein Zurücklesen der eingestellten Parameter erlaubt. Eine Beschreibung der zur Verfügung stehenden Kommandos gibt Anhang C.1.

3.4 Tests und Ergebnisse

Die Programme zur Steuerung von Hochspannungsversorgung, Diskriminatoren und Delays sind alle zunächst innerhalb des ISKP erstellt worden. Zum Testen stand ein E6–Rechner mit der Datenaufnahmesoftware, ein CAMAC–Überrahmen mit HV–Controller sowie ein HV–Überrahmen mit 40 Hochspannungskanälen zur Verfügung. Bei ersten Tests unter Laborbedingungen trat folgendes Problem auf: Ist ein Kommando an den HV–Controller gesendet worden, so braucht dieser eine beträchtliche Zeit (mehrere 1/100 sec.), um den Befehl abzuarbeiten. Die Erfahrung hat gezeigt, daß der HV–Controller innerhalb dieser Zeit nicht angesprochen werden darf, da er sonst in einen unkontrollierbaren Zustand geraten kann. Dieses Problem wird so gelöst, daß nach jedem Zugriff auf den HV–Controller eine vorgegebene Zeit gewartet wird, bis er wieder betriebsbereit ist. Dadurch wird die Arbeitsgeschwindigkeit der HV–Steuerung unvermeidlich herabgesetzt (s. Tab. 3.1). Die in der Tabelle genannte Anzahl Parameter bezieht sich darauf, wieviele Parameter (z.B. Spannung, nicht zu überschreitendes Stromlimit) tatsächlich gesetzt werden müssen. Die

Aktion	Geschwindigkeit Kanäle/sec.
Schreiben, 1 Parameter	6.4
Schreiben, 4 Parameter	2.3
Schreiben, alle (8) Parameter	1.2
Lesen	1.0
Lesen, ohne V1, I1	1.2

Tabelle 3.1: Geschwindigkeit der HV–Steuerung

Variation entsteht dadurch, daß beim Einstellen der Hochspannungsversorgung einzelne Parameter eines Kanals unverändert übernommen werden können. Beim Auslesen werden immer alle Parameter eines Kanals gelesen. Dies ist in der Schwierigkeit begründet, jeweils zu definieren, welche Parameter welchen Kanals von Interesse sind. Die Parameter Alternativspannung/–stromlimit V1/I1 werden nicht genutzt. Um eine Nutztung in der Zukunft nicht auszuschließen, kann bei der Übersetzung des Programms angegeben werden, ob V1/I1 ausgelesen werden sollen oder nicht. Ihren ersten Einsatz hat die HV–Steuerung in der ersten EDDA–Strahlzeit (23.8–31.8.93) gehabt. Es wurde mit Teilen der äußeren Detektorschale erstmals an COSY gemessen. Die Handhabung der ca. 20 benötigten Hochspannungskanäle erfolgte erfolgreich durch die HV–Steuerung. Im Detail traten zwei Probleme zutage:

- Die Datenaufnahme und die Kommunikation mit dem HV–Controller können nicht gleichzeitig erfolgen, sondern nur nacheinander.
- Eine Änderung der Einschubposition des HV–Controllers, die zumindest von einer Strahlzeit zur nächsten auftreten kann, ist zu umständlich.

Ersteres Problem ist auf die Hardware der CAMAC–Überrahmen zurückzuführen. Diese stellen nur eine Q– und X– Leitung, die Auskunft über den Erfolg eines CAMAC–Befehls geben, für alle CAMAC–Module gemeinsam zur Verfügung [lec90]. Greifen zwei Prozesse „gleichzeitig“ auf verschiedene CAMAC–Module zu, kann es passieren, daß diese sich gegenseitig den Inhalt der Q– und X– Leitungen überschreiben, bevor er ausgewertet werden konnte. Der „gleichzeitige“ Zugriff verschiedener Programme auf verschiedene CAMAC Module ist also prinzipiell nur möglich, falls die Q– und X–Leitungen nicht gebraucht werden, was aber bei der HV–Steuerung der Fall ist. Einzige Milderung des Problems ist die Automatisierung einer vorübergehenden Unterbrechung der Datenaufnahme zum Auslesen der HV (s. Anhang A.1).

Letzteres Problem ist durch Anwendung des Preprozessors der Programmiersprache C gelöst worden. Dieser erlaubt es, die Information über die Position des HV–Controllers im CAMAC–Überrahmen für viele Konfigurationsdateien gemeinsam in einer Datei abzulegen. Eine Änderung ist in dieser zentralen Datei leicht zu bewerkstelligen und wird gleichzeitig für alle Konfigurationsdateien wirksam (s. Anhang A.2).

Anfang September 1993 ist die HV–Steuerung auch im Zuge des MOMO–Experimentes (s. [jah89, jah91]) eingesetzt worden. In einer Strahlzeit am Teilchenbeschleuniger SATURNE in Saclay (nahe Paris) wurde der Teil einer „Vertexwand“, bestehend aus 112 szintillierenden Fasern, getestet. Die Betrieb der ca. 15 benötigten Hochspannungskanäle zur Versorgung der Photomultiplier erfolgte mit der vorgestellten HV–Steuerung.

In den folgenden Strahlzeiten des EDDA–Experimentes vom 25. bis 29. Oktober und vom 6. bis 10. Dezember 1993 kam auch die Software zum Steuern der Diskriminatoren zum erfolgreichen Einsatz. Die entsprechende Software für die Delays war bereits fertiggestellt, es wurden aber noch keine Delays eingesetzt. Sie ist aber bei einer Laufzeitmessung der kompletten Triggerlogik benutzt worden.

Das Programm HVstatus hat sich in diesen Strahlzeiten als zu unauffällig erwiesen. Seine Meldungen wurden von der Vielzahl anderer Software zur Datenaufnahme und –analyse überdeckt. Deshalb ist es noch einmal überarbeitet und mit weiteren Möglichkeiten zur Dokumentation einer Fehlfunktion, z.B. Anzeige im TDASmonitor, ausgestattet worden (s. Anhang D).

Kapitel 4

Anbindung an COSY–Control

Die Leittechnik des Teilchenbeschleunigers COSY ist so aufgebaut, daß jede Komponente des Beschleunigers (ein Magnet, Gruppen von Magneten, die Beschleunigungskavität etc.) eine „intelligente“ Steuerung besitzt. Soll ein Teilchenstrahl mit bestimmten Eigenschaften bereitgestellt werden, müssen alle Steuerungen („VME Targets“) dieser Einzelgeräte zunächst mit einem „Fahrprogramm versorgt“ werden. Auf ein Startsignal von der Leittechnik (COSY–Control) hin wird dieses Fahrprogramm dann ausgeführt. Strahleigenschaften wie Strahllage und –intensität werden durch Strahldiagnosegeräte aufgezeichnet, haben aber keine regelnde Rückwirkung auf den Beschleuniger. Wichtig sind diese Messungen für die Strahlentwicklung sowie für die Kontrolle der Strahlqualität im nachhinein.

Das EDDA–Experiment mißt innerhalb der Beschleunigungsrampe von COSY, d.h. es bedarf einiger Anstrengung, um Kenntnis über die momentane Strahlenergie zu bekommen. Neben den eingestellten Fahrprogrammen muß auch die genaue Zeit innerhalb des Fahrprogramms von COSY bekannt sein. Wie dieses Problem gelöst wird, ist in Abschnitt 4.1 beschrieben. Daneben sind auch Strahldiagnosedaten von COSY interessant. Z.B. geben die Strahlpositionsmonitore (Beam Position Monitors, BPM's) Auskunft sowohl über die Strahlintensität als auch über die Strahllage. Die Strahlintensität kann der Überprüfung eigener Luminositätsmessungen dienen (z.B. Messung des Sekundärelektronenstromes aus dem Fädchentarget, s.[schβ94]). Die Lage des Strahls ergibt Aufschluß über Instabilitäten innerhalb der Beschleunigungsrampe und dadurch drohenden Artefakten. Wie diese COSY–internen Daten dem EDDA–Experiment zugänglich gemacht werden, ist im Abschnitt 4.2 beschrieben.

4.1 Das Timing System

Innerhalb von COSY gibt es zwei verschiedene lokale Netzwerke: ein Daten–LAN (**L**ocal **A**rea **N**etwork) sowie ein Timing–LAN (s. Abb. 4.1). Das Daten–LAN dient der Kommunikation zwischen COSY–Control und den einzelnen Komponenten des Beschleunigers. Beispiele für die Nutzung des Daten–LAN's sind die Übertragung von Fahrprogrammen auf die VME–Targets von COSY, Abrufen von Strahldiagnosedaten und die Programmierung des Senders (s.u.).

Das Timing–LAN dient der Synchronisation der verschiedenen Komponenten von COSY untereinander sowie der Experimente mit dem Beschleuniger. Dazu schickt ein

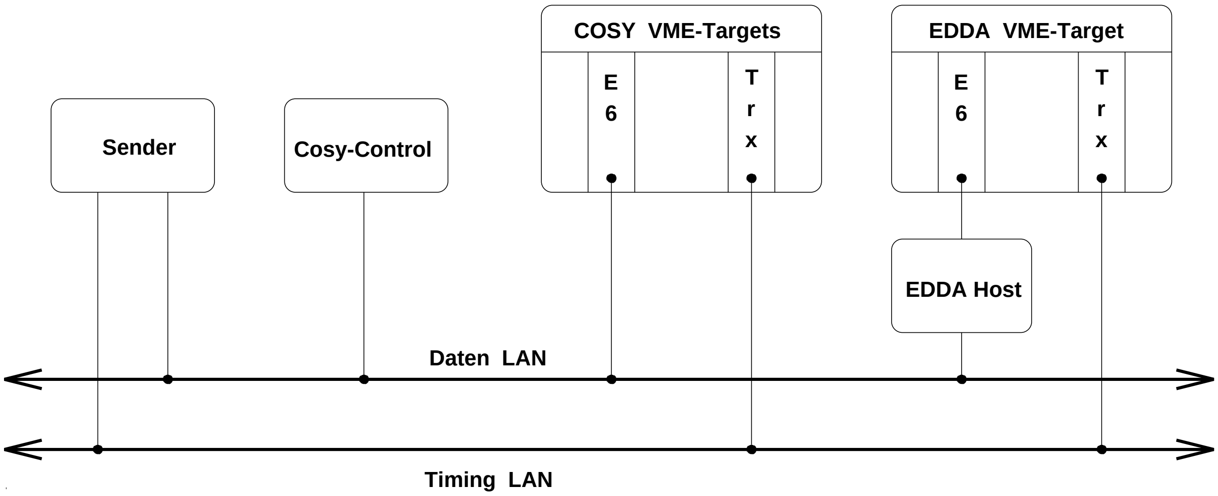

Abbildung 4.1: Das Timing System

Sender, vorprogrammiert durch COSY–Control, jede Millisekunde ein Datenpaket über das Timing–LAN. Die Empfänger dieser Datenpakete sind die VME–Targets, die dadurch ggf. zur Ausführung einer vorgegebenen Aktion angestoßen werden. Das sind zum einen die VME–Targets von COSY, z.B. Netzgeräte und die Beschleunigungskavität, sowie der Eventbuilder des EDDA–Experimentes. Jedes solche VME–Target besteht aus einem SAC30–Einschub in einem VME Überrahmen, der einen Dekodierer für die Datenpakete des Senders („Timing–Receiver“) enthält. Aufgabe des Timing–Receivers ist es, nach der Dekodierung der Datenpakete Zeitmarken in Form von TTL–Pulsen mit großer Zeitgenauigkeit (100 Nanosekunden) zu erzeugen.

Abb. 4.2 zeigt ein typisches Datenpaket des Senders. Es besteht aus Befehlskodes, die 16 Bit (einen „short integer“) umfassen. Die erste Ziffer der hexadezimal dargestellten Zahlenwerte bezeichnet den Befehl, die folgenden drei Ziffern den Operanden des Befehls („xxx“ usw. steht für variable Operanden). „8000“ bedeutet z.B. Befehl „8“, das ist Starten eines internen Zählers (s.u.). Der Operand „000“ wird in diesem Fall ignoriert. Der Befehl „5“ erzeugt einen zeitgenauen TTL–Puls an Ausgang eins. Der Operand des Befehls „yyy“ selektiert den yyy–ten Wert in einer Tabelle für Timer 1, T_1(yyy). Diese Tabelle muß zuvor prepariert worden sein. Ist T_1(yyy) gleich -1, wird der Befehl ignoriert, ansonsten wird

Ethernet Paket						
...	STOP	STATUS	TIMER 1	TIMER 2	START	...
...	2000	1xxx	5yyy	6zzz	8000	...

Abbildung 4.2: Beispiel eines Datenpakets auf dem Timing–LAN

nach $100T_1(yyy)$ Nanosekunden der gewünschte TTL–Puls ausgegeben. Die Verzögerungszeit wird dadurch abgemessen, daß ein Zähler mit einer Frequenz von 10 MHz bis $T_1(yyy)$ zählt. Dieser Zähler wird durch den „START“–Befehl in Gang gesetz.

Auf die gleiche Weise kann ein TTL–Puls am zweiten Ausgang (Timer 2) zeitgenau erzeugt werden. Im Falle des Befehls „1“ (STATUS) wird der Operand ohne Verzögerung auf dem parallelen Port des Timing Receivers zur Verfügung gestellt.

Dieser Mechanismus erlaubt es also, Zeitmarken mit 100 Nanosekunden Genauigkeit zu generieren, die auf einen bekannten Nullpunkt innerhalb der Fahrkurven von COSY bezogen sind. Damit ist eine genaue Synchronisation der Experimente mit COSY möglich. Anwendung im EDDA–Experiment findet dieses Verfahren bisher beim Fahren des Fädchentargets in den Strahl und wieder heraus. Die beiden TTL–Ausgänge heißen dort ihrer Verwendung wegen Start und Stopp: Der Startpuls initiiert das Fahren des Targets in den Strahl, der Stoppuls läßt es wieder herausfahren.

Des weiteren kann mit den TTL–Pulsen eine Uhr gestartet werden, so daß ständig eine genaue, mit COSY synchronisierte Uhrzeit zur Verfügung steht. Eine Anwendung dieser Zeitinformation ist die Bestimmung der jeweiligen Energie des Teilchenstrahls. Das ist wichtig, da in der Beschleunigungsrampe von COSY gemessen wird. Voraussetztung für eine erfolgreiche Bestimmung der Strahlenergie ist einerseits die Kenntnis des Fahrprogramms von COSY. Anderseits muß gewährleistet sein, daß der Strahl den Fahrprogrammen präzise folgt. Dies kann mit den Strahldiagnosegeäten von COSY überprüft werden. Wie die Strahldiagnosedaten dem EDDA–Experiment zugänglich sind, wird im folgenden Abschnitt beschrieben.

4.2 Daten von COSY–Control

COSY–Control verfügt über eine Reihe von Informationen über den Beschleuniger, die auch für das EDDA–Experiment wichtig sind. Dies sind zum Beispiel die Fahrprogramme, die den Verlauf der Strahlenergie in der Beschleunigungsrampe beschreiben. Ohne diese Fahrprogramme ist es nicht möglich, den gemessenen Ereignissen eine Strahlenergie zuzuordnen (s. Ende des letzten Abschnitts). Desweiteren betreibt COSY Strahldiagnosegeräte die Informationen über den tatsächlich vorhandenen Strahl gewinnen. Das sind z.B. Intensität und Abweichung des Strahls von der Sollbahn. Die Intensitätsmessungen können benutzt werden, um Redundanz zu den eigenen Luminositätsmessungen zu erreichen. Redundanz ist erstrebenswert, weil ein Fehler in der Bestimmung der Luminosität sofort einen Fehler im gemessenen Wirkungsquerschnitt bedeutet. Ebenso helfen die Strahlpositionsdaten, Artefakte in den Meßdaten aufgrund von Instabilitäten des Strahls zu erkennen.

Deshalb ist ein Weg geschaffen worden, diese Daten von COSY–Control mit auf dem Datenband zu speichern. Wie dies funktioniert, wird im folgenden am Beispiel von Diagnosedaten der Strahlpositionsmonitore beschrieben. Die Strahlpositionsmonitore bestehen

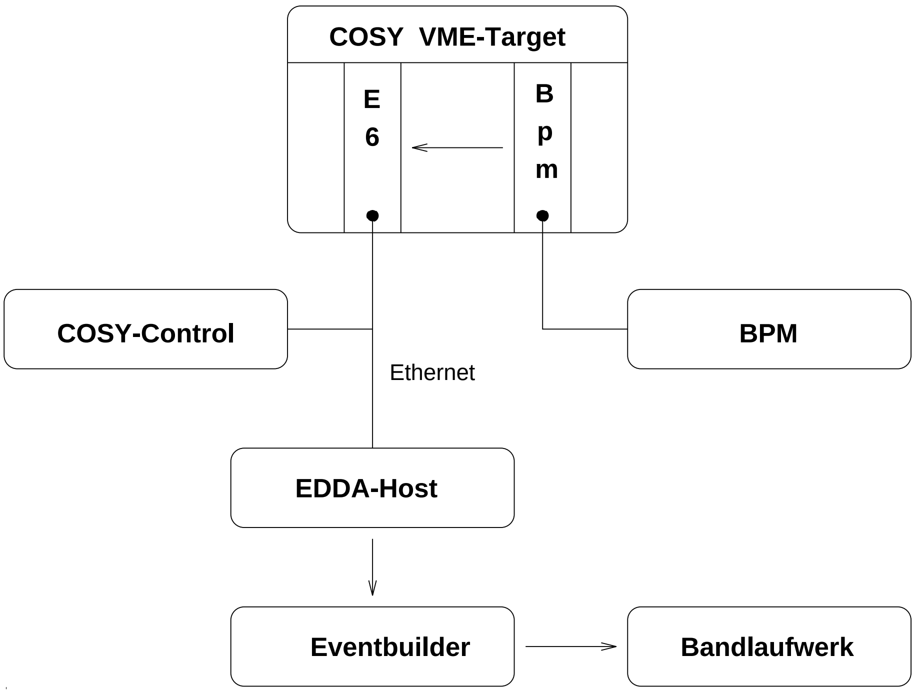

Abbildung 4.3: Übertragung von Strahldiagnosedaten

im Wesentlichen aus einem Paar von diagonal durchtrennten Zylindern, je einen für die Positionsmessung horizontal und vertikal. Durchfliegt ein Teilchenbunch den Strahlpositionsmonitor, induziert er durch seine Ladung ein Spannungssignal in den Zylinderhälften. Die Summe der Signale beider Zylinderhälften ist porportional zur Strahlintensität. Durch die Rechts–Links–Asymmetrie der diagonal geteilten Zylinderelektroden wird außerdem die Differenz der Spannungssignale proportional zu der Abweichung des Teilchenbunches von der Sollachse. Damit sind Strahlintensität und Strahlposition gemessen.

Desweiteren sind die Meßdaten der Strahlpositionsmonitore zum EDDA–Host zu übertragen. Das Grundprinzip ist, daß die Strahlpositionsmonitore von COSY–Control so eingestellt werden, daß sie ihre Daten an die Netzwerkadresse des EDDA–Hosts schicken. Dabei legen die Daten den Weg zurück, der in Abb. 4.3 gezeigt ist. Die von der Hardware der BPM's ermittelten Daten werden zunächst auf das COSY–VME–Target übertragen. Dort werden die Daten durch Mittelung über eine Sequenz von Einzelmessungen komprimiert und dann direkt an den EDDA–Host geschickt. Der Datentransport erfolgt über ein Ethernetkabel mit UDP/IP (Internet User Datagram Protokol/Internet Protokoll). UDP/IP ist ein Protokoll (Software–Mechanismus) zur Übertragung von Datenpaketen ohne Aufbau einer stehenden Verbindung zwischen Absender und Empfänger. Um die Daten entgegen zu nehmen, muß auf dem EDDA–Host eine „Empfangsstelle“ für solche Datenpakete eingerichtet werden. Dies ist ein ständig im Hintergrund laufender Prozeß, der auflaufende Datenpakete entgegennimmt und an den Eventbuilder weiterschickt (s. Anhang E). Das Weiterschicken erfolgt über TCP/IP (Internet Transmission Control Protocol). TCP/IP nutzt hier die gleiche Hardware (Ethernet) wie UDP/IP. Der wesentliche Unterschied zwischen beiden Protokollen ist, daß TCP/IP eine stehende, bidirektionale Verbindung aufbaut (entsprechend einem Telefongespräch), während UDP/IP nur Datenpakete verschicken kann (entsprechend dem Verschicken von Briefen mit der Post). Auf dem Eventbuilder werden die Daten dann entgegengenommen, in den allgemeinen Datenstrom integriert und mit auf Band geschrieben. Damit stehen die Strahldiagnosedaten zur Berücksichtigung bei der „offline“–Analyse zur Verfügung.

4.3 Erste Tests

Die erstellte Software zur Aufnahme etwa der BPM–Daten wurde zuerst innerhalb des Computernetzwerkes des ISKP getestet. Dazu wurden die BPM's durch einen einfachen Sender ersetzt, der Textzeilen über UDP/IP verschickt. Dieser Aufbau enthielt auch schon das Weiterleiten der Daten an den Eventbuilder und auf das Datenband.

Soweit vorbereitet, kam es in der dritten EDDA–Strahlzeit vom 6. – 10. Dezember 1993 zu einer ersten Erprobung am laufenden Teilchenbeschleuniger COSY. Dabei gelang es auf Anhieb, einzelne Datensätze von den BPM's bis auf den Eventbuilder zu übertragen. Inhaltliche Details sowie realistischere Betriebsmodi der BPM's stehen noch aus, dürften aber keine prinzipiellen Probleme mehr aufwerfen.

Kapitel 5

Ergebnisse

Zur Realisierung von Steuerung und Überwachung der Hochspannungsversorgung, Diskriminatoren und Delays im EDDA–Experiment ist die benötigte Software erstellt worden. Sie ist in Strahlzeiten der Experimente EDDA und MOMO erprobt worden (s. Abschnitt 3.4) und soll in beiden Experimenten weiterhin benutzt werden. Die Anbindung an COSY–Control, die bereits von D. Theis vorbereitet worden war, wurde realisiert (s. Abschnitt 4.3). Neben der Bereitstellung der erforderlichen Funktionalität ist auch immer auf Allgemeinheit und Modularität der einzelnen Programmteile geachtet worden. So konnte die Schnittstelle, welche ursprünglich für das Schreiben der Steuerprogramme über TDAS auf Band entwickelt wurde, dann auch unverändert für die Aufgabe verwandt werden, Diagnosedaten von COSY–Control auf Band zu dokumentieren. Benutzerschnittstelle und Funktionalität der Steuerprogramme sind getrennt und damit unabhängig voneinander änderbar. Dies ermöglicht es z.B., eine komfortablere Benutzerschnittstelle auf dem leistungsfähigeren Host zu erstellen, ohne sich mit dem Steuerprogramm selbst auseinandersetzen zu müssen.

Anhang A

HV–Steuerung

Dieser Anhang erläutert einige technische Aspekte des Steuerprogramms der HV näher. Im einzelnen werden das Steuermenü, die Syntax der benutzten Textdateien sowie der Programmablauf anhand eines Beispiels dargelegt.

A.1 Das HV Untermenü

Das HV–Untermenü in TDASmenu ist die zentrale Steuerstelle für die Hochspannungsversorgung. Es sieht folgendermaßen aus:

```
**********************************************************************
*                         TDAShvcontrol menu                         *
**********************************************************************

G: get HV values, write to text file
T: get HV values, write to text file and Tape
S: set HV from text file
K: shutdown HV using text file 'hvkill'
E: edit text file
D: show directory of text files
Q: terminate TDAShvcontrol process
P: restart TDAShvcontrol process
H: help function of TDAShvcontrol
R: return to main menu

**********************************************************************
```

Die einzelnen Menüpunkte arbeiten wie folgt:

G: get... Mit diesem Menüpunkt kann der Ist–Zustand der Hochspannungsversorgung gelesen werden. Nach Auswahl dieses Kommandos wird zunächst nach einer Kanalauswahldatei (s. Anhang A.2) und dann nach einer Ergebnisdatei zur Aufnahme des Resultats gefragt. Hieraufhin wird eine eventuell laufende Datenaufnahme unterbrochen (HV–Steuerung und Datenaufnahme schließen einander aus), der Ist–Zustand der Hochspannungskanäle ausgelesen und in der spezifizierten Ergebnisdatei abgelegt. Falls die Datenaufnahme unterbrochen wurde, wird sie automatisch wieder aufgenommen. Gestalt und Inhalt der Ergebnisdatei sind in Anhang A.3 beschrieben.

T: get...and tape Dieses Kommando gleicht dem vorhergehenden, nur wird die Ergebnisdatei zusätzlich mit auf das Datenband geschrieben. Voraussetzung dafür ist, daß das Bandlaufwerk auch initialisiert ist.

S: set... Set dient der Konfiguration der Hochspannungsversorgung. Es muß eine Konfigurationsdatei angegeben werden, die alle benötigten Parameter enthält (s. Anhang A.2). Wenn kein Fehler auftritt, werden diese Parameter in die Hardware der Hochspannungsversorgung übertragen und sofort aktiv. Zur Überprüfung wird die Anzahl der gesetzten Kanäle angezeigt.

Bei Set wird gegebenenfalls die laufende Datenaufnahme vorübergehend unterbrochen, ebenso wie bei get... . In der Regel wird es jedoch nicht sinnvoll sein, die Parameter der Hochspannungsversorgung während der Datenaufnahme zu verändern.

K: shutdown... Dieser Menüpunkt ist eine Abkürzung für Set und Angabe der Konfigurationdatei „hvkill“ (der Buchstabe „K“ soll an „**K**ill“ erinnern). „hvkill“ enthält einen möglichst großen „ramp down“–Wert (Geschwindigkeit, mit der die Hochspannung heruntergefahren wird) sowie die Stellung „off“. Damit ist eine Möglichkeit geschaffen, die Hochspannungsversorgung schnell auszuschalten, auch falls die HV–Hardware nicht zugänglich ist.

E: edit... Mit edit wird ein Editor aufgerufen, mit dem die Konfigurationsdateien erstellt bzw. modifiziert und die Ergebnisdateien gesichtet werden können.

D: show directory... „Show directory“ zeigt die Liste aller Konfigurations– und Ergebnisdateien an.

Q: terminate... Terminate wird für debugging–Zwecke benutzt. Es beendet den befehlsausführenden Prozeß.

P: restart... Restart wird auch für debugging–Zwecke benutzt. Es startet den befehlsausführenden Prozeß TDAShvcontrol erneut.

H: help... Help ist eine Standardfunktion innerhalb von TDASmenu und gibt eine kommentierte Liste aller Befehle aus.

R: return... Return kehrt zum Hauptmenü von TDASmenu zurück.

A.2 Konfigurationdateien

Die Konfigurationsdateien enthalten in tabellarischer Form alle die Parameter, die zur Einstellung einer Konfiguration der Hochspannungsversorgung notwendig sind. Ihre Syntax soll an folgendem Beispiel erläutert werden:

```
                    *****  HV configuration  *****

#include "slot.h"

    Cr.   Ch.    V0      I0      V1      I1    RUP    RDW   TRIP   ON/OFF
--------------------------------------------------------------------------
>>    0 |   14 |  1200 |   240 |     - |     - |  100 |  100 |     0 |  off

Dieser Kanal ist eingeschaltet:
>>    0 |    7 |  1300 |   250 |     - |     - |   10 |  100 |     0 |   on
```

Bevor das HV–Steuerprogramm selbst diese Datei liest, wird zunächst der Preprozessor der Programmiersprache C aufgerufen [mic84]. Dieser führt alle mit einem '#' beginnenden Direktiven aus. In diesem Fall ist dies ein „#include“ Befehl, der veranlaßt, daß der Inhalt der Datei `slot.h` an dieser Stelle eingesetzt wird. `slot.h` könnte folgendermaßen aussehen:

```
#ifndef SLOT_H
#define SLOT_H

CAMAC crate no.         >> 15
HV controller slot no.  >>  7

#endif
```

`slot.h` enthält im wesentlichen die Information, in welchem Einschub welchen CAMAC–Überrahmens sich der HV–Controller befindet. Die übrigen Preprozessor–Direktiven dienen nur dazu, zu garantieren, daß der Inhalt von `slot.h` höchstens einmal eingesetzt wird. Es ist nicht verboten, die Informationen von slot.h direkt in die Konfigurationsdatei zu schreiben. Das gezeigte Vorgehen hat aber große Vorteile, wenn man verschiedene Konfigurationdateien benutzt, die auf denselben HV–Controller zugreifen. Dann müssen nämlich alle diese Dateien dieselbe Information über die Position des HV–Controllers beinhalten, was umständliches Ändern und die Gefahr von Inkonsistenzen in sich birgt. Der „include“–Mechanismus zentralisiert die mehrfach gebrauchte Information und vermeidet dadurch obige Probleme.

Hat der Preprozessor seine Arbeit getan, wird die bearbeitete Datei vom eigentlichen HV–Steuerprogramm gelesen. Datensätze werden durch ein vorangehendes „>>“ gekennzeichnet, der Rest wird als Kommentar interpretiert. Als erstes werden die Nummern des CAMAC–Crates und des Einschubs, den der HV–Controller belegt, erwartet. Darauf folgen die Parameter für die einzelnen HV–Kanäle. Die Abkürzungen bedeuten im einzelnen:

cr.	Nummer des HV–Überrahmens („crate“)
ch.	Kanalnummer („channel no.“)
V0	Sollspannung für VSEL = 0 (**V**oltage **SEL**ected, default = 0)
I0	maximal zulässiger Strom für ISEL = 0
V1	Sollspannung für VSEL = 1
I1	maximal zulässiger Strom für ISEL = 1
RUP	ramp up, Geschwindigkeit, mit der die Hochspannung hochgefahren wird, in Volt/Sekunde
RDW	ramp down, Geschwindigkeit, mit der die Hochspannung heruntergefahren wird, in Volt/Sekunde
TRIP	maximal zulässige Überschreitungsdauer des eingestellten Stromlimits in zehntel Sekunden
ON/OFF	Kanal ist ein–/ausgeschaltet

Ein „-“ bedeutet, daß dieser Wert unverändert bleibt. Die Einheit, in der die Parameter angegeben werden, hängen von den verwendeten HV–Einschüben ab. Siehe [caeα91, caeβ91] und [dieα94] für weitere Datailinformationen.

A.3 Ergebnisdateien

Das Resultat eines get–Kommandos ist eine Ergebnisdatei, die den Ist–Zustand der Hochspannungsversorgung in folgender Form wiedergibt:

```
CAMAC crate no. >> 15  HV controller slot no. >> 7

  CR CH VMON IMON   V0   I0  V1   I1  RUP RDW TRIP  OF TR ON OV UV OC RU RD
>> 0  7 1299    0 1300  250   0 2000  10 100    0   0  0  1  0  0  0  0  0
>> 0 14    0    0 1200  240   0 2000 100 100    0   1  0  0  0  0  0  0  0
```

Einige der Abkürzungen sind bereits oben erklärt worden, die übrigen bedeuten:

VMON	gemessene Spannung
IMON	gemessener Strom
OF	ausgeschaltet („off")
TR	aufgrund einer Fehlerbedingung ausgeschaltet („tripped")
ON	eingeschaltet („on")
OV	Überschreitung der Sollspannung („overvoltage")
UV	Unterschreitung der Sollspannung („undervoltage")
OC	Überschreitung des programmierten Stromlimits („overcurrent")
RU	Hochspannung wird hochgefahren („ramp up")
RD	Hochspannung wird heruntergefahren („ramp down")

Die Ergebnisdateien enthalten alle Informationen über die HV–Kanäle, die mittels Software erhalten werden können.

A.4 Kanalauswahldateien

Kanalauswahldateien geben bei Leseoperationen (z.B. „get") an, welche Hochspannungskanäle überhaupt ausgelesen werden sollen. Einziger syntaktischer Unterschied zu den Konfigurationsdateien ist, daß nur die ersten beiden Spalten der Parametertabelle angegeben zu werden brauchen. In der Praxis wird man aber dort, wo Kanalauswahldateien gebraucht werden, immer Konfigurationsdateien angeben. Diese enthalten ja gerade die Kanäle, die benutzt werden und damit zum Auslesen in Frage kommen. Die überschüssigen Parameter werden einfach ignoriert.

A.5 Beispiel für den Programmablauf

Dieses Beispiel soll die Abläufe innerhalb des HV–Steuerprogramms illustrieren. Betrachtet wird der Befehl „Tape", der den tatsächlichen Zustand der Hochspannungsversorgung ausliest und auf dem Datenband ablegt.

Nach Aufruf des Kommandos wird zunächst nach einer Kanalauswahldatei gefragt (s. Anhang A.4). Diese ist zuvor mit einem aus dem Untermenü aufrufbaren Editor modifiziert bzw. erstellt worden. Auf diese Weise wird definiert, von welchen Kanälen der Ist–Zustand ausgelesen und auf Band geschrieben werden soll. Nach Angabe einer Ergebnisdatei, die den Ist–Zustand aufnehmen soll, wird ein Befehl der Form

Ergebnispipe TAPE Kanalauswahldatei Ergebnisdatei an den ausführenden Prozeß TDAShvcontrol geschickt. Pipe bedeutet dabei einen FIFO–Buffer (**f**irst **i**n **f**irst **o**ut), der wie eine sequentielle Datei angesprochen wird. Der Befehl selbst wird auch in eine Pipe geschrieben, die mit dem Standard–Input vom Prozeß TDAShvcontrol verbunden ist. Dieser interpretiert den Befehl und liest die Kanalauswahldatei ein. Die darin spezifizierten Kanäle werden mit CAMAC–Befehlen ausgelesen. Zuvor wird jedoch ggf. die Datenaufnahme vorübergehend unterbrochen, da CAMAC–Zugriffe von TDAShvcontrol und der eigentlichen Datenaufnahme nicht parallel durchgeführt werden können. Der ausgelesene Ist–Zustand wird im Textformat in der Ergebnisdatei abgelegt. Bis hierhin unterscheidet sich der Befehl „Tape“ noch nicht vom Befehl „Get“. „Tape“ überträgt jedoch zusätzlich die Ergebnisdatei zum Prozeß „TDASacquire“ und damit letztendlich auf Band.

Eine Besonderheit des Befehls „Tape“ ist, daß er optional automatisch am Anfang eines jeden Datenaufnahmezyklus' ausgeführt werden kann. In diesem Fall wird der gleiche Befehl nicht direkt vom Benutzer, sondern implizit vom Befehl „BegRun“ (begin run, Datenaufnahmezyklus starten) mittels Interrupt aufgerufen.

Anhang B

Diskriminatorsteuerung

Eine Erläuterung einiger technische Aspekte des Steuerprogramms für die Diskriminatoren gibt dieser Anhang. Im einzelnen werden das Steuermenü und die Syntax der benutzten Textdateien dargelegt.

B.1 Das Diskriminator Untermenü

Das Untermenü zum Setzen der Diskrimiatorschwellen in TDASmenu sieht folgendermaßen aus:

```
**********************************************************************
*                     TDASthresholdcontrol menu                      *
**********************************************************************

G: get thresholds, select slots via text file
S: set threshold from text file
T: get thresholds, write to text file and tape
E: edit text file
D: show directory of text files
Q: terminate TDASthresholdcontrol process
P: restart TDASthresholdcontrol process
H: help function of TDASthresholdcontrol
R: return to main menu

**********************************************************************
```

Nun folgt eine Beschreibung der einzelnen Kommandos.

G: get... Dieses Kommando liest den eingestellten Sollzustand der Diskriminatoren aus. Das Resultat hat also nichts mit einer Überwachung der Schwellenspannung zu tun, es wird lediglich der zuletzt eingestellte Wert angegeben. Nach Auswahl dieses Kommandos wird zunächst nach einer Kanalauswahldatei und dann nach einer Ergebnisdatei zur Aufnahme des Resultats gefragt. Gestalt und Inhalt der Ergebnisdatei ist in Anhang B.3 beschrieben.

T: get...and tape Dieses Kommando gleicht dem vorhergehenden, nur wird die Ergebnisdatei zusätzlich mit auf das Datenband geschrieben. Voraussetzung dafür ist, daß das Bandlaufwerk zum Schreiben vorbereitet ist.

S: set... Set dient der Einstellung der Diskriminatorschwellen. Es muß eine Konfigurationsdatei angegeben werden, die die Schwellenspannungen enthält (s. Anhang B.2). Tritt kein Fehler auf, werden die Schwellen in den Diskriminatoren gesetzt und sofort aktiv. Zur Überprüfung wird die Anzahl der eingestellten Diskriminatoren angezeigt.

Die übrigen Menüpunkte entsprechen denen des HV–Untermenüs, sie sind unter Anhang A.1 beschrieben.

B.2 Konfigurationdateien

Die Konfigurationsdateien enthalten in tabellarischer Form alle die Parameter, die zur Einstellung der Diskriminatoren notwendig sind. Ihre Syntax zeigt folgendes Beispiel:

```
                   ***  discriminator setting  ***

Note: threshold ==   -1: switch to local mode
      threshold == 1024: set manual threshold value

CAMAC crate   slot         mask register 15...0       threshold[mV]
-------------------------------------------------------------------
>>  15         16     0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0    15
>>  15         18     0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0    15
```

Datensätze sind durch „>>" bezeichnet, alles übrige wird als Kommentar interpretiert. „CAMAC crate" und „slot" spezifizieren die Einschubposition des Diskriminators. Mit dem „mask register" können einzelne Kanäle maskiert, d.h. totgeschaltet werden, wobei „0" einen Kanal aktiviert und „1" einen Kanal totschaltet. „Threshold" meint die (negative) Schwellenspannung des Diskriminators, angegeben in mV.

B.3 Ergebnisdateien

Die verwendeten Diskriminatoren erlauben es nur, die zuletzt eingestellten Parameter noch einmal zurückzulesen, ein Messen der tatsächlich aktiven Schwelle ist nicht möglich. Entsprechend enthalten die Ergebnisdateien den zuletzt eingestellten Sollzustand:

```
              read discriminator setting

CAMAC crate   slot       mask 15...0                 threshold
--------------------------------------------------------------
     15        16    0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0     15
     15        18    0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0     15
```

Eine detailiertere Beschreibung der Diskriminatorsteuerung gibt [dieß94].

Anhang C

Steuerung der Delays

Im folgenden soll das Untermenü der Delay–Steuerung sowie die benutzten Konfigurationsdateien beschrieben werden.

C.1 Das Delays Untermenü

Das Untermenü der Delay–Steuerung, das der Einstellung der programmierbaren Delays dient, umfaßt folgende Punkte:

```
*****************************************************************
*                    TDASdelaycontrol menu                      *
*****************************************************************

T: set DELAY values, write to Tape as well
S: set DELAY from text file
E: edit text file
D: show directory of text files
Q: terminate TDASdelaycontrol process
P: restart TDASdelaycontrol process
H: help function of TDASdelaycontrol
R: return to main menu

*****************************************************************
```

Die einzelnen Menüpunkte leisten dabei folgendes:

T: set..., write to Tape as well Mit diesem Befehl werden die Verzögerungszeiten der einzelnen Kanäle programmiert. Die dazu notwendigen Daten werden aus der anzugebenden Konfigurationdatei entnommen, die anschließend mit auf das Datenband geschrieben wird. Ein Auslesen der Verzögerungskanäle läßt die Hardware der Delays nicht zu, deshalb wird der eingestellte Zustand dokumentiert.

S: set... Set entspricht dem „Tape"–Kommando, allerdings wird die Konfigurationsdatei nicht mit auf das Datenband geschrieben.

Die übrigen Menüpunkte entsprechen denen des HV–Untermenüs, sie sind im Anhang A.1 beschrieben.

C.2 Konfigurationsdateien

Die Konfigurationsdateien enthalten in tabellarischer Form alle Parameter, die zur Einstellung der Delays notwendig sind. Die Syntax wird durch folgendes Beispiel gezeigt:

```
                        ***  DELAY CONFIGURATION FILE  ***

CAMAC crate   slot no.   channel   delay
----------------------------------------
>>    15          17           0        0
>>    15          17           5        4
>>    15          17          12        8
```

Die einzustellenden Parameter werden durch „>>" eingeleitet, der Rest wird als Kommentar interpretiert. „CAMAC crate" und „slot no." bezeichnen die Position des Delayeinschubs, „channel" selektiert den Kanal innerhalb des Moduls, und „delay" gibt die Verzögerung an. Der Zahlenwert der Verzögerung liegt im Bereich von 0 bis 15, die Einheit ist je nach Untermodell der Delays 1 ns, 2 ns oder 4 ns. Darüberhinaus beschreibt [dieγ94] die Delaysteuerung.

Anhang D

Das Programm HVstatus

HVstatus ist ein Programm, das die Hochspannungsversorgung überwacht. Die Syntax sowie die Ausgabe dieses Programms werden im folgenden beschrieben.

Der Aufruf erfolgt z.Z. aus der Kommandoebene des Betriebssystems, aber eine Integration in die Menüsteuerung erfolgt in naher Zukunft. Die Syntax ist:

hvstatus [Optionen]

Als Optionen sind möglich:

–f Konfdatei	gibt die Konfigurationsdatei an, aus dem die aktuelle Einschubposition des HV–Controllers sowie die zu überwachenden HV–Überrahmen entnommen werden. Fehlt diese Option, wird TDAScnfpath/hvdefault.hvc verwendet.
–l logfile	gibt ein anderes Logfile als TDASroot/hv.log an.
–m	keine Fehlermeldung auf das Datenband schreiben.
–q	unterdrückt die Ausgabe des Status der HV auf das Terminal (quiet). Das ist sinnvoll, wenn der Bildschirm bereits durch den TDASmonitor belegt ist.
–t time	gibt die Zeit zwischen zwei Überprüfungen an.

Nach dem Start wird im eingestellten Zeitabstand der Status aller in der Konfigurationsdatei auftauchenden HV–Überrahmen abgefragt. Falls die Option „-q“ nicht angegeben wird, erfolgt eine Ausgabe der Form

```
crate  PowOn  Passw  Keyb   Alarm   Vx     Ix    HV_ON
  0      1      0      1      0      0      0      1
```

auf den Bildschirm. Dabei bedeuten:

crate	Nummer des HV–Überrahmens
PowOn	1 = eingeschaltet
Passw	0 = kein Passwort
Keyb	1 = manuelle Einstellung am Gerät möglich
Alarm	1 = Fehlerbedingung wurde detektiert
Vx	V0 oder V1 selektiert
Ix	I0 oder I1 selektiert
HV_ON	0 = HV ist gesperrt

Im Falle von Alarm = 1 wird eine zusätzlich Warnung ausgegeben.

Vergleichbare Einträge erfolgen im Falle eines Fehlers im Logfile der HV–Steuerung, in der Fehler–Logdatei von TDAS (sichtbar im TDASmonitor) und auf dem Datenband. Detailiertere Information über den aufgetretenen Fehler kann nur durch Auslesen der HV mit der „get“ Menüfunktion erhalten werden. Das liegt in der Verantwortung des Benutzers, da es mit einer vorübergehenden Unterbrechung der Datenaufnahme verbunden ist.

Anhang E

Die Verbindung HOST–Eventbuilder

Man möchte noch während der Datenaufnahme eine Kontrolle darüber haben, ob auch wirklich sinnvolle Daten aufgenommen werden. Dies erfolgt über ein online–Datenauswerteprogramm, das auf dem Host arbeitet. Dazu müssen aber die Daten vom Eventbuilder über Netzwerk auf den Host übertragen werden. Dies geschieht über Ethernet und TCP/IP. Für den genannten Zweck genügte eine unidirektionale Verbindung vom Eventbuilder zum Host. Durch die Einbeziehung der Strahldiagnosedaten von COSY–Control in die Datenaufnahme mußte eine leistungsfähigere Verbindung (auf Softwareebene) entwickelt werden. Diese erlaubt es jetzt Programmen auf dem Host, sich zu einem bestimmten Prozeß auf dem Eventbuilder zu verbinden. Das sind z.B. der Prozeß „TDASaccessdc" zum Übertragen von Daten zum Host oder der Prozeß „TDASccserverdc" zum Entgegennehmen der Strahldiagnosedaten von COSY–Control. Diese Verbindungen werden auf dem Eventbuilder vom Prozeß „TDASaccessd", der immer vorhanden ist, vermittelt.

Die Aufnahme einer Verbindung wird vom Host initiiert. Dazu nutzt das Programm auf dem Host, das eine Verbindung aufbauen möchte, die dazu erstellte Funktion „connect_tdas_common". Diese Funktion stellt zunächst eine Verbindung zum Vermittler „TDASaccessd" her und gibt diesem an, welcher Prozeß auf dem Eventbuilder die Verbindung entgegen nehmen soll. Auf dem Eventbuilder ruft der „TDASaccessd" dann das gewünschte Programm auf und übergibt diesem die Verbindung zur Funktion „connect_tdas_common" auf dem Host. Jede weitere Kommunikation findet dann unmittelbar zwischen dem Programm auf dem Host, das „connect_tdas_common" aufgerufen hat, und dem Zielprozeß auf dem Eventbuilder, z.B. dem Prozeß „TDASaccessdc" („TDASaccessd connect"), statt. Das Protokoll dieser Kommunikation ist dann individuell festgelegt.

Literaturverzeichnis

[ack93] K. Ackerstaff et al., Nucl. Instr and Meth. A 335 (1993) 113–120

[bis90] J. Bisplinghoff und F. Hinterberger, Proc. Conf. on Particle Production near Threshold, AIP Conf. Proc. 221 (Nashville, 1990), Am. Inst. of Phys. 1990, p. 312

[bis93] J. Bisplinghoff et al., Nucl. Instr. and Meth. A 329 (1993) 151

[caeα91] C.A.E.N.: Model SY 127 High Voltage System, Technical Information Manual, C.A.E.N. 1991

[caeβ91] C.A.E.N.: Model SY 127 High Voltage System CAMAC Controller, Technical Information Manual, C.A.E.N. 1991

[dieα94] O. Diehl: TDASHVcontrol User's Guide, ISKP der Universität Bonn, 1993

[dieβ94] O. Diehl: TDASthresholdcontrol User's Guide, in Vorbereitung, ISKP der Universität Bonn, 1994

[dieγ94] O. Diehl: TDASdelaycontrol User's Guide, in Vorbereitung, ISKP der Universität Bonn, 1994

[gro93] R. Groß–Hardt: Simulation des EDDA–Detektors — erste Ergebnisse, Diplomarbeit, ISKP der Universität Bonn, 1993

[jah89] R. Jahn: A precision study of near threshold two meson production via the reaction $p+d \rightarrow ^3He + \pi^+\pi^-$ and $p+d \rightarrow ^3He + K^+K^-$, proposal, ISKP der Universität Bonn, 1989

[jah91] R. Jahn, Near threshold two meson production in hadronic fusion reactions, in Particles And Fields Serias 41 (Particle Production Near Threshold, Nashville 1990), American Institute of Physics, New York, 1991

[lah92] U. Lahr: EDDA — Ein Detektor zur Messung von Anregungsfunktionen der elastischen Proton–Proton Streuung an COSY, Dissertation, ISKP der Universität Bonn, Juli 1992

[lec90] LeCroy Corporation: 1990 Research Instrumentation Catalog, LeCroy Corporation 1990

[mic84] Microware Systems Corporation: OS-9/68000 Language Manuals, Microware 1984

[schα94] F. Schwandt: TDAS OS–9 ISKP Programmer's Guide, ISKP der Universität Bonn, 1994

[schβ94] H. Scheid: Dissertation, in Vorbereitung, ISKP der Universität Bonn, 1994

[sco93] W. Scobel et al., Status of the EDDA–Experiment at COSY, Physika Scripta Vol. 48 S. 92–94, 1993

Danksagung

An dieser Stelle möchte ich allen danken, die zu dieser Arbeit beigetragen haben, insbesondere

Prof. Dr. R. Maschuw, der mir die Durchführung dieser Arbeit am Institut für Strahlen– und Kernphysik ermöglicht hat,

Prof. Dr. J. Bisplinghoff, unter dessen Anleitung diese Arbeit entstanden ist, für seine ständige Unterstützung und sein Engagement für EDDA,

Dipl. Phys. F. Schwandt für seine fachliche Unterstützung, zahlreichen Anregungen und die ständige Bereitschaft, Fragen zu beantworten,

Dipl. Phys. C. Henrich für seine ständige Diskussionsbereitschaft und vielfachen Vorschläge,

Dipl. Phys. H. Scheid für seine Unterstützung als System–Verwalter der E6–Rechner,

Dr. D. Theis, der entscheidende Vorarbeit für die Anbindung an COSY–Control geleistet hat,

den Gruppenmitgliedern Dipl. Phys. F. Mosel, Dipl. Phys M. Schulz–Rojahn und A. Heine für die gute Zusammenarbeit sowie allen Mitgliedern der EDDA–Kollaboration.

Besonderer Dank gebührt meinen Eltern und Großeltern, die mein Studium der Physik erst ermöglicht haben.