# Entwicklung und Test der dritten H1-Triggerstufe

**Development and Testing of the Third Level Trigger for H1**

Vom Fachbereich Physik
der Universität Dortmund
zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften
genehmigte

**Dissertation**

von

Diplom-Physiker Jürgen Naumann
aus Aachen

Dortmund

Dezember 2002

**Kurzfassung**

Um nach Umbau des HERA-Speicherrings von der um den Faktor fünf erhöhten Luminosität profitieren zu können, indem bei einer effizienten Unterdrückung des Untergrunds gleichzeitig eine hohe Selektivität bei der Datenaufzeichung exklusiver Endzustände ermöglicht wird, baut die H1-Kollaborationen einen schnellen, auf Spurkammerinformationen basierenden Trigger.

Dieser Fast Track Trigger (FTT) gliedert sich in die ersten drei Stufen des bestehenden H1-Triggersystems ein. Zunächst werden für ausgewählte Drähte der zentralen Spurkammer Spursegmente in vier radialen Triggerlagen identifiziert, um diese in der zweiten Stufe zu vollständigen Spuren zusammenzufügen. Eine nachfolgende Spurparameteranpassung liefert die dreidimensionalen Eingangsdaten für die als FTT-L3 bezeichneten dritte Stufe des Fast Track Triggers.

Aufgabe von FTT-L3 ist die Identifikation exklusiver Endzustände innerhalb von etwa 100 $\mu$s. Im Rahmen der vorliegenden Arbeit wurde ein Konzept für FTT-L3 entwickelt und in einem Testaufbau umgesetzt. Als Plattform für Selektionsalgorithmen dient eine Farm von Prozessorkarten, die in einem VME-System integriert sind. Neben der Programmierung dieser Prozessorkarten wurde ein Konzept für den schnellen Datentransfer innerhalb des Systems und zu verschiedenen Schnittstellen entwickelt. Durch Testmessungen konnte gezeigt werden, daß das beschriebene System in der Lage ist, die Spezifikationen zu erfüllen.

Weiterhin wurde ein modulares Datenauslesesystem für den gesamten FTT entwickelt und in Betrieb genommen. Damit ist es möglich, Daten des FTT über das zentrale H1-Datennahmesystem aufzuzeichnen.

**Abstract**

To benefit from the increase in luminosity by a factor of five anticipated after the upgrade of the HERA storage ring, the H1 collaboration is building a Fast Track Trigger (FTT). It's aim is to provide a high selectivity for exclusive final states accompanied by an efficient background reduction.

The FTT fits into the first three levels of the existing H1 trigger system. The first level identifies track segments within each of the four radial trigger layers formed by selected signal wires of the central H1 drift chambers. The segments are taken as input data for the second level where they are combined to complete tracks. A subsequent track parameter fit provides full three-dimensional track information which are passed to the third level FTT-L3.

FTT-L3 should be able to identify exclusive final states within 100 $\mu$s. Within the scope of this theses a concept for FTT-L3 has been developed and a test set-up has been built. The selection algorithms are implemented on a farm of processor boards integrated in a VME system. In addition to the programming of these boards several data paths for high speed data transmission within the system and to interfaces have been set up. Test measurements have revealed that the system described in this theses is able to fulfill the given specifications.

In addition a modular data read-out system for the FTT has been developed which is described in this theses. After commissioning it has become possible to pass FTT data to the central H1 data acquisition.

# Inhaltsverzeichnis

# Kapitel 1

# Einleitung

Wie häufig kann man einen Apfel teilen, ohne daß die Eigenschaft “Apfel” verloren geht? Aufgrund solcher philosophischer Überlegungen begründeten Leukipp von Milet und sein Schüler Demokrit im fünften vorchristlichen Jahrhundert den Atomismus, die Lehre, daß alles Seiende aus kleinsten, selbst unteilbaren Teilchen aufgebaut ist. Diese Teilchen nannten sie nach dem griechischen Wort für “unteilbar” Atom. Weitere Einzelheiten der vorchristlichen Sicht der Natur sind zum Beispiel in [Luk-60 v.Chr.] beschrieben.

Es dauerte allerdings bis etwa 1860, bis auf einer Größenskala von $10^{-10}$ m Teilchen entdeckt wurden, die der Definition von Leukipp und Demokrit sehr nahe kamen: Diese Atome sind die kleinste Einheit, aus denen ein Element aufgebaut ist und sie sind der Träger der chemischen Elementeigenschaften. Zusätzlich gelang es 1869 D. I. Mendelejew und J. L. Meyer unabhängig voneinander eine Periodizität in den Eigenschaften von Elementen zu finden, die letztlich zur Aufstellung des Periodensystems der Elemente führte. Das Auffinden solch einer Ordnung in der Naturbeschreibung gibt naturwissenschaftlich interessierten Menschen das Gefühl, eine richtige Beschreibung und somit ein tieferes Verständnis für die sie umgebende Natur entwickelt zu haben. Eine wirkliche Erklärung für die Elementeigenschaften blieb allerdings noch aus und wurde erst gefunden, als sich das Atom als nicht atomar herausstellte.

Geiger ließ am Anfang des letzten Jahrhunderts $\alpha$-Teilchen an einer sehr dünnen Goldfolie streuen und beobachtete eine unerwartete Winkelverteilung der gestreuten Teilchen, insbesondere, daß sehr viele Teilchen die Folie unbeeinflußt durchqueren konnten [Gei-1909]. Rutherford konnte die Resultate quantitativ unter der Annahme beschreiben, daß die $\alpha$-Teilchen an Objekten mit einer Ausdehnung von maximal $10^{-14}$ m im Mittelpunkt eines Atoms streuen, wohingegen das restliche Volumen eines Atoms annähernd leer ist [Rut-1911]. Er interpretierte diese Objekte als kleine, positiv geladenen Kerne, die fast die gesamte Masse eines Atoms in sich vereinigen. Sie sind von einer Hülle aus Elektronen umgeben, die ein Atom nach außen hin elektrisch neutral erscheinen läßt und dessen chemische Eigenschaften bestimmt.

Während die Elektronen bis heute als unteilbar angesehen werden, gelang es noch zwei Mal, weitere Unterstrukturen in Atomkernen zu finden. Zunächst wurden das Proton [Nut-1913] und später auch das Neutron [Cha-1932a, Cha-1932b] entdeckt und als die Bausteine eines Atomkerns identifiziert. Damit war es möglich, den Aufbau der Natur mit nur wenigen Teilchen zu beschreiben. In der Mitte der zwanzigsten Jahrhunderts wurden allerdings zahlreiche weitere Teilchen mit Massen in der Größenordnung der des Protons oder Neutrons, aber verschiedenen Eigenschaften bezüglich ihrer Streuprozesse mit Materie entdeckt, die unter dem Begriff Hadronen[1] zusammengefaßt wurden.

[1] griechisch adro = groß, stark

| | Quarks | | Leptonen | |
|---|---|---|---|---|
| 1. Familie | $d$<br>down | $u$<br>up | $e$<br>Elektron | $\nu_e$<br>Elektronneutrino |
| 2. Familie | $s$<br>strange | $c$<br>charm | $\mu$<br>Myon | $\nu_\mu$<br>Myonneutrino |
| 3. Familie | $b$<br>bottom/beauty | $t$<br>top | $\tau$<br>Tau | $\nu_\tau$<br>Tauneutrino |

*Tabelle 1.1: Übersicht über die als elementar angenommenen Teilchen mit denen der Aufbau der Materie vollständig beschreibbar ist. Die QCD ordnet den Quarks einen weiteren Freiheitsgrad, die Farbladung zu, so daß es sich eigentlich jeweils um drei verschiedene Teilchen mit den Farbladungen* rot, grün *und* blau *handelt. Jedem Teilchen ist desweiteren ein Antiteilchen mit entgegengesetzter Ladung bei ansonsten gleichen Eigenschaften zugeordnet. Die Eigenschaften aller Teilchen sind in [Hag-2002] beschrieben.*

Ein regelrechter “Teilchenzoo” führte zu theoretischen Bemühungen, auch die Hadronen als aus wenigen fundamentalen Teilchen zusammengesetzt zu beschreiben. Gell-Mann und Zweig postulierten 1964 eine neue Teilchenart, die Quarks genannt wurden [Gel-1964, Zwe-1964] und die Anfang der 70er Jahre am SLAC[2]-Teilchenbeschleuniger in Streuexperiment nachgewiesen werden konnten [Bre-1969, Blo-1969]. Sowohl die theoretischen Vorarbeiten als auch der experimentelle Nachweis wurden mit Nobelpreisen ausgezeichnet. Nach heutigem Verständnis ist die gesamte Materie aus einer Reihe von Elementarteilchen mit halbzahligem Spin, sogenannten Fermionen, aufgebaut. Sie unterteilen sich in Leptonen[3], zu denen zum Beispiel das eine ganzzahlige Ladung tragende Elektron und die ladungsneutralen Neutrinos gehören, und in Quarks mit drittelzahliger elektrischer Ladung, aus denen die Hadronen aufgebaut sind. Zu jedem Teilchen existiert zusätzlich ein Antiteilchen, das bei sonst gleichen Eigenschaften die entgegengesetzte Ladung trägt. Tabelle 1.1 gibt einen Überblick über alle bekannten Elementarteilchen mit der üblichen, paarweisen Einordnung in sogenannte Familien. Sie stellt im übertragenen Sinne das Periodensystem der Elementarteilchenphysik dar.

Neben den Bausteinen der Natur wurde auch deren Wechselwirkung miteinander untersucht. Bis heute sind vier Wechselwirkungen bekannt:

- elektromagnetische Wechselwirkung
- schwache Wechselwirkung
- starke Wechselwirkung
- Gravitation.

Durch die elektromagnetische Wechselwirkung wird das Verhalten von Teilchen aufgrund ihrer Ladung beschrieben. Die schwache Wechselwirkung beschreibt radioaktive

[2] **S**tanford **L**inear **Ac**celerator

[3] griechisch lepto = zart, dünn

Prozesse und läßt sich mit der elektromagnetischen Wechselwirkung für höhere Energien zu einer gemeinsamen, der elektroschwachen Wechselwirkung zusammenfassen. Mit Hilfe der starken Wechselwirkung lassen sich Vorgänge innerhalb der Hadronen beschreiben. Diese Wechselwirkungen lassen sich im Rahmen von Quantenfeldtheorien beschreiben, die jeder Wechselwirkung ein oder mehrere Eichbosonen, also Austauschteilchen mit Spin 1, zuordnen. Für die elektroschwache Wechselwirkung sind dies Photon, $Z^0$, und $W^{\pm}$, für die starke Wechselwirkung im Rahmen der QCD[4] die Gluonen. Die Menge der bisher beschriebenen Theorien der Teilchen und ihrer Wechselwirkungen wird als Standardmodell der Elementarteilchenphysik bezeichnet. Die Gravitation dominiert auf großen Längenskalen und spielt in der Teilchenphysik keine Rolle. Sie läßt sich bisher auch nicht im Rahmen einer Quantenfeldtheorie formulieren.

In der Tradition Geigers und Rutherfords werden auch heute noch Streuexperimente betrieben und entwickelt, um Strukturen und mögliche Unterstrukturen der Materie zu untersuchen. Aufgrund der Heisenberg'schen Unschärferelation [Hei-1927] müssen im Schwerpunktssystem der beteiligten Stoßpartner immer höhere Energien und Impuls erreicht werden, um immer kleinere Strukturen auflösen zu können. Daher werden heutzutage umfangreiche Anlagen betrieben, in denen Teilchenstrahlen zunächst auf hohe Energien beschleunigt und anschließend mit einem weiteren Strahl oder einem festen Target zur Kollision gebracht werden. Eine dieser Beschleunigeranlagen befindet sich am DESY[5] in Hamburg mit seinem großen Elektron-Proton-Speicherring HERA[6].

## 1.1 HERA–Speicherring

Der HERA–Speicherring [Dre-1984] ist entwickelt worden, um Strukturen im Proton zu untersuchen. In Abbildung 1.1 ist der 6,3 km umfassende HERA-Ring sowie das komplexe System aus Vorbeschleunigern schematisch dargestellt. Protonen werden in Paketen mit mehreren Milliarden Teilchen in den Ring injiziert und auf eine Energie von 920 GeV beschleunigt. Als Sonde zur Untersuchung dienen wahlweise Elektronen oder Positronen[7], die ebenfalls in Paketen mit einer Energie von 27,6 GeV in einem separaten Strahlrohr in entgegengesetzte Richtung im HERA-Tunnel umlaufen. An zwei Wechselwirkungspunkten werden die Teilchenpakete mit einer Rate von 10,4 MHz zur Kollision gebracht und die Reaktionsprodukte durch den H1-Detektor in der nördlichen und durch den ZEUS-Detektor in der südlichen Experimentierhalle vermessen. An den anderen beiden Experimentierpunkten nutzt zum einen das HERA-B-Experiment den Protonenstrahl und zum anderen das HERMES-Experiment den Elektronenstrahl, um diese jeweils mit einem festen Target in Wechselwirkung zu bringen. Am Beispiel des H1-Detektors wird in Kapitel 2 der typische Aufbau eines Experimentes an kollidierenden Strahlen beschrieben. Die aus den Strahlenergien resultierende Schwerpunktsenergie von etwa 300 GeV erlaubt es nach der Heisenbergschen Unschärferelation, Strukturen von $10^{-18}$ m aufzulösen. Eine Auswahl damit möglicher Messungen wird nach Einführung einiger Begriffe in den folgenden Abschnitten beschrieben.

---

4 **Q**uanten**c**hromo**d**ynamik
5 **D**eutsches **E**lektronen–**SY**nchrotron
6 **H**adron–**E**lektron–**R**ing**A**nlage
7 Im folgenden wird dieser Tatsache nicht Rechnung getragen und generisch von Elektronen gesprochen

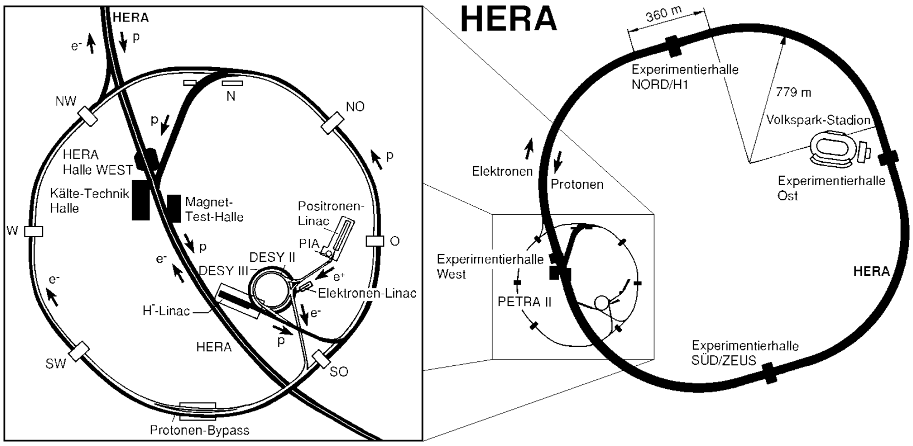



*Abbildung 1.1: Schematische Ansicht des HERA–Speicherrings (rechts) und dem zugehörigen System von Vorbeschleunigern (vergrößert links)*

### 1.1.1 Messung der Protonstruktur

Die Wechselwirkung zwischen dem einlaufenden Elektron und einem Quark des Protons kann durch den Austausch eines Trägers der elektroschwachen Wechselwirkung beschrieben werden. Bei den bei HERA erreichbaren Schwerpunktsenergien ist der Austausch von $W$- und $Z$-Bosonen in weiten Bereichen des zugänglichen, kinematischen Phasenraums aufgrund ihrer hohen Masse stark unterdrückt, so daß im folgenden nur der Photonaustausch betrachtet werden soll. Die Abbildungen 1.2 und 1.3 zeigen verschiedene Prozesse der durch ein Photon übertragenen Elektron-Proton-Wechselwirkung. Die Variablen $\mathbf{k}$ und $\mathbf{P}$ bezeichnen darin die Viererimpulse des einlaufenden Elektrons und Protons, $\mathbf{k}'$ entsprechend den des auslaufenden Elektrons. Das Proton kann durch die Wechselwirkung entweder in einen hadronischen Endzustand $X$ aufgebrochen werden oder aber nach einer elastischen Streuung intakt bleiben. In beiden Fällen soll im folgenden der Viererimpuls dieses auslaufenden Systems durch $\mathbf{P}'$ bezeichnet werden. Die Wechselwirkung selbst kann als Kopplung des ausgetauschten Photons an ein Quark verstanden werden, wie es in Abbildung 1.2a gezeigt ist. Das ausgetauschte Photon kann entweder direkt an ein Quark (Abbildung 1.2a), oder indirekt durch den zwischengeschalteten Austausch eines Quarks in einem sogenannten Boson-Gluon-Fusionsprozeß (Abbildung 1.2b) an ein Gluon koppeln. Letzteres wird im Rahmen der QCD als Prozess höherer Ordnung bezeichnet und weiter unten beschrieben werden.

Zur Beschreibung der Ereigniskinematik werden vier lorenzinvariante Größen verwendet, die unter der bei HERA zulässigen Vernachlässigung der Elektron- und Protonmassen durch die Relation

$$Q^2 = sxy \tag{1.1}$$

verknüpft sind. Mit $s$ wird das Quadrat der Schwerpunktsenergie des Elektron-Proton-Systems bezeichnet, die bei HERA seit 1998 $\sqrt{s} \approx 318$ GeV beträgt. Das Quadrat des in Abbildung 1.2 als $\mathbf{q}$ bezeichneten Viererimpulses des ausgetauschten Photons ist bei

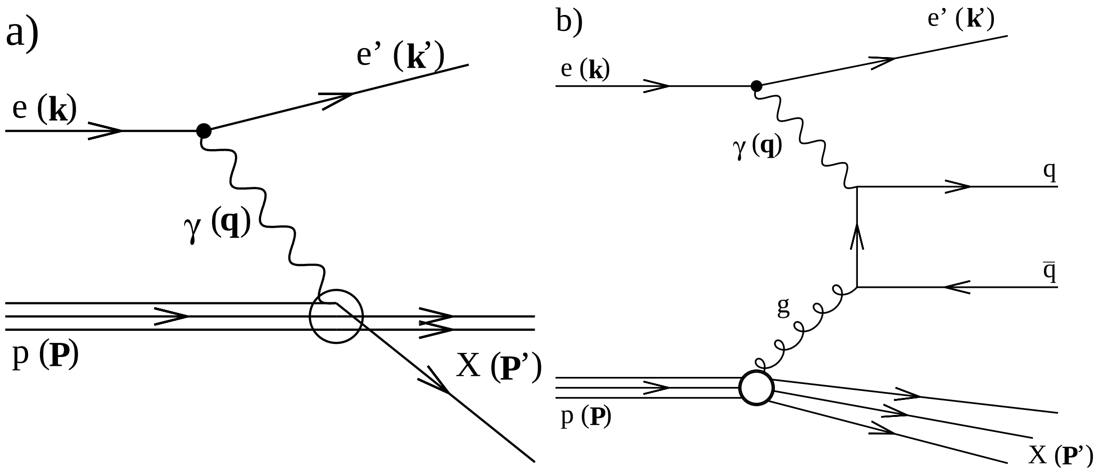



*Abbildung 1.2: Feynmandiagramme von Elektron-Proton-Wechselwirkungsprozessen. Abbildung a) zeigt das direkte Ankoppeln eines Photons $\gamma$ an ein Quark im Proton, Abbildung b) einen sogenannten Boson-Gluon-Fusionsprozeß, bei der eine Quarkschleife erzeugt wird. In Klammern ist die Bezeichnung des jeweiligen Viererimpulses angegeben.*

HERA stets negativ, so daß hier stattdessen

$$Q^2 = -\mathbf{q}^2 = -\left(\mathbf{k} - \mathbf{k}'\right)^2 = 4 \cdot E_e E'_e \cdot \sin^2\left(\frac{\theta_{e'}}{2}\right) \tag{1.2}$$

verwendet wird. Die Virtualität $Q^2$ läßt sich aus der Energie des einlaufenden ($E_e$) und auslaufenden ($E_{e'}$) Elektrons, sowie dessen Streuwinkel $\theta_{e'}$ berechnen und als Auflösungsvermögen des Photons für Strukturen im Proton interpretieren. Für kleine $Q^2$ kann das Photon als "quasi-reell" angesehen werden, so daß bei Ereignissen mit $Q^2 \leq 1\ \mathrm{GeV}^2$ von *Photoproduktionsereignissen* gesprochen wird. Die anderenfalls vorliegenden *Elektroproduktionsereignisse* werden auch als tiefinelastische Streuung, DIS[8], bezeichnet. Weiterhin werden die Bjorken-Skalenvariablen

$$x = \frac{Q^2}{2 \cdot \mathbf{P} \cdot \mathbf{q}} \tag{1.3}$$

und die Inelastizität

$$y = \frac{\mathbf{q} \cdot \mathbf{P}}{\mathbf{k} \cdot \mathbf{P}} \tag{1.4}$$

verwendet. Im naiven Quark-Parton-Modell [Bjo-1969, Fey-1969] kann $x$ als Impulsbruchteil des an der Wechselwirkung beteiligten Partons bezogen auf den Gesamtimpuls des Protons interpretiert werden. Die Inelastizität $y$ gibt im Ruhesystem des Protons den relativen Energieverlust des Elektrons an. Die Variablen $x$ und $y$ können nur Werte im Bereich ]0,1] annehmen.

[8] **D**eep **I**nelastic **S**cattering

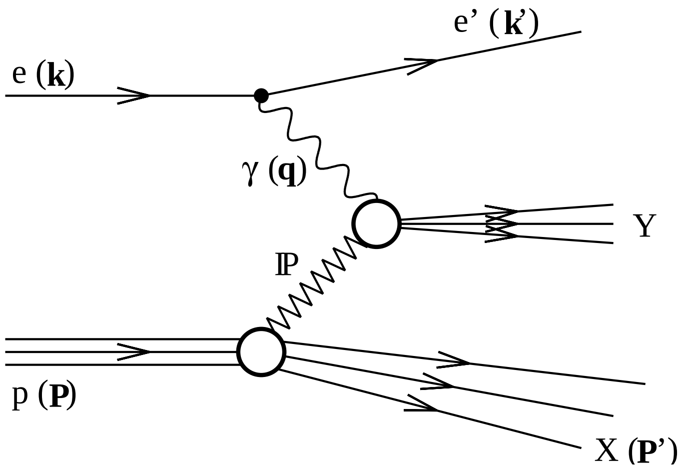



*Abbildung 1.3: Feynmandiagramm der durch ein Photon $\gamma$ und das Austauschteilchen $I\!P$ übertragenen, diffraktiven ep-Wechselwirkung. Da das $I\!P$ keine Farbladung trägt, werden zwischen dem Protonrest $X$ und den Zerfallsprodukten $Y$ keine weiteren Teilchen erzeugt.*

Bei den bisher beschriebenen Prozessen verläßt ein Farbladung tragendes Objekt das Proton, das mit dem verbleibenden, farbgeladenem Protonrest ein Farbdipolfeld ausbildet. Da es Eigenschaft der dieses beschreibenden QCD ist, daß der Energieinhalt des Feldes mit steigendem Abstand der Ladungsträger zunimmt, können Energieschwellen zur Teilchenpaarerzeugung überschritten und aus Quarkpaaren weitere Hadronen im Rapiditätsbereich[9] zwischen Protonrest und gestreutem Teilchen erzeugt werden. Bei dieser sogenannten Hadronisierung gehen auch die Reaktionsprodukte in farbneutrale Objekte über.

Als dritte Ereignisklasse soll hier die in Abbildung 1.3 gezeigte, sogenannte diffraktive Streuung erwähnt werden, bei der das Photon über ein die Quantenzahlen des Vakuums tragendes und damit farbneutrales Objekt $I\!P$ mit dem Proton wechselwirkt. Da hierbei kein Farbdipolfeld erzeugt wird, ist diese Ereignisklasse durch einen unterdrückten Teilchenfluß im Rapiditätsbereich zwischen Protonrest $X$ und Zerfallsprodukten $Y$ der Photon-$I\!P$-Wechselwirkung charakterisiert. Zur Beschreibung der Ereigniskinematik bedient man sich hier zusätzlich der Mandelstam-Variable $t$, die das Quadrat des Viererimpulsübertrags bezeichnet, der von dem einlaufenden Proton auf den erzeugten Endzustand $Y$ übergeht. Sie ist definiert als

$$t = \left(\mathbf{P} - \mathbf{P}'\right)^2. \tag{1.5}$$

#### 1.1.1.1 Protonstrukturfunktion $F_2$

Bei einer festen Schwerpunktsenergie $\sqrt{s}$ reicht die Bestimmung von zwei der drei verbleibenden Größen in Gleichung 1.1 zur vollständigen Beschreibung der Ereigniskinematik der untersuchten Prozesse aus. Als quantitative Größe wird typischerweise der

[9] Ein Maß ist die Pseudorapidität $\eta$, die über $\eta = -\frac{1}{2}\ln(\tan\frac{\theta}{2})$ mit dem Polarwinkel verknüpft ist.

Wirkungsquerschnitt $\sigma$ angegeben, der ein Maß für die Anzahl beobachteter Ereignisse in einem bestimmten kinematischen Bereich ist. Für kleine Werte von $Q^2$, bei denen nur der Photonaustausch beträgt, läßt sich zum Beispiel der differentielle Wirkungsquerschnitt $d^2\sigma/dxdQ^2$-Bereich schreiben als

$$\frac{d^2\sigma(x,Q^2)}{dxdQ^2} = \frac{4\pi\alpha_{em}^2}{xQ^4}\left[\left(1-y+\frac{y^2}{2}\right)F_2(x,Q^2) - \frac{y^2}{2}F_L(x,Q^2)\right], \tag{1.6}$$

wobei $\alpha_{em}$ die elektromagnetische Kopplungskonstante und $F_2$ und $F_L$ Funktionen zur Beschreibung der inneren Struktur des Protons -sogenannte Strukturfunktionen- sind. Für kleine Werte von $y$ kann der Beitrag durch $F_L$ allerdings vernachlässigt werden. Auf die Bedeutung von $F_2$ soll im Folgenden näher eingegangen werden.

Gemäß dem naiven Quark-Parton-Modell sollte das Proton aus drei Quarks, zwei up-Quarks und einem down-Quark bestehen. Geht man davon aus, daß der Protonimpuls sich ausschließlich und paritätisch aus den Impulsanteilen der Quarks zusammensetzt, wäre $F_2$ nur bei $x = 1/3$ ungleich Null. Da ein durch höheres $Q^2$ gesteigertes Auflösungsvermögen keine weiteren Informationen brächte, bestünde darüberhinaus keine Abhängigkeit von diesem. Es zeigte sich jedoch, daß durch die Wechselwirkung der Quarks untereinander auch Impulse übertragen werden. Der Aufbau des Protons muß daher durch ein dynamisches Modell beschrieben werden, in dem $F_2$ von $x$ abhängt:

$$F_2(x) = \sum_{i=1}^{2} e_i^2 x q_i(x). \tag{1.7}$$

Für die ersten beiden Quarksorten wird jeweils die von $x$ abhängige Quarkdichtefunktion $q(x)$ skaliert mit $x$ und der Quarkladung $e_i$ aufsummiert. Die als “Scaling” bezeichnete Annahme, daß $F_2$ von $Q^2$ unabhängig ist, wurde bei ersten Streuexperimenten am SLAC bei $x \approx 0.2$ zunächst bestätigt [Bre-1969, Blo-1969].

Abbildung 1.4 hingegen zeigt Meßergebnisse für $F_2$ von Experimenten mit festen Targets und aus $ep$-Daten von HERA über einen großen $x$- und $Q^2$-Wertebereich. Deutlich ist zu sehen, daß Scaling nur für den am SLAC vermessenen Bereich des Partonimpulses $x$ erfüllt ist und somit zusätzliche Vorgänge im Proton bei der Beschreibung seiner Struktur berücksichtigt werden müssen. Weiterhin entspricht die Summe der Quarkimpulse nur etwa der Hälfte des Protonimpulses. Dies kann durch das naive Quark-Parton-Modell ebenfalls nicht beschrieben werden. Im Rahmen der QCD [Pol-1973, Gro-1973, Gro-1973, Gro-1974] wird die Wechselwirkung der Quarks untereinander durch den Austausch von Gluonen beschrieben. Diese können zum einen wiederum kurzzeitig in virtuelle Quark-Antiquarkpaare fluktuieren, zum anderen aber auch weitere Gluonen emittieren, da sie selber Farbladung tragen. Auf diese Weise werden neben den drei Valenzquarks weitere virtuelle, sogenannten Seequarks erzeugt, die auf kleinen Längenskalen ebenfalls zu $F_2$ einen Beitrag leisten. Da sie durch Abstrahlungsprozesse entstehen und daher nur Bruchteile des Impulses des ursprünglichen Teilchen tragen, werden sie typischerweise mit kleinen Werten von $x$ produziert und verursachen einen Anstieg von $F_2$ in diesem Bereich. Abbildung 1.5 bestätigt dieses Verhalten an Hand von Meßdaten.

Die explizite Abhängigkeit der Quarkdichtefunktionen von $Q^2$ ist durch dessen Interpretation als Auflösungsvermögen zu verstehen. Abbildung 1.6 zeigt schematisch Strukturmessungen durch ein Photon bei zwei verschiedenen Werten von $Q^2$ ($Q_1^2 < Q_2^2$). Da

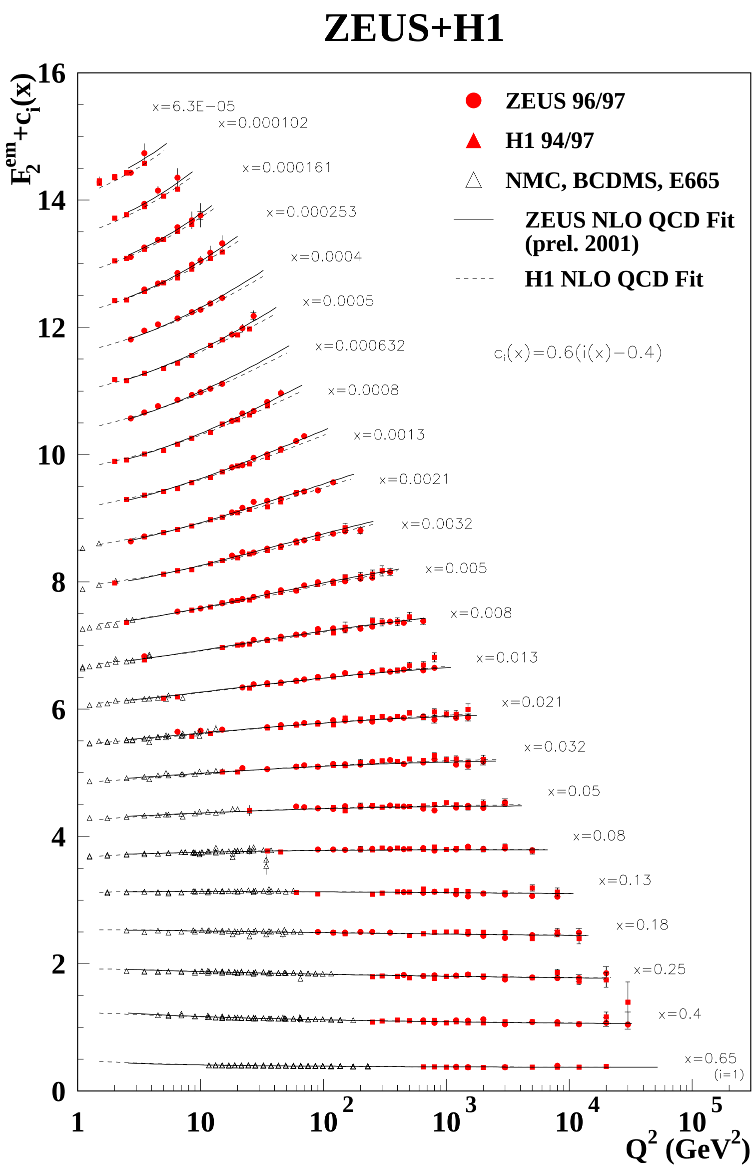



*Abbildung 1.4: Werte der Strukturfunktion $F_2(x, Q^2)$ gemessen an verschiedenen Experimenten über einen weiten kinematischen Bereich. Während durch Experimente an einem festen Target zunächst nur Bereiche mittlerer und großer x bei kleinen Werten von $Q^2$ untersucht werden konnten [Ben-1989, Ada-1996, Arn-1997], erweiterten die HERA-Experimente H1 und ZEUS die Sensitivität in Bereiche großer $Q^2$ und kleiner x [Adl-2000, Adl-2001a, Adl-2001b, Bre-2000, Che-2001]. Der Übersichtlichkeit halber sind die Meßwerte abhängig von x skaliert dargestellt. Die gezeigten Meßwertanpassungen basieren auf DGLAP-Modellen, die auch Terme höherer Ordnung beinhalten. Sie sind im kinematischen Bereich bis hinunter zu $Q^2 = 3{,}5$ GeV$^2$ sehr gut in der Lage, die Meßwerte zu beschreiben.*

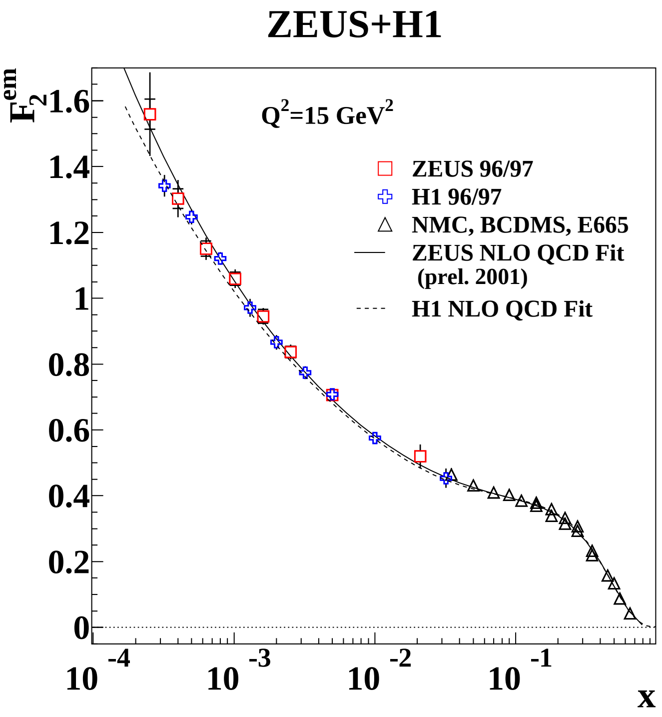



*Abbildung 1.5: Strukturfunktion $F_2$ bei einem Wert von $Q^2 = 15$ GeV$^2$ in Abhängigkeit von x. Die Datensätze sind eine Untermenge der in Abbildung 1.4 gezeigten Daten. Bemerkenswert ist der steile Anstieg bei kleinen Werten von x, der vor den HERA-Messungen nur von wenigen Modellrechnungen vorgegesagt worden war.*

die Struktur des Protons wie oben beschrieben auch kurzzeitige Fluktuationen auf kleinen Längenskalen beinhaltet und die Größe der durch das Photon auflösbaren Strukturen proportional zu $1/Q^2$ ist, zeigt sich für verschiedene Meßbereiche von $Q^2$ ein unterschiedliches Bild des Protons. Während bei kleinen Werten von $Q^2$ überwiegend die durch die Valenzquarks bestimmte Struktur gemessen wird, können mit steigendem $Q^2$ auch die durch Abstrahlungsprozesse hervorgerufenen Beiträge erfaßt werden. Mit steigendem $Q^2$ steigt die gemessene Teilchendichte im Proton, und damit auch $F_2$ an.

Um die genannte Ergebnisse zu berücksichtigen, muß die Formel 1.7 modifiziert werden. Zum einen sind die Beiträge aller Quarksorten $n_f$ und zusätzlich der Dichtefunktionen $\bar{q}$ ihrer Antiteilchen zu berücksichtigen, zum anderen diese Dichtefunktionen nun abhängig von $Q^2$ anzunehmen:

$$F_2(x, Q^2) = \sum_{i=1}^{n_f} e_i^2 x \left(q_i(x, Q^2) + \bar{q}_i(x, Q^2)\right) . \tag{1.8}$$

Die Dichtefunktionen selbst sind mit bisherigen Methoden nicht analytisch zu berechnen, sondern müssen ausgehend von gemessenen Werten störungstheoretisch entwickelt werden. Im Rahmen der QCD kommen dabei häufig von Dokshitzer, Gribov,

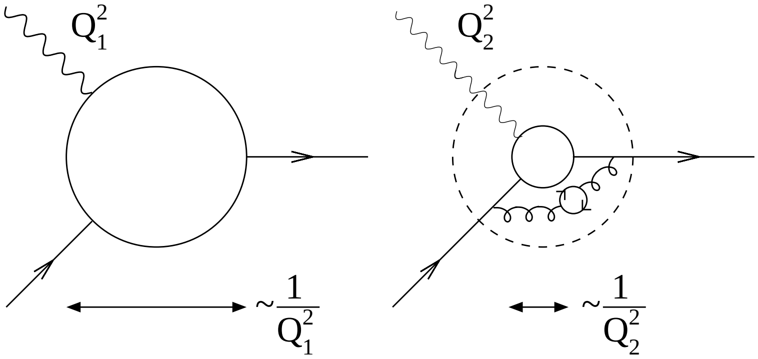



*Abbildung 1.6: Auflösungsvermögen bei der Untersuchung der Protonstruktur. Während bei einem kleinen Impulsübertrag $Q_1^2$ nur relativ große Strukturen sichtbar werden, können mit steigendem $Q^2$ ($Q_1^2 < Q_2^2$) auch Beiträge durch Abstrahlungs- und Paarbildungsprozesse auf kleinen Längenskalen aufgelöst werden. Damit steigt die effektive Partondichte mit steigendem $Q^2$ an.*

Lipatov, Altarelli und Parisi entwickelte, sogenannte DGLAP-Entwicklungsgleichungen [Dok-1977, Gri-1972, Gri-1972, Alt-1977] zum Einsatz, die in einem weiten Phasenraum die gemessenen Dichtefunktionen erfolgreich beschreiben können:

$$\begin{aligned}
\frac{dq_i(x,Q^2)}{d\ln Q^2} &= \frac{\alpha_s(Q^2)}{2\pi}\int_x^1 \frac{dz}{z}\left[\sum_j q_j(z,Q^2)\; P_{qq}\left(\frac{x}{z}\right) + g(z,Q^2)\; P_{qg}\left(\frac{x}{z}\right)\right] \\
\frac{dg(x,Q^2)}{d\ln Q^2} &= \frac{\alpha_s(Q^2)}{2\pi}\int_x^1 \frac{dz}{z}\left[\sum_j q_j(z,Q^2)\; P_{gq}\left(\frac{x}{z}\right) + g(z,Q^2)\; P_{gg}\left(\frac{x}{z}\right)\right]. \qquad (1.9)
\end{aligned}$$

Ausgehend von bekannten Verteilungen für jede Quarksorte $i$ bei einem Startpunkt $Q_0^2$ kann durch die obere Gleichung die Änderung der Quarkdichtefunktionen $q_i(x,Q^2)$ bei sich änderndem $\ln(Q^2)$ berechnet werden. Da die Funktionen auch Beiträge von Quarks aus Paarbildungsprozessen $g \to q\bar{q}$ beinhalten, ist auch die Verteilungsfunktionen für Gluonen $g(x,Q^2)$ zu berücksichtigen. Die Entwicklung dieser Funktion in Abhängigkeit von $\ln(Q^2)$ wird durch die zweite Gleichung beschrieben. Durch Gluonabstrahlungsprozesse $q \to qg$ und Paarbildungsprozesse hängen $q_i(x,Q^2)$ und $g(x,Q^2)$ gegenseitig voneinander ab. Die Kopplung der beiden Gleichungen wird mathematisch durch sogenannte Teilungsfunktionen ("Splitting functions") $P_{K'K}(\frac{x}{z})$ erreicht, die die Wahrscheinlichkeit angeben, daß ein Teilchen $K$ mit dem Impulsanteil $x$ durch Abstrahlung eines Quarks oder Gluons in ein Teilchen $K'$ mit dem Impulsanteil $z$ ($z < x$) übergeht. Können an einem gegebenen Punkt $(x, Q_0^2)$ die Partondichten erfolgreich parametrisiert werden, ist es möglich, diese durch Einsetzen in die DGLAP-Gleichung auch für andere Werte von $Q^2$ zu berechnen. Dahingegen ist es mit diesen Gleichungen nicht möglich, die Entwicklung der Partondichten in Abhängigkeit von $x$ vorherzusagen, so daß diese für feste Werte von $Q^2$ experimentell bestimmt werden muß.

Die Vorgehensweise bei der Berechnung der Partondichteverteilungen mit Hilfe der DGLAP-Gleichungen soll am Beispiel eines Ansatzes von Glück, Reya und Vogt [Glü-1991] erläutern werden. Für einen relativ niedrigen Wert von $\mu^2 = Q_0^2 = 0,45$ GeV$^2$

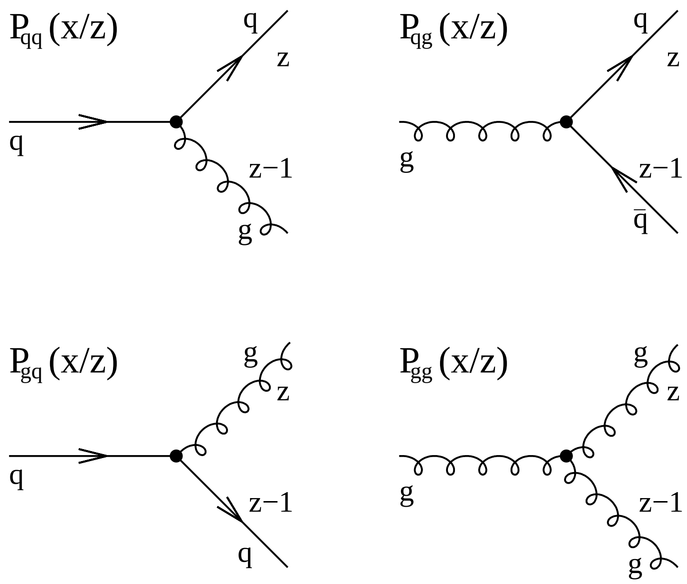



*Abbildung 1.7: Anschauliche Bedeutung der Splitting-Funktionen $P_{K'K}(x/z)$ zur Berechnung von Teilchendichten in Hadronen. Beschrieben werden Prozesse der Gluonabstrahlung $q \rightarrow qg$, die Quark-Paarproduktion $g \rightarrow q\overline{q}$ und das sogenannte "Gluon-Splitting" $g \rightarrow gg$.*

wird versucht, gemessene Partondichteverteilungen analytisch durch eine Parameteranpassung an eine geeignete Funktion zu beschreiben. Im beschriebenen Ansatz wird davon ausgegangen, daß bei dem Startpunkt $Q_0^2$ nur die Beiträge von Quarks der ersten Familie und von Gluonen berücksichtigt werden müssen. Die resultierenden Partondichteverteilungen sind in Abbildung 1.8 a) gezeigt. Abbildung 1.8 b) zeigt die daraus durch die DGLAP-Gleichung entwickelten Dichteverteilungen für $Q^2 = 5\,\mathrm{GeV}^2$ [Glü-1998]. Die Splittingfunktionen generieren nun auch einen Anteil von $s$-Quarks, der zur Struktur des Protons beiträgt. Desweiteren fällt auf, daß der Anteil der Valenzquarks zu Gunsten anderer Partonen, vor allem des Gluons, abnimmt. Diese Ergebnisse können durch eine Anpassung der Partondichteverteilungen an Meßdaten überprüft werden [Bot-2000]. Abbildung 1.8 c) zeigt die aus HERA-Daten resultierenden Partondichteverteilungen bei $Q^2 = 10\,\mathrm{GeV}^2$. Die Gluondichtefunktion $g$ und der alle Seequarks einschließende Beitrag $S$ sind um den Faktor 20 reduziert dargestellt.

Die anfängliche Extraktion von $g$ aus der Anpassung an Meßdaten ist stark sensitiv auf die dabei verwendeten Modelle. Alternativ wird daher versucht, $g$ direkt zu messen, wofür sich insbesondere Boson-Gluon-Fusionsereignisse eignen. In Abbildung 1.2b ist zu sehen, daß dabei das ausgetauschte Photon über eine Quarkschleife an ein Gluon des Protons ankoppelt. Ein gut zu detektierender Endzustand dieses Wechselwirkungsmechanismus ist die Produktion von zwei sogenannten Teilchenjets aus der Hadronisierung der beiden in verschiedene Raumwinkelbereiche produzierten Quarks. Die gute Detektierbarkeit ist allerdings nur für relativ große Transversalimpulse der Jets gegeben, so daß grade für den interessanten Bereich bei kleinen $x$ nach anderen Endzuständen zu suchen ist. Weiterhin können sich aus dem Quark-Antiquarkpaar gebundene Zustände,

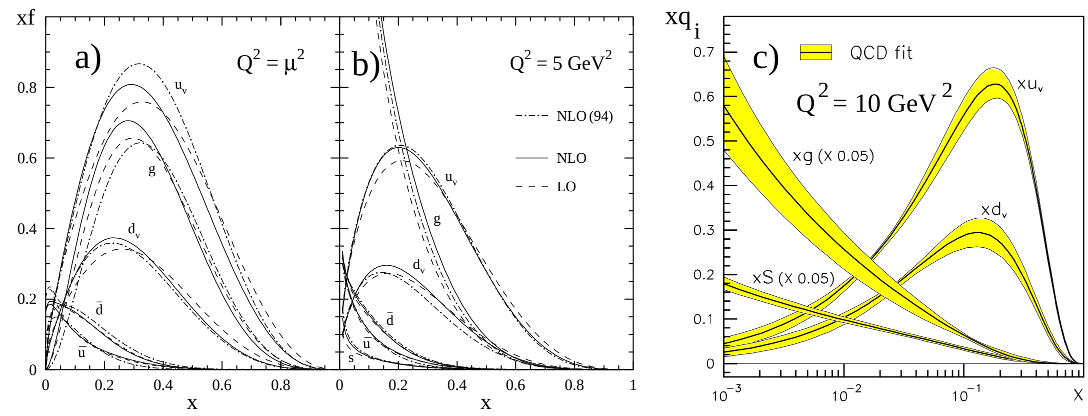



*Abbildung 1.8: Partondichteverteilungen im Proton bei verschiedenen Werten von $Q^2$. Abbildung a) zeigt die in [Glü-1991] angenommene Verteilung bei $\mu = 0{,}45$ GeV$^2$, aus der durch die DGLAP-Gleichungen die Dichteverteilungen bei höheren $Q^2$ entwickelt werden können. In Abbildung b) ist die bespielhaft für $Q^2 = 5$ GeV$^2$ gezeigt [Glü-1998]. Experimentell können die Partondichten aus einer Parameteranpassung an HERA-Meßdaten extrahiert werden (Abbildung c) [Bot-2000].*

sogenannten Mesonen, bilden. Beispielhaft genannt sei hier die direkte Produktion von $J/\Psi$-Mesonen als $c\bar{c}$-Zustand, oder aber von $D$-Mesonen, die ein einzelnes $c$-Quark in Begleitung eines Quarks der ersten Familie enthalten. Abbildung 1.9 zeigt die durch die Analyse von angeregten $D$-Mesonen am H1-Experiment gemesse Gluondichtefunktion [Adl-1999] im Vergleich mit einer Extraktion aus einer Meßdatenanpassung, sowie einer weiteren Modellrechnung [Lai-1996]. Die Aussagekraft dieser Analyse wird bisher allerdings noch durch statistische Unsicherheiten in der Größenordnung von 25% limitiert. Eine größere, verfügbare Datenmenge wäre daher wünschenswert. Weitere dann möglich werdende physikalische Untersuchungen, sind in [Ing-1996] beschrieben.

### 1.1.2 Umbau des HERA-Ringes

Die Ereignisrate $R_{Proze}$ eines bestimmten Prozesses läßt sich mit Hilfe des jeweiligen Wirkungsquerschnittes $\sigma_{Proze}$ durch

$$R_{Proze} = L \cdot \sigma_{Proze} \tag{1.10}$$

berechnen. $L$ ist dabei die Luminosität des Speicherringes:

$$L = \frac{N_p \cdot N_e \cdot \nu}{2\pi \cdot \sigma_p^2 \cdot \sigma_e^2}. \tag{1.11}$$

Die Luminosität kann durch eine Erhöhung der Teilchenzahl $N_x$ oder der Kollisionsfrequenz $\nu$, beziehungsweise eine Verringerung der Strahlquerschnitte $\sigma_x$ gesteigert werden.

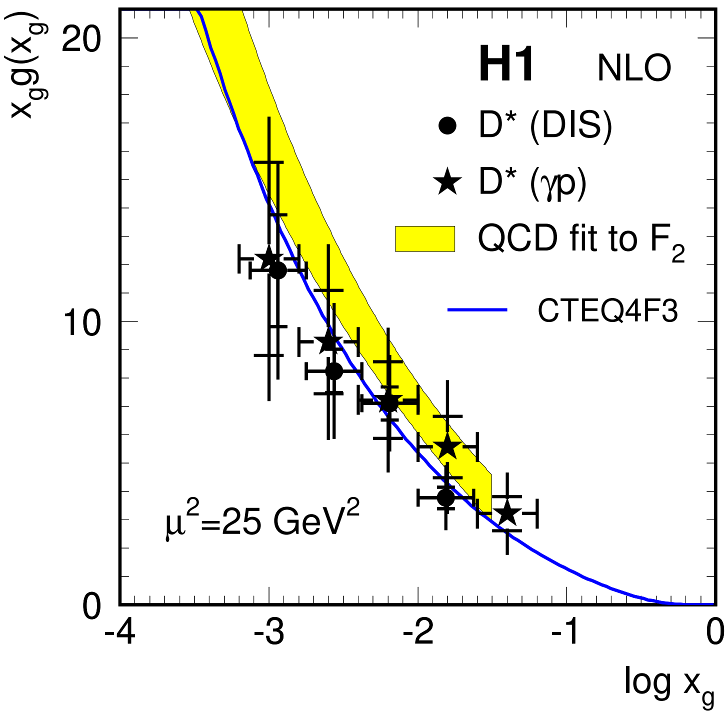



*Abbildung 1.9: Gluondichteverteilung g in Abhängigkeit von* $\log(x)$. *Die Datenpunkte wurden durch eine Analyse von in Photo- ($\gamma p$) und Elektroproduktion (DIS) erzeugten $D^*$-Mesonen [Adl-1999] ermittelt. Die inneren Fehlerbalken zeigen nur den statistischen, die äußeren die quadratisch addierten statistischen und systematischen Fehler. Zum Vergleich ist die aus der Skalenverletzung von $F_2$ ermittelte Gluondichteverteilung und die Ergebnisse einer auf den DGLAP-Gleichungen beruhende Modellrechnung [Lai-1996] eingetragen.*

Um die den Experimenten zur Analyse zu Verfügung stehende Datenmenge deutlich steigern zu können, wurde seit Ende des Jahres 2000 der HERA-Ring umgebaut [Sch-1998]. Der umgebaute Ring wird im folgenden als HERA-II bezeichnet. Eine höhere Luminosität soll zum einen durch höhere Strahlströme, also durch höhere Teilchenzahlen pro Paket erreicht werden. Technisch aufwendiger ist die zusätzliche Reduzierung der Strahlquerschnitte an den beiden Strahl-Strahl-Wechselwirkungspunkten durch Einbau supraleitender Fokussierungsmagnete. Diese mußten in die bestehenden Experimente integriert werden, um möglichst nahe an den Wechselwirkungspunkt herangebracht werden zu können. Insgesamt soll es damit möglich werden, die Luminosität im Vergleich zu HERA-I um einen Faktor fünf von $1,4 \cdot 10^{31}$ cm$^{-2}$s$^{-1}$ auf $7,6 \cdot 10^{31}$ cm$^{-2}$s$^{-1}$ zu steigern. Zusätzlich zu der in dem Zeitraum von 1993-2000 aufgezeichneten, integrierten Luminosität von etwa 120 pb$^{-1}$ pro Experiment, können dann bis 2007 etwa 1fb$^{-1}$ akkumuliert werden. Tabelle 1.1.2 zeigt die erwarteten Ereignisraten für verschiedene Bereiche von $Q^2$ bei HERA-II.

Mit der gesteigerten Luminosität werden allerdings auch die experimentellen Bedingungen schwieriger. Zum einen steigt durch die gesteigerten Strahlströme die Wahr-

| Kinematischer Bereich | Rate [Hz] |
|---|---|
| $Q^2 < 1\,\mathrm{GeV}^2$ | 1000 |
| $1 < Q^2 < 10\,\mathrm{GeV}^2$ | 40 |
| $10\,\mathrm{GeV}^2 < Q^2$ | 4 |

*Tabelle 1.2: Erwartete Ereignisraten für verschiedene Bereiche von $Q^2$ bei HERA-II [H1-1999c].*

scheinlichkeit für strahlinduzierte Untergrundereignisse. Insbesondere die Protonen können durch Wechselwirkung mit Atomen des Restgases im Strahlrohr oder der Strahlrohrwand selbst Teilchenschauer erzeugen, die in den Detektor gelangen. Andererseits steigt die Intensität der sogenannten Synchrotronstrahlung an, die von jedem transversal beschleunigten Teilchenstrahl ausgeht. Die abgestrahlte Leistung für einen aus der Teilchensorte $x$ bestehenden Strahl läßt sich aus

$$P = \frac{2}{3}\frac{e}{4\pi\epsilon_0}\left(\frac{E_x}{m_x c^2}\right)^4 \frac{I \cdot \Delta\phi}{\rho} \qquad (1.12)$$

berechnen. Durch die $1/m^4$-Abhängigkeit spielt die Synchrotronstrahlung nur für den Elektronenstrahl, nicht aber für den Protonenstrahl eine Rolle. Bei HERA-II ist von einer Erhöhung der Synchrotronstrahlungsleistung auszugehen, da neben dem höheren Strahlstrom $I$ auch die geometrischen Parameter des Strahls in der Nähe des Wechselwirkungspunkts verändert wurden. Durch die zusätzliche Fokussierung wird der Krümmungswinkel $\Delta\phi$ größer und der Krümmungsradius $\rho$ kleiner. Die Rate der durch Synchrotronstrahlung induzierten Untergrundereignisse im Detektor wird bei Erreichen des maximalen Strahlstroms voraussichtlich um mehr als einen Faktor fünf ansteigen.

## 1.2 Problemstellung und Ziel der Arbeit

Wie in den vorhergehenden Abschnitten gezeigt, geht mit dem Ausbau des HERA-Rings zwar eine Erhöhung der Rate physikalischer Ereignisse, leider aber auch des Untergrunds einher. Entsprechend höhere Anforderungen müssen die Datennahmesysteme der Detektoren erfüllen, die aus der hohen Zahl an Ereignissen im Detektor die physikalisch relevanten auszufiltern in der Lage sein müssen. Ereignisse mit einem hohen $Q^2$ können nach wie vor bei kleinen Raten mit hoher Effizienz durch die Identifikation des Elektrons selektiert werden. Für Ereignisse mit kleinem oder sogar gegen Null gehendem $Q^2$, wird die aus der ausschließlichen Elektronidentifikation resultierende Rate aber so groß, daß nur ein von den Experimentierbedingungen abhängiger Anteil der Ereignisse aufgezeichnet werden kann. Deren Auswahl erfolgt nicht durch weitere Selektionskriterien, sondern rein zufällig. Dies schränkt künstlich die statistische Genauigkeit von Analysen seltener, spezifischer Endzustände in diesem Bereich ein oder kann sie sogar unmöglich machen.

Um dies zu vermeiden, baut die H1-Kollaboration eine schnelle Auswerteelektronik, FTT[10], die es erlaubt, die gewünschten exklusiven Endzustände schon während der

[10] **F**ast **T**rack **T**rigger = Schneller Spurtrigger

Datennahme zu selektieren und vollständig der weiteren Analysekette zuzuführen. Der FTT fügt sich konzeptionell in die ersten drei Stufen des in Kapitel 2 beschriebenen H1-Triggersystems ein. Die erste Stufe FTT-L1 identifiziert Spursegmente im Detektor, die durch die zweite Stufe FTT-L2 zu vollständigen Spuren zusammengesetzt werden. Die anschließend berechneten, dreidimensionalen Spurparameter dienen als Eingabewerte der dritten Stufe FTT-L3, auf der durch auf Prozessoren implementierte Selektionsalgorithmen schließlich spezifische Endzustände identifiziert werden sollen.

Ziel dieser Arbeit war es, ein Konzept für FTT-L3 zu entwickeln und die technische Umsetzbarkeit zu überprüfen. Dazu war es notwendig, geeignete Prozessorkarten zur Bearbeitung der Selektionsalgorithmen auszuwählen und in ein Gesamtsystem zu integrieren. Dieses muß in der Lage sein, FTT-L3 Daten zuzuführen, diese lokal zu verteilen und anschließend die Ergebnisse weiterzuleiten. Desweiteren wurden Algorithmen für den Datentransfer, die Datenanalyse und die allgemeine Systemsteuerung entwickelt. Strenge Zeitvorgaben bedingen dabei den Einsatz eines Echtzeitbetriebssystems.

Im folgenden Kapitel werden zunächst die einzelnen Komponenten des H1-Experiments vorgestellt, gefolgt von einer Beschreibung des H1-Datenselektions- und Datennahmesystems. Nach einem Überblick über die Komponenten des FTT in Kapitel 3, wird in Kapitel 4 detailiert auf Konzept, Entwicklung und Arbeitsweise der Prozessorfarm eingegangen. Da Faktoren wie schnelle Datenübertragung und die Anwendung eines Echtzeitbetriebssystems auf modernen Prozessorkarten auch bei der Entwicklung eines Datennahmesystems für den gesamten FTT eine große Rolle spielen, konnte ein solches im Rahmen dieser Arbeit ebenfalls entwickelt werden. Es wird abschließend in Anhang A beschrieben.

# Kapitel 2

# H1-Detektor und Triggersystem

Zunächst soll im folgenden ein Überblick über die einzelnen Komponenten des H1-Detektors gegeben werden. Anschließend wird detailiert auf das Datennahmesystem des Experiments eingegangen. Diese gliedert sich in zwei Teile: Das Triggersystem erkennt interessante Ereignisse im Detektor und steuert den zeitlichen Ablauf des Datentransfers, für den ein Datenauslesesystem entwickelt wurde.

## 2.1 H1-Detektor

Der H1-Detektor ist um einen der nominellen Wechselwirkungspunkte der Proton- und Elektronpakete im HERA-Ring aufgebaut. Dieser Punkt definiert den Ursprung des H1-Koordinatensystems. Als $z$-Achse wird die Richtung des auslaufenden Protonenstrahls gewählt. Durch die zur Mitte des HERA-Rings zeigende $x$-Achse und die nach oben zeigende $y$-Achse wird ein rechtshändiges Koordinatensystem aufgespannt. Alternativ werden auch Polarkoordinaten verwendet. Dieses beschreibt einen Ort durch den radialen Abstand $r$ zur Strahlachse, den Azimutalwinkel $\varphi$, der ausgehend von der $x$-Achse in der $\vec{x}\vec{y}$-Ebene gemessen wird, und den Polarwinkel $\theta$, der von der $z$-Achse ausgehend den Winkel in der $\vec{r}\vec{z}$-Ebene angibt.

Der Aufbau des Detektors wird durch die Anforderung bestimmt, Eigenschaften von Teilchen wie Ladung, Energie und Impuls mit großer Genauigkeit und in einem möglichst großen Raumwinkelbereich um den Wechselwirkungspunkt zu messen. Da verschiedene Teilchenarten unterschiedliche Wechselwirkungen in Materie erfahren, kann ein geeigneter radialer Aufbau des Detektors zu einer Teilchenidentifikation beitragen. Die verschieden hohen Strahlenergien begründen dabei den asymetrischen Aufbau des H1-Detektors bezüglich der Vorwärts- und Rückwärtsrichtung.

Im folgenden werden die Komponenten des Zentraldetektors, wie sie für die Datennahme bei HERA-II eingebaut sind, genauer vorgestellt. Ein besonderes Augenmerk soll dabei auf die zentrale Spurkammer gelegt werden, da diese die Eingangssignale für den in dieser Arbeit beschriebenen Spurtrigger liefert. Abbildung 2.1 zeigt den zentralen Bereich des H1-Detektors. Für Messungen unter sehr kleinen oder grossen Streuwinkeln sind weitere Detektoren entlang der HERA-Strahlrohre angebracht, von denen hier nur ein System zur Identifikation von unter sehr kleinen Winkeln gestreuten Elektronen erwähnt werden soll. Für eine ausführlichere Beschreibung des Detektors sei auch auf [Abt-1997] verwiesen.

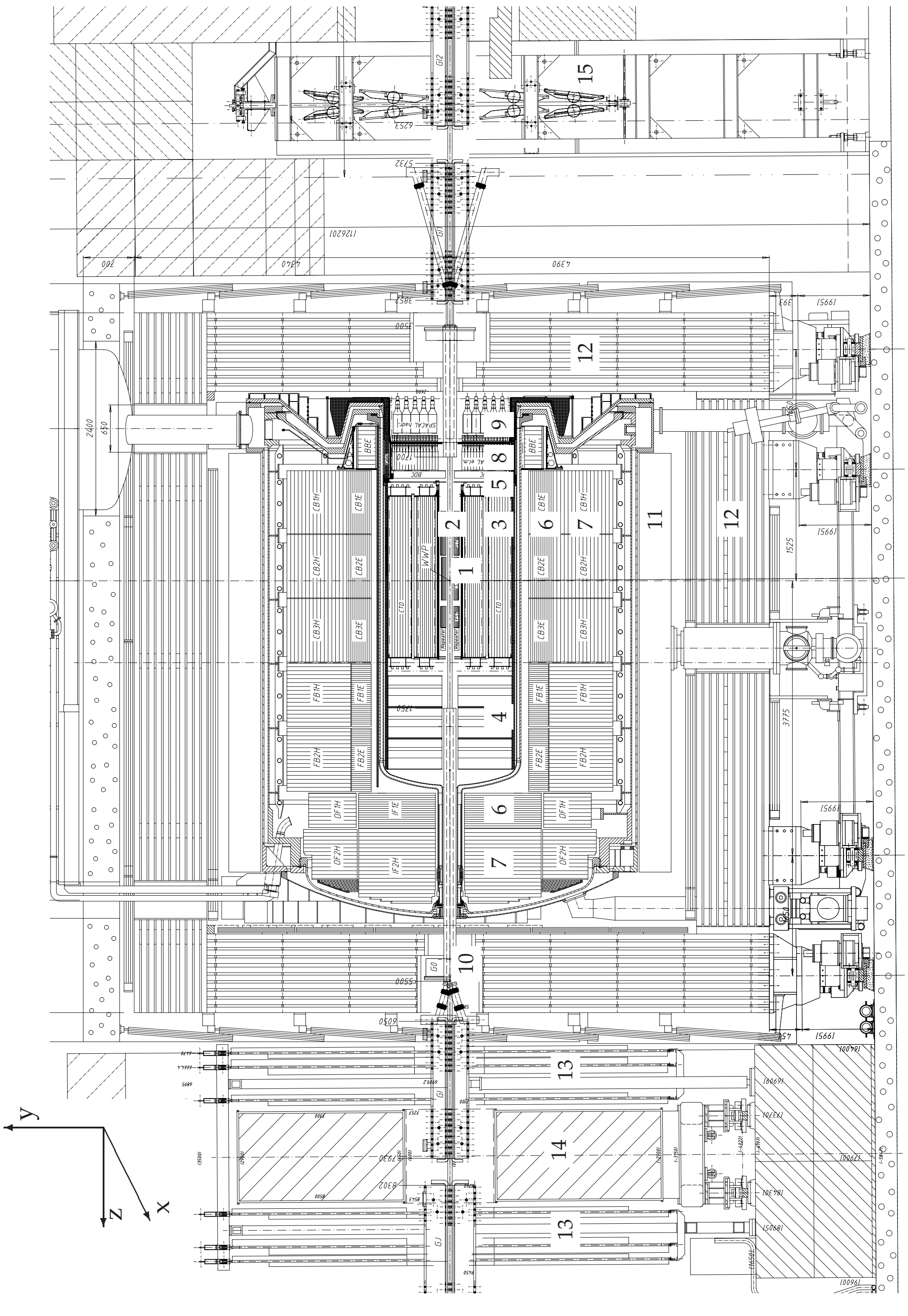



*Abbildung 2.1: Längsschnitt durch den H1-Detektor für HERA-II. [Abt-1997], aktualisiert.*

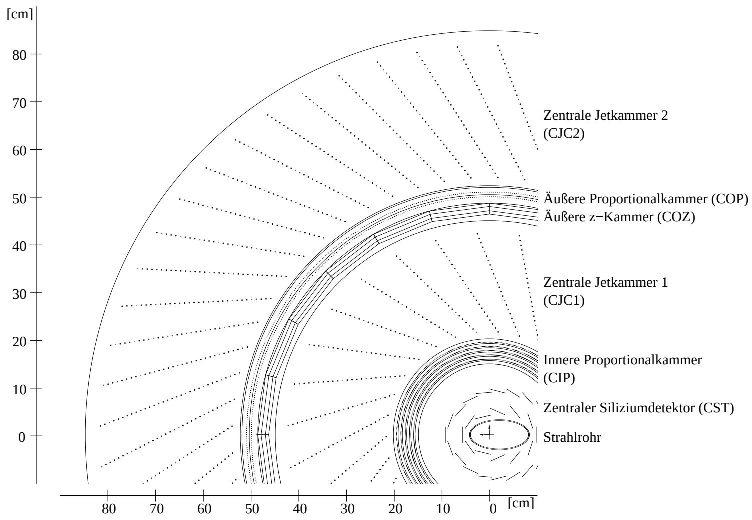



*Abbildung 2.2: Querschnitt durch das zentrale Spurdetektorsystem des H1-Detektors für HERA-II. Die Aufgaben der konzentrisch um das Strahlrohr angeordneten Kammern ist im Text beschrieben. Der Übersichtlichkeit halber sind in den Jetkammern nur die Signaldrähte eingezeichnet.*

### 2.1.1 Spurdetektoren

Um die Bahnen geladener Teilchen bestimmen zu können, sind um den Wechselwirkungspunkt $\boxed{1}$ eine Reihe von Spurdetektoren installiert. Eine supraleitenden Solenoidspule $\boxed{11}$ erzeugt in diesem Bereich ein homogenes Magnetfeld von 1,15 Tesla parallel zur Strahlrichtung. Dieses zwingt geladene Teilchen auf Kreisbahnen und ermöglicht dadurch die Bestimmung von deren Ladung und Impuls.

Direkt am Strahlrohr befinden sich mehrere Lagen von Siliziumdetektoren $\boxed{2}$. Der parallel zum Strahlrohr ausgerichtete zentrale Siliziumspurdetektor, CST[1], ermöglicht es, zwei Punkte einer Spur mit einer Genauigkeit von 12 $\mu$m in der $r$-$\varphi$-Ebene und 22 $\mu$m in der $z$-Richtung zu bestimmen [Pit-2000]. Unter einem Winkel von $162° \leq \theta \leq 176°$ in den rückwärtigen Bereich gestreute Teilchenspuren können mit dem BST[2] [Hen-2000] nachgewiesen werden, dessen Siliziumlagen senkrecht zum Strahlrohreingebaut sind. Der nahezu baugleiche FST[3] [H1-1999a] ermöglich den Nachweis von Spuren im Vorwärtsbereich von $8° \leq \theta \leq 16°$.

Im Zentralbereich von $-1,1$ m $\leq z \leq 1,1$ m umgeben zylindrische Spurdetektoren $\boxed{3}$ konzentrisch das Strahlrohr. Sie sind für drei verschiedene Aufgaben optimiert: Vermessung der Spuren in der $r$-$\varphi$-Ebene durch sogenannte Jetkammern, Bestimmung von Spurpunkten in der $z$-Richtung durch $z$-Kammern und Bereitstellung schneller Signale für das Triggersystem durch Vieldrahtproportionalkammern. Abbildung 2.2 zeigt einen Querschnitt dieses Bereichs des Detektors. Nachdem für die Datennahmeperiode HERA-I zuinnerst eine dreilagige Proportionalkammer und eine $z$-Kammer eingebaut

1 **C**entral **S**ilicon **T**racker
2 **B**ackward **S**ilicon **T**racker
3 **F**orward **S**ilicon **T**racker

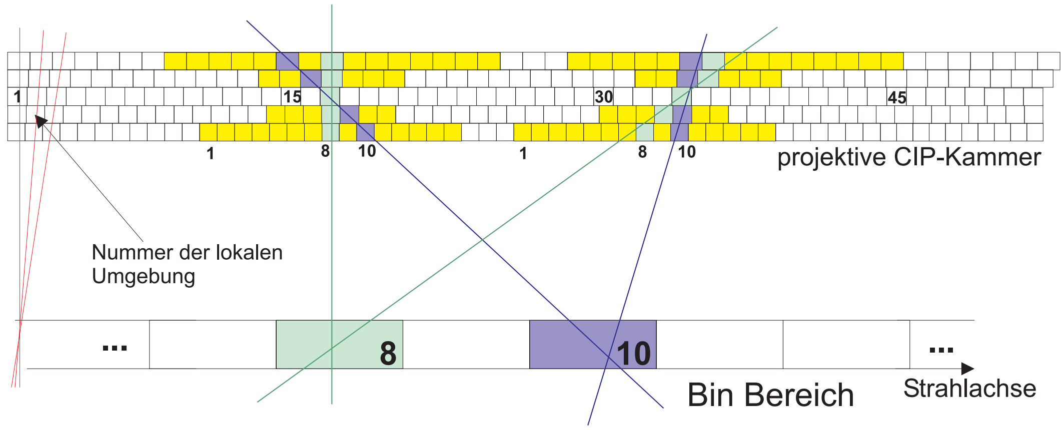



*Abbildung 2.3: Schematische Ansicht der CIP2000-Kammer mit projektiv zum Wechselwirkungspunkt (Bin 10) hin ausgerichteten Pads. Aus Treffern in verschiedenen Lagen wird versucht, eine Teilchenspur in der rz-Ebene zu rekonstruieren und diese zur Strahlachse zu extrapolieren. [Urb-2000a]*

waren, wurden diese bei HERA-II durch eine nun fünflagige Proportionalkammer mit größerer Akzeptanz in $z$-Richtung abgelöst [Cuj-1998]. Die CIP2000[4] genannte Kammer besteht radial aus 5 Lagen, die in $z$-Richtung in einzelne, sogenannte Pads unterteilt sind. Wie in Abbildung 2.3 zu sehen, wurde bei der Anordnung der Pads eine sogenannte projektive Geometrie gewählt, so daß für jeden Winkel $\theta$ die Pads zum Wechselwirkungspunkt hin ausgerichtet hintereinander liegen. Die Kammer ist nach 500 ns in der Lage, Spurkoordinaten an das assoziierte Triggersystem zu liefern [Bec-2000]. In Abschnitt 2.2.1.2 wird beschrieben, wie daraus die $z$-Koordinate des Orts der Wechselwirkung ermittelt werden kann.

Außerhalb der CIP befinden sich zwei Jetkammern. Die CJC1[5] hat eine radiale Ausdehnung von 203 mm $\leq r \leq$ 451 mm und überdeckt damit einen Winkelbereich von $22,3\,^\circ \leq \theta \leq 157,7\,^\circ$ bei voller radialer Akzeptanz. Teilchenspuren mit $10,5\,^\circ < \theta < 22,3\,^\circ$, beziehungsweise $157,7\,^\circ < \theta < 169,5\,^\circ$ durchqueren zwar zunächst die Kammer, verlassen diese aber durch ihre Endkappen und können somit nur teilweise vermessen werden. Bei 530 mm $\leq r \leq$ 844 mm befindet sich die äußere Jetkammer, CJC2, die maximal einen Winkelbereich von $25,7\,^\circ \leq \theta \leq 154,3\,^\circ$, beziehungsweise bei voller radialer Akzeptanz von $37,5\,^\circ \leq \theta \leq 142,5\,^\circ$ abdeckt. Die Kammern sind mit einem Ar-$CO_2$-$CH_4$-Gasgemisch gefüllt, das von geladenen Teilchen ionisiert wird. Parallel zum Strahlrohr gespannte, positiv geladene Signaldrähte und negativ geladene Kathodendrähte erzeugen ein elektrisches Feld, das die erzeugten Elektronen zum Signaldraht driften läßt. Da nahe des Drahts große elektrische Feldstärken herrschen, werden die Elektronen beschleunigt und können genug Energie aufnehmen, um weitere Gasatome zu ionisieren. Dabei kommt es zu einer lawinenartigen Zunahme der Zahl freier Ladungsträger, die entlang der Feldlinien driften und dadurch einen Ladungspuls auf dem Spurkammer-

[4] **C**entral **I**nner **P**roportional Chamber
[5] **C**entral **J**et **C**hamber 1

draht induzieren. Dieser Puls wird an beiden $z$-Enden der Kammer über dort eingebaute Vorverstärker ausgelesen und der weiteren Auswertung zugeführt.

Mit Hilfe eines sogenannten $Q$-$t$–Algorithmus werden Treffer auf den Signaldrähten identifiziert. Der Name leitet sich daraus ab, daß Treffer durch einen raschen Anstieg der Signalhöhe, die der auf dem Draht deponierten Ladung $Q$ proportional ist, über eine Rauschschwelle charakterisiert sind. Dem gefundenen Signal kann eine Zeitinformation $t$ mit einer Genauigkeit von 0,5 ns zugeordnet werden, aus deren Differenz zum Zeitpunkt der Wechselwirkung $t_0$ die Driftzeit der Elektronen vom Ort der Ionisation in der Kammer bis zum Signaldraht bestimmbar wird. Ist für jeden Ort der Kammer die Richtung der elektrischen und magnetischen Felder bekannt, lassen sich sogenannte Driftvektoren berechnen, denen ein freies, zu einem Signaldraht driftendes Elektron folgt. Ist zusätzlich die Driftgeschwindigkeit von Elektronen im Kammergas und die genaue Lage der Signaldrähte bekannt, kann mit Hilfe von Driftzeit $t - t_0$ und Driftvektor der Ort der Primärionisation und somit ein Punkt entlang der Teilchenspur in der $r$-$\varphi$-Ebene mit einer Genauigkeit von 170 $\mu$m rekonstruiert werden. Durch Analyse des Signalhöhenverhältnisses an den beiden Drahtenden läßt sich zusätzlich eine $z$-Information mit einer Präzision von 22 mm ermitteln.

In Abbildung 2.2 ist zu erkennen, daß die Signaldrähte in Ebenen angeordnet sind, die in der CJC1 radial aus jeweils 24 Drähten, in der CJC2 aus jeweils 32 Drähten bestehen. Durch diese werden die Kammern in 30 (CJC1), beziehungsweise 60 (CJC2) Zellen unterteilt, die jeweils von Lagen aus Kathodendrähten begrenzt werden. Deren Spannung ist abhängig von der Entfernung zum nächsten Signaldraht so eingestellt, daß ein möglichst homogenes Driftfeld in der Zelle erzeugt wird. Zusätzlich sind die Signaldrähte durch jeweils zwei geerdete Potentialdrähte gegeneinander abgeschirmt, um ein Übersprechen der Signale zu benachbarten Drähten zu reduzieren. Aus Gründen der Übersichtlichkeit sind in Abbildung 2.2 allerdings nur die Signaldrähte dargestellt. Die Zellen sind um 30° zur radialen Richtung geneigt. Dadurch wird zum einen erreicht, daß für hochenergetische Spuren mit kleiner Krümmung die Elektronen durch das Magnetfeld nahezu senkrecht zur Signaldrahtebene driften und dadurch die Ortsauflösung optimiert wird. Desweiteren können aufgrund der Neigung die Ambiguitäten besser aufgelöst werden, die entstehen, da nicht bekannt ist, auf welcher Seite des Drahtes die Primärionisation stattgefunden hat. Meist ist es nur mit einer der beiden Lösungen möglich, längere Spuren über Zellgrenzen hinweg zu rekonstruieren.

Wie in Abbildung 2.4 veranschaulicht, läßt sich eine gekrümmte Spur durch eine Helix mit fünf Parametern beschreiben:

- Spurkrümmung $\kappa$:
  $\kappa$ ist der Kehrwert des Radius der Spur, negativ für Spuren positiv geladener Teilchen und entsprechend positiv für negativ geladene Teilchen. Unter Kenntnis des Magnetfelds läßt sich der Transversalimpuls eines Teilchen leicht aus
  $$p_t = \frac{0.3 \cdot B[\text{Tesla}]}{|\kappa|[1/\text{cm}]} \tag{2.1}$$
  bestimmen.

- Abstand $d_{ca}$:
  Der Abstand zwischen dem Ursprung des H1-Koordinatensystems und dem

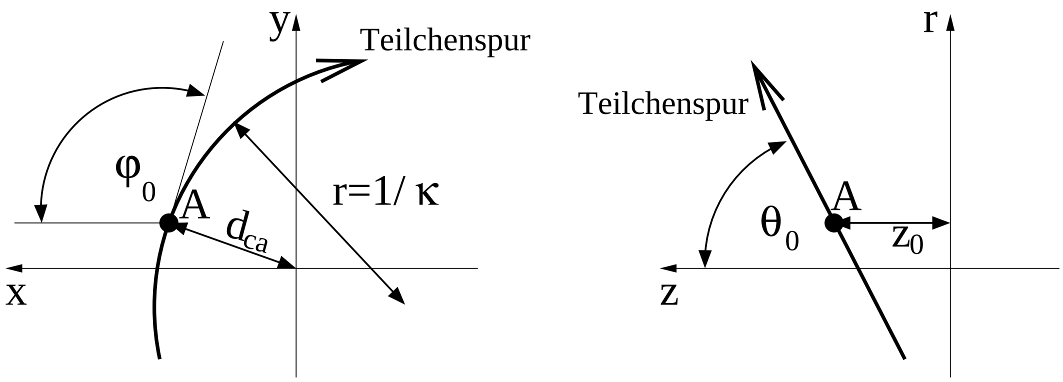



*Abbildung 2.4: Projektion einer Spur in die xy-Ebene (links) und die rz-Ebene (rechts) des H1-Koordinatensytems. Eingezeichnet sind die fünf im Text erläuterten Spurparameter, durch die die Spur beschreibende Helix festgelegt wird. Zusätzlich ist der Punkt* A *markiert, an dem Orts- und Richtungsvektor der Spur einen rechten Winkel bilden und somit der Abstand zwischen Spur und Koordinatenursprung minimal ist.*

Punkt, an dem die Spur diesem am nächsten kommt, ist durch $d_{ca}$[6] gegeben. Das Vorzeichen ist mit dem von $\kappa$ verknüpft und wird zum Beispiel in [Pre-1992] erläutert.

- Azimutalwinkel $\varphi_0$:
  Der Winkel zwischen der $x$-Achse und der Tangente an die Spur im durch $d_{ca}$ festgelegten Punkt wird mit $\varphi_0$ bezeichnet.

- Abstand $z_0$:
  $z_0$ ist definiert als Abstand des $d_{ca}$-Punktes vom Koordinatenursprung entlang der $z$-Achse.

- Polarwinkel $\theta_0$:
  Der Winkel zwischen Spur und $z$-Achse im $d_{ca}$-Punkt wird mit $\theta$ bezeichnet.

Da wie oben beschrieben die CJC nur eine relative ungenaue Bestimmung der $z$-Koordinate erlauben, ist zwischen den beiden Jetkammern eine $z$-Kammer COZ[7] eingebaut. Signaldrähte senkrecht zum Strahlrohr ermöglichen eine $z$-Messung mit einer Präzision von 200 - 500 $\mu$m abhängig vom Polarwinkel der Spur. Direkt anschließend noch innerhalb der CJC2 ist eine weitere Proportionalkammer COP[8] eingebaut, die wie die CIP schnelle Signale für das H1-Triggersystem liefert.

In den Raumwinkelbereich von $7^\circ \leq \theta \leq 25^\circ$ gestreute Teilchenspuren werden von einem System vorwärtiger Spurkammern FTD[9] $\boxed{4}$ vermessen. Dieses besteht aus mehreren Driftkammern, die jeweils für die Messung der $\varphi$- oder $r$-Koordinate optimiert sind.

---

6 **D**istance of **C**losest **A**pproach = Abstand der größten Annäherung
7 **C**entral **O**uter **Z**-Chamber
8 **C**entral **O**uter **P**roportional Chamber
9 **F**orward **T**racking **D**etektor

Ebenfalls nur kurz soll hier der rückwärtige Bereich erwähnt werden, in dem für die HERA-II-Meßperiode eine Proportionalkammer, BPC[10], $\boxed{5}$ zur Spurerkennung dient.

Obwohl es sich nicht um Spurkammern im eigentlichen Sinne handelt, soll an dieser Stelle kurz auf das Flugzeitsystem, TOF[11]-System, eingegangen werden. Zahlreiche Szintillatorstationen entlang des Strahlrohrs messen mit hoher Zeitauflösung Teilchen, die durch Wechselwirkung des Strahls mit Restgasatomen im Strahlrohr oder dem Strahlrohr selbst entstehen. Dadurch können für das Triggersystem enge Zeitfenster für die Separation echter Wechselwirkungen von Untergrundereignissen bereitgestellt werden. Zu diesem Zweck dient auch die sogenannte VETO-Wand $\boxed{15}$, die sich in Protonstrahlrichtung vor dem H1-Detektor befindet. Eine ausführliche Beschreibung des TOF-Systems bei HERA-I findet sich in [Wis-1998].

### 2.1.2 Kalorimeter

Der Wechselwirkungspunkt ist nahezu hermetisch mit Instrumenten zur Energiemessung —Kalorimetern— umgeben. Den zentralen und vorwärtigen Bereich deckt ein Flüssigargon-Kalorimeter mit einer Winkelakzeptanz von $4^\circ \leq \theta \leq 154^\circ$ ab [And-1993]. Bei diesem Samplingkalorimeter dient neben dem Argon als aktivem Material Blei in der elektromagnetischen $\boxed{6}$, beziehungsweise Eisen in der hadronischen Sektion $\boxed{7}$ als Absorber. Die fein segmentierte Bauweise mit über 44000 Auslesezellen erlaubt neben der Energiemessung eine gute Ortsauflösung, die schon für Triggerzwecke zu Verfügung steht.

Im rückwärtigen Bereich von $153^\circ \leq \theta \leq 178^\circ$ ist seit 1995 das Spaghettikalorimeter, SpaCal, eingebaut [H1S-1997], das aus in eine Bleimatrix eingelassenen Szintillatorfasern besteht. Die elektromagnetische $\boxed{8}$ und hadronische Sektion $\boxed{9}$ unterscheiden sich hier hauptsächlich durch die Packungsdichte der Fasern und ihrer Durchmesser. Durch die Einteilung in $4{,}05 \times 4{,}05\ \mathrm{cm}^2$ große Zellen wird in der elektromagnetischen Sektion neben einer guten Energieauflösung auch eine gute Ortsauflösung erreicht. Da durch die Szintillatortechnik zusätzlich eine präzise Zeitinformation gewonnen werden kann, steht mit dem SpaCal auch ein gutes Instrument für einen Elektrontrigger im rückwärtigen Bereich des Detektors zu Verfügung. Dies wird in Abschnitt 2.2.1.3 vertieft. Hadronen, die nicht vollständig in einem der Kalorimeter gestoppt werden, können auch noch in dem im folgenden Kapitel beschriebenen zentralen Myonsystem nachgewiesen werden.

Teile des hadronischen Endzustandes, die unter sehr kleinen Winkeln gestreut werden, können im PLUG[12]-Kalorimeter $\boxed{10}$ nachgewiesen werden. Für die HERA-II-Periode sind dafür im vorwärtigen Bereich nahe dem Strahlrohr alternierend Schichten aus Szintillatormaterial und Blei eingebaut worden. Zur Messung der Luminosität werden bei H1 Ereignisse der Bethe-Heitler-Streuung $ep \rightarrow ep\gamma$, deren Wirkungsquerschnitt sehr präzise zu berechnen ist, aufgezeichnet. Elektron und Photon werden dabei unter sehr kleinen Winkeln gestreut und durch zwei Detektoren am Strahlrohr in Richtung des auslaufenden Elektronstrahls registriert [H1-1996]. Hier soll nachfolgend nur der sogenannte Elektron-Tagger (ET33) betrachtet werden, der 33,4 m vom Wechselwirkungs-

[10] **B**ackward **P**roportional **C**hamber
[11] **T**ime **o**f **F**light
[12] plug = Pflock, Stöpsel

punkt entfernt Elektronen durch Emission von Cerenkovlicht in einem Kristall identifiziert. Dieser Detektor kann auch zum Nachweis von Elektronen aus Photoproduktionsereignissen verwendet werden. Der Photondetektor dient dann als Vetozähler. Um die räumliche Akzeptanz für den Elektronnachweis bei Photoproduktionsereignissen zu steigern, wurden bei HERA-I zwei zusätzliche ET-Stationen eingebaut: Ein weiterer Cerenkovdetektor (ET44)bei $z = -43,2$ m [And-1996] und ein aus überzähligen SpaCal-Modulen aufgebauter Detektor bei $z = -8$ m (ET08) [And-2000]. Bei HERA-II stehen durch den Umbau der Magnetkonfigurationen nur noch zwei Stationen bei $z = -6$ m und $z = -40$ m zur Verfügung [And-1998].

### 2.1.3 Myonsysteme

Zur Rückführung des durch die Magnetspule $\boxed{11}$ erzeugten Feldes ist diese von einem Eisenjoch $\boxed{12}$ umgeben, das sich aus einzelnen Plattenschichten zusammensetzt. Deren Zwischenräume sind in einem Winkelbereich von $4^\circ \leq \theta \leq 171^\circ$ mit 16 Lagen von Streamer-Rohr-Kammern instrumentiert. Das Joch ist zusätzlich außen und im Zentralbereich auch innen mit Myonkammern bestückt. Diese Detektoren bilden zusammen das zentrale Myonsystem, das für Myonen eine Ortsauflösung von 5 bis 15 mm erreicht. Mit einer geringen Wahrscheinlichkeit werden andere Teilchen nicht vollständig in den weiter innen liegenden Detektorkomponenten absorbiert und deponieren Energie in den Streamer-Rohr-Kammern, die somit auch eine kalorimetrische Funktion haben.

Zum Nachweis von Myonen in Vorwärtsrichtung dient das Vorwärts-Myon-Spektrometer, das Myonspuren durch mehrere Driftkammerlagen $\boxed{13}$ in einem Winkelbereich von $3^\circ \leq \theta \leq 17^\circ$ nachweist. Das Feld eines Toroid-Magneten $\boxed{14}$ zwingt die Myonen dabei auf Kreisbahnen und ermöglichst eine Impulsmessung.

## 2.2 H1-Triggersystem

Das H1-Triggersystem selektiert interessante Ereignisse im H1-Detektor und synchronisiert die Auslese der einzelnen Subdetektoren in Bezug auf den HERA-Takt, der durch die Kollisionsrate der Teilchenbündel von 10,4 MHz gegeben ist. Da für die Datenauslese die Detektorelektronik etwa 1 ms lang angehalten werden muß und der vollständige Datensatz eines Ereignisses schließlich nur mit etwa 10 Hz gespeichert werden kann, muß das Triggersystem eine Ratenreduktion von sechs Größenordnungen bei minimaler Detektortotzeit leisten können. Abbildung 2.5 zeigt das vierstufige H1-Triggerschema mit jeweiliger Angabe der maximalen Eingangsrate und der zur Verfügung stehenden Bearbeitungszeit.

Die Triggerentscheidung jeder Stufe wird durch die zentrale Triggerlogik CTL[13] aus Informationen von Subtriggersystemen, die zum Teil Detektorkomponenten direkt zugeordnet sind, ermittelt. In diesen wird eine schnell verfügbare Untermenge von Detektorinformationen verarbeitet, die im folgenden als Triggerdaten bezeichnet wird. Daraus berechnen die Subtriggersysteme sogenannte Trigger-Elemente (TE), die bestimmten topologischen, energetischen oder zeitlichen Informationen entsprechen. Beispiele für TEs

[13] **C**entral **T**rigger **L**ogic

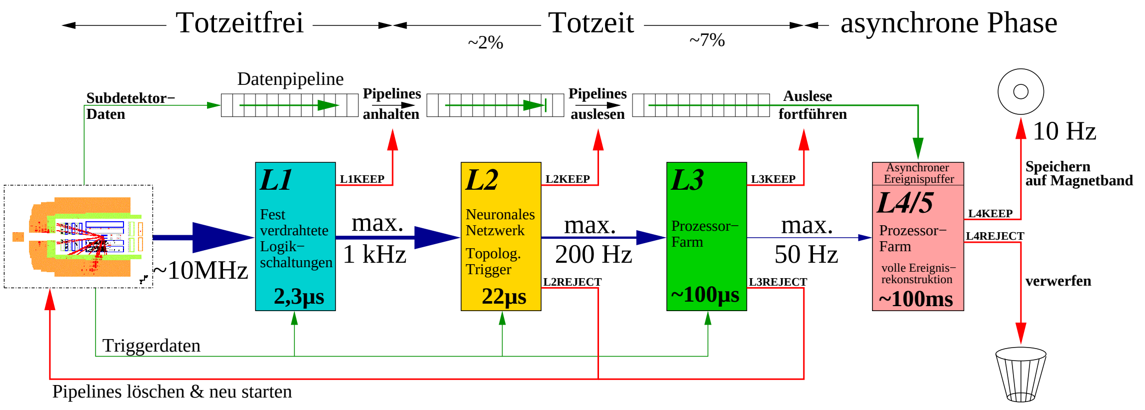



*Abbildung 2.5: Übersicht über die Stufen des H1-Triggersystems. Angegeben sind die Eingangsraten, für die eine Stufe maximal ausgelegt ist, sowie die Zeit, die dort für die Berechnung einer Triggerentscheidung jeweils zu Verfügung steht. Parallel dazu ist durch die Datenpipelines die Funktionsweise des Datenauslesesystems dargestellt. Nur während der L2- und L3-Entscheidungszeit wird Detektortotzeit akkumuliert.*

finden sich in den nächsten Abschnitten, in denen die Charakteristiken der einzelnen Stufen und der Aufbau einzelnen Subtriggersysteme dieser Stufe vorgestellt werden. Zunächst wird dabei jeweils auf die Funktion und Strategien der zentralen Triggerlogik für diese Stufe eingegangen.

Die im folgenden genannten Triggerraten beziehen sich auf das voll ausgebaute System. Bei Wegfall einer Triggerstufe muß die Rate auf den vorhergehenden Stufen schon so weit reduziert werden, daß maximal mögliche Eingangsraten der folgenden Stufen nicht überschritten werden. Eine positive Entscheidung einer Triggerstufe wird mit **Keep** bezeichnet, das Verwerfen eines Ereignisses mit **Reject**.

Da das Datenauslesesystem durch die CTL synchronisiert wird, sind beide eng verknüpft. Daher ist auch das Verhalten des Auslesesystems in Abbildung 2.5 mitskizziert und soll bei der Vorstellung der einzelnen Triggerstufen mit berücksichtigt werden. Ergänzend soll an dieser Stelle schon erwähnt werden, daß jedes Subdetektorsystem durch die CTL über sogenannte STCs[14] synchronisiert wird. Darüber werden die zentralen Entscheidungen jeder Triggerstufe sowie Informationen über Beginn und Ende der Datennahme an die Systeme gemeldet. Die zeitlich begrenzte, zusammenhängende Aufzeichnung von Ereignissen bei gleichbleibenden experimentellen Bedingungen wird als Run bezeichnet.

Über zwei PQZP[15] und Quickbus[16] genannte Bussysteme (im folgenden auch unter dem Begriff L2L3-Bus zusammengefaßt) werden die L1-Triggerdaten auch höheren Triggerstufen zu Verfügung gestellt. Die über den L2L3-Bus versendeten Daten werden in den jeweiligen Abschnitten der L1-Subtriggersysteme grob beschrieben.

[14] **S**ubsystem **T**rigger **C**ontroller = Elektronik zum Signalaustausch zwischen Subsystem und CTL
[15] **P**arallel **Q**uickbus **Z**ero-Suppression **P**rocessor
[16] für Daten des Flüssigargon-Kalorimeters

### 2.2.1 Triggerstufe L1

Jede Detektorkomponente muß sicherstellen, daß bei fortlaufender Datennahme die Informationen aller Auslesekanäle noch mindestens 3 $\mu$s nach einer Wechselwirkung zu Verfügung stehen. Dazu werden diese Informationen meistens in einem, im folgenden Pipeline genannten, FIFO[17]-Speicher gehalten. Während dieser Zeit kann totzeitfrei die L1-Triggerentscheidung getroffen werden. Durch die strenge Zeitvorgabe kommt für deren Berechnung hauptsächlich festverdrahtete Elektronik zur Analyse der Triggerdaten eines Subdetektors zum Einsatz. Im Falle einer **L1Keep**-Entscheidung werden die Pipelines angehalten und Detektortotzeit wird akkumuliert. Eine explizite **L1Reject**-Entscheidung wird nicht generiert, da die Detektordaten nach einiger Zeit automatisch aus den Pipelines gelöscht werden. Schon durch L1 kann eine Ratenreduktion auf ungefähr 1 kHz erreicht werden.

Um die L1-Entscheidung zu berechnen, liefern die Subtriggersysteme bis zu 256 TE an die CTL. Besondere Anforderungen werden dabei an die Synchronisation der verschiedenen Subtriggersysteme gestellt, um zu garantieren, daß alle zu einem HERA-Takt gehörenden TE gleichzeitig die CTL erreichen [Sef-1995]. Nach einem konfigurierbaren Schema werden dort durch logische UND- oder ODER-Verknüpfungen[18] von Triggerelementen 128 sogenannte L1-Subtriggerbits $s^{L1}$ gebildet. Eine ODER-Verknüpfung aller $s^{L1}$ liefert dann die L1-Entscheidung:

$$L1Keep = \sum_{i=0}^{127} s_i^{L1} \tag{2.2}$$

Da manche $s^{L1}$ grundsätzlich oder abhängig von der Strahlqualität eine sehr hohe Rate aufweisen und die Bandbreite des Systems limitiert ist, ist die Möglichkeit einer Ratenuntersetzung vorgesehen. Ein Zähler mit variierbarem Überlauf $n$ sorgt dafür, daß nur jedes $n$-te Auftreten eines positiven $s^{L1}$ in der finalen ODER-Operation berücksichtigt wird. Die Berechnung der Untersetzungsfaktoren ist mittlerweile weitgehend automatisiert und optimiert [Sch-1999].

Die L1-Subtriggersysteme können grob in vier Kategorien eingeteilt werden:

- Spurtrigger
- Kalorimetertrigger
- Myonsystemtrigger
- Elektron-Taggertrigger

Zusätzliche wird das TOF-System verwendet, um globale Vetobedingungen bereitzustellen. Beispielhaft sollen diejenigen Systeme vorgestellt werden, die für die spätere Beschreibung des FTT eine Rolle spielen.

[17] **F**irst **I**n **F**irst **O**ut = Datenreihenfolge bei Ein- und Ausgabe gleich

[18] ODER-Verknüpfungen werden in folgenden Formeln durch Additionen, UND-Verknüpfungen durch Multiplikationen dargestellt

#### 2.2.1.1 DCR$\phi$-Trigger

Aufgabe des DCR$\phi$-Triggers [Wol-1992] ist es, in der CJC Spuren in der $r\varphi$-Ebene zu identifizieren und grob deren Spurparameter zu bestimmen. Um die enge Zeitvorgabe einzuhalten, werden nur Signale von 10 radialen Drahtlagen, 7 in der CJC1 und 3 in der CJC2, ausgewertet. Zur Spuridentifikation werden die Eingangssignale mit etwa 10000 im vorraus berechneten Signalmustern verglichen und daraus grobe $\varphi$- und $p_t$-Informationen abgeleitet. Die erreichbare Auflösung erlaubt nur die Aussage, ob Impulsschwellen von 400, beziehungsweise 800 MeV für eine Spur überschritten werden. Spuren unterhalb von 400 MeV werden nicht erfaßt, wodurch auch keine exakte Bestimmung der Spurmultiplizitäten möglich ist. Die Granularität der Masken erlaubt es, für Spuren mit einem Transversalimpuls von weniger als 1 GeV das Ladungsvorzeichen zu bestimmen. Da nur Muster für Spuren mit $d_{ca} \leq 2$ cm verwendet werden, können mit Hilfe des DCR$\phi$-Triggers Untergrundereignisse durch kosmische Myonen oder Strahl-Rohr-Wechselwirkungen mit einem Ursprung abseits der Strahlachse identifiziert und effizient unterdrückt werden. Durch verschiedene TE wird dem zentralen Trigger die Zahl der gefundenen Signalmuster mitgeteilt, wobei auch eine zusätzliche Information über Ladungsvorzeichen oder Impulsbereich kodiert sein kann. Für den L2L3-Bus wird für 45 $\varphi$-Sektoren das Auftreten von Spuren oberhalb einer der beiden Impulsschwellen unter Berücksichtigung des Ladungsvorzeichens in einem Bitmuster kodiert.

#### 2.2.1.2 $z$-Vertex-Trigger

Neben schnellen Zeitinformationen liefern die Proportionalkammern durch ihre Segmentierung auch eine Ortsinformation, aus der der $z$-Vertex-Trigger versucht, Spuren in der $rz$-Ebene zu rekonstruieren. Wurden während der HERA-I-Phase Information der alten CIP, der COP und mehrere Lagen von Proportionalkammern im Vorwärtsbereich verwendet, stützt sich bei HERA-II die Entscheidung auf die COP und die neue, fünflagige CIP2000 [Urb-2000b]. Zusätzlich wird durch die CIP alleine die $z$-Koordinate des Vertex mit schlechterer Auflösung, aber größerer Überdeckung in $z$-Richtung bestimmt. Die in Abschnitt 2.1.1 beschriebene Padaufteilung erlaubt es in der CIP, aus Signalen mehrerer Lagen in Richtung des nominellen Vertex zeigende Spuren, sogenannte Strahlen zusammenzusetzen. Wie in Abbildung 2.3 dargestellt, werden die Schnittpunkte der Strahlen mit der Strahlrohrachse bestimmt und deren $z$-Koordinaten in einem virtuellen Histogramm registriert. Kann in einem Intervall des Histogramms ein eindeutiges Maximum von Einträgen identifiziert werden, wird dadurch die $z$-Position des Ereignisvertex festgelegt. Untergrundereignisse werden durch eine flache Verteilung ohne ausgeprägtes Maximum oder ein weit außerhalb des nominellen Bereichs liegendes Maximum charakterisiert. Verschiedene Trigger-Elemente spiegeln die Verteilung der Einträge im $z$-Vertex-Histogramm wieder. Über den L2L3-Bus wird anschließend das gesamte Histogram übertragen.

#### 2.2.1.3 SpaCal-IET

Der SpaCal-IET[19] analysiert Energieeinträge in dem in Abschnitt 2.1.2 beschriebenen, elektromagnetischen SpaCal und vergleicht diese mit vorgegebenen Schwellenwerten. Jeweils vier elektromagnetische Kalorimeterzellen werden zu einer sogenannten Triggerzelle zusammengefaßt, indem die darin deponierten Energien aufsummiert werden. Abbildung 2.6 zeigt, wie durch eine Schiebefenster-Technik ("Sliding Window Technique") aus vier Triggerzellen sogenannten Trigger-Türmen ("Trigger-Tower") dergestalt aufgebaut werden, daß eine Triggerzelle jeweils vier verschiedenen Trigger-Towern zugeordnet wird. Übersteigt die Energiesumme eines Trigger-Towers einen vor drei vorgegebenen Schwellwerten, wird ein entsprechendes Triggerelement aktiviert. Der Bereich nahe des Strahlrohrs wird dabei vom restlichen SpaCal separat behandelt. Für den L2L3-Bus wird die Energieverteilung im SpaCal als Bitmuster aufbereitet, wobei als zusätzliche grobe Information jeweils 4×4 Trigger-Tower durch eine logische ODER-Operation zu einem LIET[20]-Bereich verbunden werden. Ein gesetztes LIET-Bit zeigt ein Überschreiten einer der Energieschwellen in einem der assoziierten Trigger-Tower an.

#### 2.2.1.4 Kalorimeter-Tower-Trigger

Für das Flüssigargon-Kalorimeter werden noch in der ersten Elektronikstufe im elektromagnetischen Teil 16 und im hadronischen Teil vier Auslesezellen zu sogenannten Triggerzellen zusammengefaßt. Aus bis zu vier elektromagnetischen und bis zu sechs hadronischen Triggerzellen werden in $\theta$-Richtung 23 und in $\varphi$-Richtung bis zu 32 Trigger-Tower aufgebaut. Wie in Abbildung 2.7 zu sehen ist, sind diese auf den Wechselwirkungspunkt ausgerichtet, so daß von dort kommende Teilchen ihre gesamt Energie möglichst in einem einzelnen Trigger-Tower deponieren. Für jeden Triggertower werden separat für elektromagnetischen und hadronischen Bereich zwei Energiesummen gebildet und diese mit vordefinierten Schwellenwerten verglichen, die hauptsächlich elektronisches Rauschen unterdrücken sollen. Aus jeweils bis zu vier Trigger-Towern werden abschließend insgesamt 240 sogenannte Big-Tower gebildet, die zur Bildung von Triggerelementen verwendet werden. Big-Tower mit einem Energieeintrag im elektromagnetischen, aber nicht im hadronischen Bereich werden dabei als Elektronkandidat interpretiert. Zusätzlich werden Energiesummen über bestimmte Bereiche des Kalorimeters gebildet und daraus der transversale Energiefluß eines Ereignisses bestimmt. Ein wichtiges Kriterium ist dabei, ob der Transversalimpuls des Elektronkandidaten durch die restlichen Energieeinträge balanciert wird. Diese Informationen werden in TE kodiert an die CTL geschickt. Über den L2L3-Bus werden Energiesummen in unterschiedlicher Granularität und Informationen über Big-Rays weitergeleitet.

Im Laufe der HERA-II-Datennahmeperiode soll zusätzlich ein Jet-Trigger in Betrieb genommen werden [H1-1999b]. Dieser erlaubt es, aus den Trigger-Towern räumlich begrenzte Energiedeposition im Kalorimeter, sogenannte Jets, zu identifizieren und dadurch die Topologie von Ereignissen mit höherer Präzision bestimmen zu können. Informationen über diese Jet-Kandidaten sollen anschließend über den L2L3-Bus bereitgestellt werden.

[19] **I**nclusiv **E**lectron **T**rigger
[20] **L**ocal **I**nclusiv **E**lectron **T**rigger

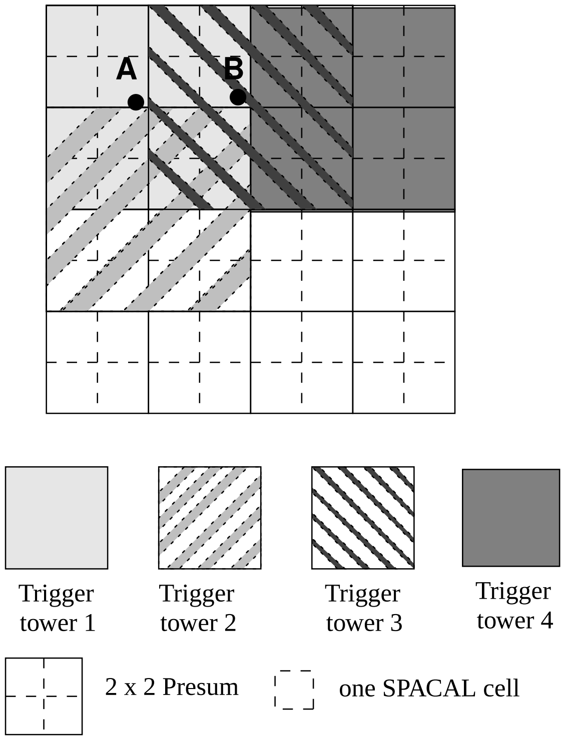



*Abbildung 2.6: Bildung von Trigger-Towern im SpaCal. 2×2 Kalorimeterzellen bilden eine Triggerzelle. Durch eine Sliding-Window-Technik wird eine Triggerzelle jeweils vier überlappenden Trigger-Towern zugeordnet, in denen die Energie von 2×2 Triggerzellen aufsummiert wird. Die Position des Fensters mit der höchsten Energiesumme wird für die weitere Triggerberechnung verwendet.*

#### 2.2.1.5 Digitaler Myon-Trigger

Durch Koinzidenzen von Treffern in bis zu fünf Lagen der Streamer-Rohr-Kammern werden Myon-Kandidaten des zentralen Digitalen Myon-Triggers definiert. Trigger-Elemente werden auf Basis von Myonkandidaten und deren Multiplizität für fünf Bereiche, die inneren und äußeren Bereiche der beiden Endkappen und den Zentralbereich berechnet. Da bei Zerfällen von Teilchen mit schweren Quarks häufig Myonen entstehen, eignen sich diese Trigger-Elemente in Verbindung mit anderen Trigger-Signalen zu deren Selektion. Für die höheren Triggerstufen werden über den L2L3-Bus Ortsinformationen mit feinerer Segmentierung zu Verfügung gestellt, indem jeder Bereich in jeweils 16 Module aufgeteilt wird. Wird in einem Modul mindestens ein Myonkandidat gefunden, wird dies durch ein gesetztes Bit angezeigt.

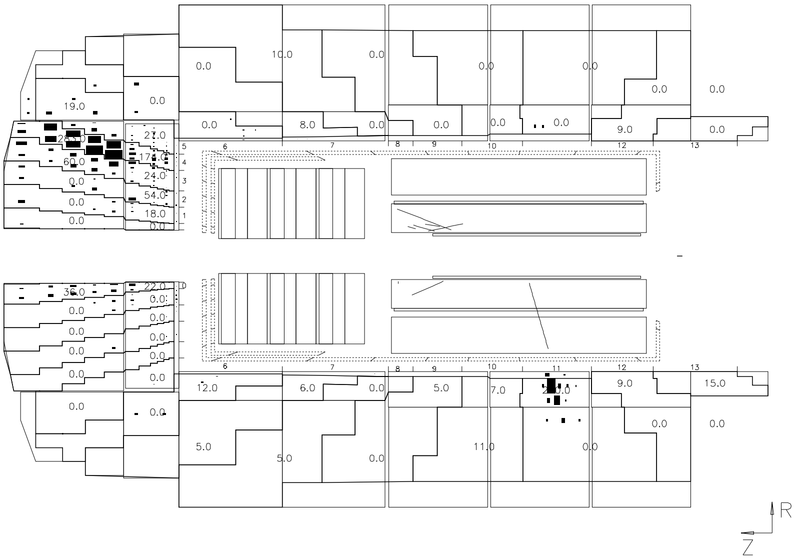


*Abbildung 2.7: Geometrische Struktur der Big Tower im Flüssigargon-Kalorimeter in der r-z-Projektion [And-1993]. Kalorimeterzellen werden dergestalt zu Big Tower zusammengefaßt, daß diese auf den Wechselwirkungspunkt ausgerichtet sind. Im linken oberen Teil ist die Signatur eines hadronischen, im mittleren unteren Teil eines elektromagnetischen Schauers durch ein gestreutes Elektron zu sehen.*

#### 2.2.1.6 Elektron-Tagger-Trigger

Ein Elektronkandidat mit einem Energieeintrag oberhalb einer konfigurierbaren Schwelle in einem der Elektron-Tagger (ET) kann zur Selektion von Photoproduktionsereignissen verwendet werden. Das entsprechende TE kodiert dabei, welcher ET angesprochen hat. Da die Detektoren, wie in Abschnitt 2.1.2 besprochen, nahe dem Strahlrohr installiert sind, wird die Triggerrate von Untergrundereignissen dominiert, die mit etwa 30 kHz auftreten [Gog-1994]. Die Rate physikalisch interessanter Ereignisse ist im Vergleich dazu drei Größenordnungen niedriger [And-1988]. Triggerelemente der ETs werden bei normaler Datennahme daher immer durch eine Koinzidenz mit TEs von anderen Triggersystemen verbunden. Für den L2L3-Bus werden keine Daten bereitgestellt.

### 2.2.2 Triggerstufe L2

Eine Entscheidungszeit von 22 $\mu$s erlaubt auf der zweiten Triggerstufe den Einsatz aufwendigerer Elektronik und die Kombination von Daten mehrerer Detektorkomponenten. Durch ein **L2Keep** beginnt die Auslese der Pipelines, die anderenfalls durch ein **L2Reject**[21] gelöscht und zur weiteren Datennahme neu gestartet werden. Durch L2 wird die Ereignisrate um einen Faktor vier auf etwa 200 Hz reduziert.

Abbildung 2.8 zeigt den aktuellen Aufbau der L2-Systeme, die über den L2L3-Bus

[21] Die Entscheidung **L2Reject** wird durch das Signal **FastClear** den STCs übermittelt.

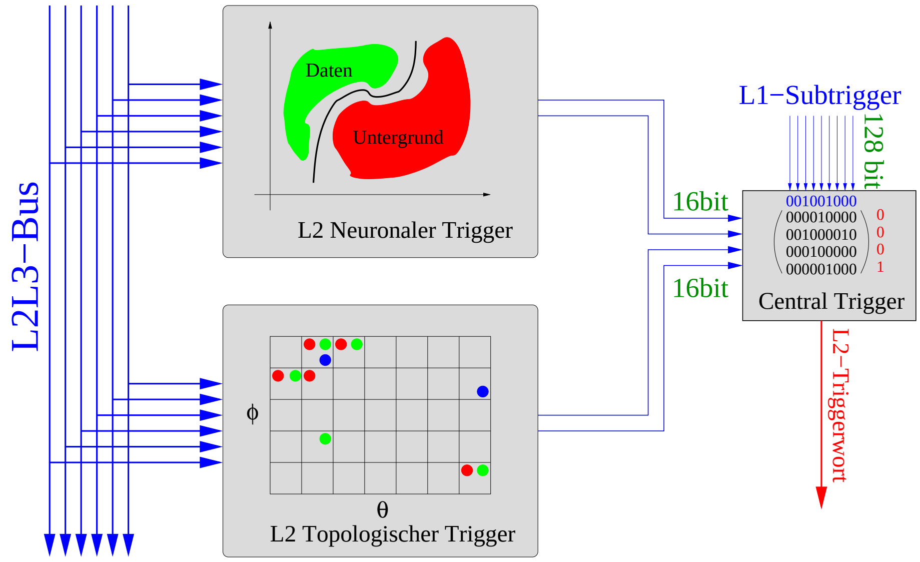



*Abbildung 2.8: Schematische Darstellung der zweiten Stufe des H1-Triggersystems. Über einen speziellen L2L3-Bus beziehen die beiden zur Zeit eingesetzten Systeme Daten. Im Neuronalen Netzwerk-Trigger werden physikalisch interessante Ereignisse durch in Software implementierte neuronale Netze von Untergrundereignissen separiert. Der Topologische Trigger überlagert Informationen verschiedener Subdetektorkomponenten in der $\theta$-$\varphi$-Ebene und sucht darin nach Signaturen verschiedener Ereignisklassen. Die dabei jeweils generierten L2-Subtriggerbits werden zur Validierung von L1-Entscheidungen verwendet.*

Daten beziehen. Die beiden Systeme Neuronaler Netzwerk-Trigger und Topologischer Trigger, die in den folgenden Abschnitten kurz vorgestellt werden sollen, senden an die CTL zur Zeit 32 L2-Triggerelemente, die dort logisch zu 48 L2-Subtriggerbits $s^{L2}$ kombiniert werden.

Eine Ratenreduzierung wird erreicht, indem für ausgewählte L1-Subtriggerbits eine Validierung durch die $s^{L2}$ gefordert wird [Nic-1998]. Dazu wird aus dem Vektor der aktuellen L1-Subtriggerbits $\vec{s}^{L1}$ und einer Validierungsmatrix $M^{L2}$ ein Validierungsvektor $\vec{v}^{L2}$ berechnet.

$$v_i^{L2} = M_{ij}^{L2} * s_j^{L1} \tag{2.3}$$

Dies soll an Hand von Abbildung 2.9 mit einer reduzierten Anzahl von L1- und L2-Subtriggerbits veranschaulicht werden. $M^{L2}$ gibt an, durch welchen $s^{L2}$ ein $s^{L1}$ validiert werden muss, um zu einem **L2Keep** beizutragen. Da dafür keine Informationen der L2-Subtriggersysteme benötigt werden, kann diese Berechnung direkt nach einem **L1Keep** durchgeführt werden. Zusätzlich ist die Möglichkeit implementiert, daß ein $s^{L1}$ keiner Validierung auf L2 bedarf. Zur L2-Entscheidungszeit genügt es dann, die Einträge des Vektors der aktuellen L2-Subtriggerbits $\vec{s}^{L2}$ mit denen von $\vec{v}^{L2}$ durch eine logische ODER-Operation zu verknüpfen. Ist mindestens ein $s^{L1}$ durch einen $s^{L2}$ validiert worden, generiert die CTL ein **L2Keep**.

$$L2Keep = \sum_{i=0}^{31} s_i^{L2} * v_i^{L2} \tag{2.4}$$

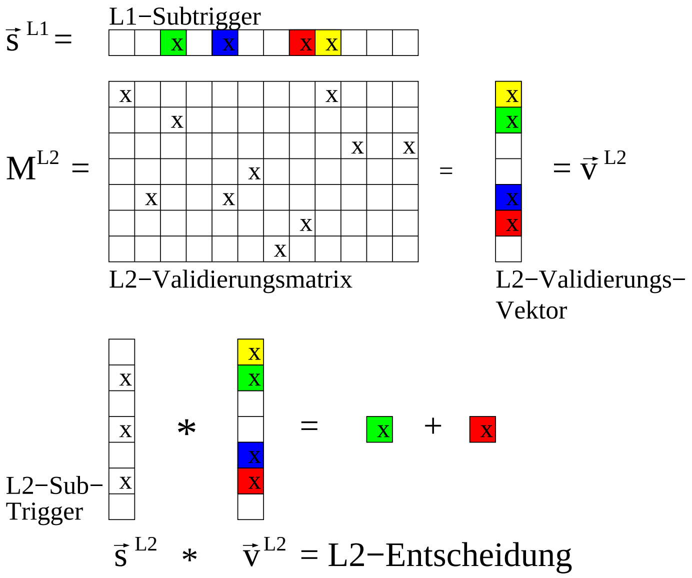



*Abbildung 2.9: Schematische Darstellung der Generierung der L2-Entscheidung in der zentralen Triggerlogik. Durch eine Validierungsmatrix wird für jedes L1-Subtriggerbit bestimmt, durch welches L2-Subtriggerbit es zu validieren ist. Damit kann sofort nach einem* **L1Keep** *ein L2-Validierungsvektor berechnet werden. Gibt es Koinzidenzen zwischen Einträgen in diesem Validierungsvektor und dem L2-Subtriggervektor, generiert die CTL eine positive L2-Entscheidung* **L2Keep***.*

Auch auf dieser Triggerstufe gibt es die bei 2.2.1 beschriebenen Möglichkeiten der Ratenuntersetzung.

#### 2.2.2.1 Topologischer Trigger

Der Topologische Trigger [Biz-1991, Bei-1998] projiziert Ortsinformationen verschiedener Subdetektoren in eine gemeinsame $\theta$-$\varphi$-Ebene, die, wie in Abbildung 2.10 skizziert, als 16×16-Matrix realisiert ist. Dabei können insbesondere für Kalorimeterinformationen mit verschiedenen Energieschwellen nacheinander 150 unterschiedliche Projektionsvorschriften abgearbeitet werden [Sch-1997]. Mit Hilfe unterschiedlicher Methoden, die hier nicht näher erläutert werden sollen, wird in dieser Matrix nach Signaturen physikalisch interessanter Ereignisse gesucht und durch einen Gütefaktor die Wahrscheinlichkeit quantifiziert, daß es sich nicht um ein Untergrundereignis handelt. Durch den Vergleich der Summe der Gütefaktoren verschiedener Methoden mit einem vorgegebenen Schwellenwert werden TE generiert und an die CTL geschickt.

#### 2.2.2.2 Neuronale Netzwerktrigger

Der L2 Neuronale Netzwerktrigger [Köh-1997] ist aus einer Farm von Parallelrechnern aufgebaut, auf denen neuronale Netzwerke implementiert sind. Aus den aufbereiteten Triggerdaten berechnet ein Netzwerk innerhalb von 8 $\mu$s ein 8-Bit-Ausgangssignal. Über-

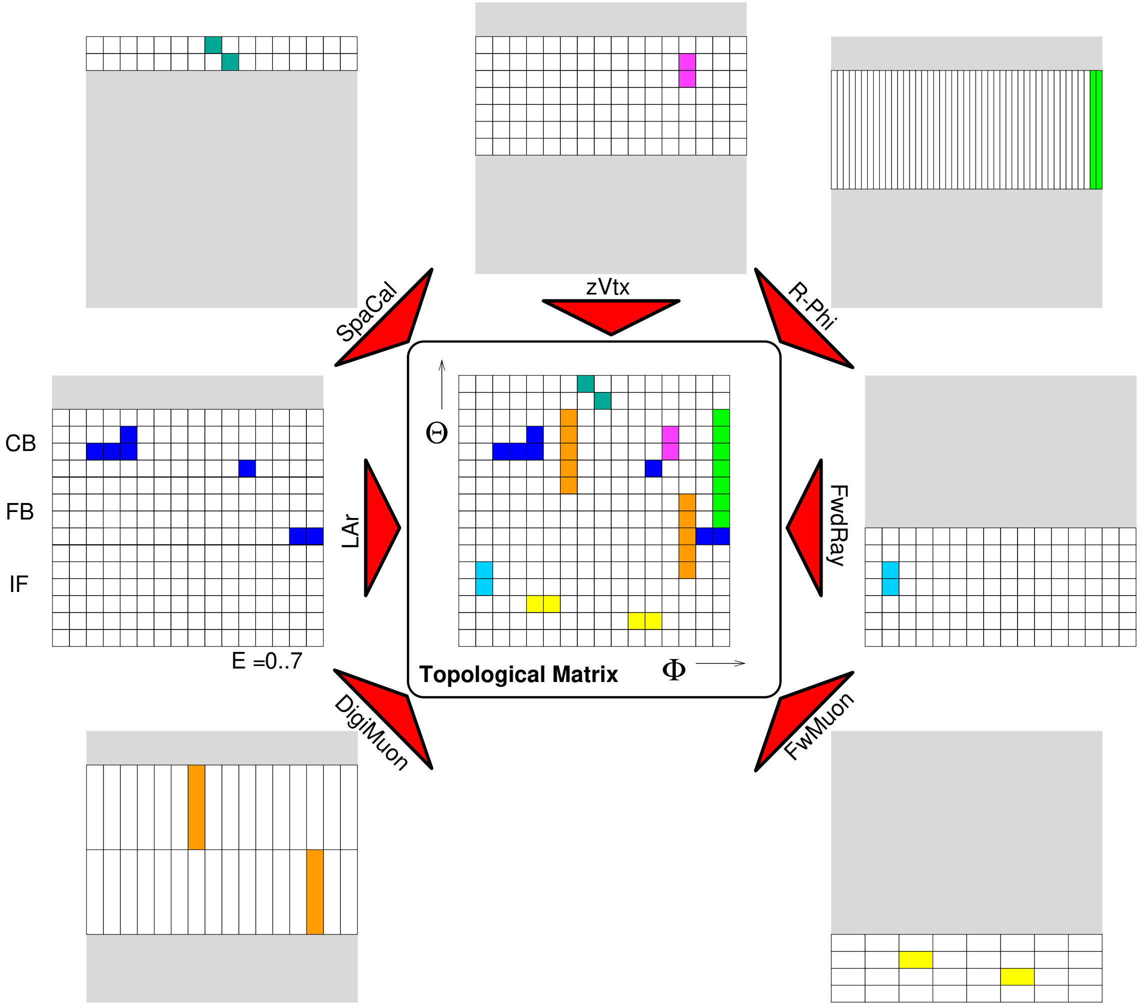



*Abbildung 2.10: Schematische Darstellung der Funktionsweise des L2 Topologischen Triggers nach [Hof-2000]. Aus den Triggerdaten werden für jede Detektorkomponente Ortsinformationen extrahiert und in $\theta$-$\varphi$-Matrizen unterschiedlicher Granularität und Winkelüberdeckung eingetragen. Durch verschiedene Projektionsvorschriften werden die einzelnen Matrizen in einer gemeinsamen $16\times16$-Matrix überlagert, um darin topologische Muster bekannter Ereignisklassen zu identifizieren.*

steigt dessen Wert eine vorbestimmte Schwelle, wird ein entsprechendes TE an die CTL gesendet. Vorbereitend werden die Netzwerke mit Daten von Untergrund- und Physikereignissen trainiert, wobei sowohl reale Daten als auch solche aus Simulationsprogrammen zum Einsatz kommen. Wie dabei das Netzwerk "lernt", die beiden Ereignistypen unterscheiden zu können, wird zum Beispiel in [Krä-1998] erläutert.

### 2.2.3 Triggerstufe L3

L3 arbeitet während der etwa 1 ms dauernden Datenauslese und kann diese durch ein **L3Reject** noch abbrechen. Um eine sinnvolle Ratenreduktion zu erreichen, sollte dieses allerdings schon nach etwa 100 $\mu$s verfügbar sein [Bar-1988a]. Diese Zeit reicht mit dem heutigen Stand der Technik aus, um bei L3 programmierbare Prozessoren einsetzen zu können. Wird dort ein **L3Keep** generiert, wird die Auslese fortgeführt, bis die Subdetektordaten vollständig in Datenpuffer transferiert worden sind und damit für die nachfolgende Triggerstufe zu Verfügung stehen. Mit der anschließenden Reaktivierung der Pipelines endet die Totzeit. Der aktuelle Aufbau der nachfolgenden Datenverarbeitungselektronik verlangt an dieser Stelle eine Ausgangsdatenrate von weniger als etwa 50 Hz.

Die Generierung einer L3-Entscheidung ist analog zu der bei L2 realisiert. Auch hier wird aus einer Validierungsmatrix ein Validierungsvektor vorab berechnet, wobei der Eingabevektor alle durch ein $s^{L2}$ validierten $s^{L1}$ enthält. Im Gegensatz zu L2 wird die L3-Entscheidung nicht nur nach einem festen Zeitpunkt, sondern auch schon vorher kontinuierlich berechnet. Sobald alle L3-Subtriggerbits, die zur Validierung benötigt werden, eingetroffen sind, wird die finale Entscheidung getroffen. Ziel ist es, durch ein schnelles **L3Reject** die Totzeit zu reduzieren. Im Gegensatz zu dem in [Bar-1988a] beschriebenen Schema wird die aktuelle Implementierung der L3-Stufe jedoch nicht primär zur Erkennung von Untergrundereignissen eingesetzt, so daß nicht ein einzelnes, nicht gesetztes $s^{L3}$ ein **L3Reject** bewirken kann [Sch-2002a].

Ende der 80er Jahre wurde schon einmal ein L3-System geplant und schließlich auch aufgebaut [Bar-1988a]. Die Triggerdaten wurden dabei von speziellen Karten empfangen und gespeichert, deren Speicher direkt durch den Prozessorbus von Prozessoren des Typs "29k"angesprochen werden konnte. An Hand der Daten sollten durch den oder die Prozessoren hauptsächlich Untergrundereignisse im Detektor herausgefiltert werden. Nach dem Einbau des Systems in die H1-Elektronik stellte sich allerdings heraus, daß die schnellsten damals verfügbaren Prozessoren mit einer Frequenz von 25 MHz noch nicht ausreichend Rechenleistung für aufwendige Triggerentscheidungen boten. Das System ist seitdem so konfiguriert, daß für jedes Ereignis ein **L3Keep** generiert wird.

### 2.2.4 Triggerstufe L4/5

Asynchron zum HERA-Takt arbeitet als vierte Triggerstufe eine Farm von kommerziellen Prozessorkarten. Während der HERA-I-Phase fand hier nur eine vorläufige Rekonstruktion der Ereignisse statt, da aktuelle Kalibrationsdaten für Subdetektoren nicht zu Verfügung standen [Cam-1992]. Die endgültige Rekonstruktion konnte erst zeitverzögert auf einem als L5 dienendem Multiprozessorrechner stattfinden. Ein neues Kalibrationsschema erlaubt für HERA-II die vollständige Rekonstruktion eines Ereignisses unter Ausnutzung aller Detektor- und Kalibrationsdaten schon auf der nun als L4/5 bezeichneten Prozessorfarm [Cam-1998]. Die innerhalb von etwa 100 ms extrahierten physikalischen Eigenschaften der im Streuprozeß identifizierten Teilchen werden anschließend zusammen mit den ursprünglichen Detektordaten mit einer Rate von etwa 10 Hz dauerhaft abgespeichert. Die neuerliche Reduzierung der Rate wird durch die Rekonstruktion spezifischer Endzustände und Identifikation von Untergrundereignissen erreicht.

# Kapitel 3

# Fast Track Trigger

Im Gegensatz zu allen anderen Triggersystemen im H1-Detektor, die jeweils für eine bestimmte Triggerstufe entwickelt wurden, wurde für den Fast Track Trigger (FTT) [Bai-2001b, Bai-2001a] erstmalig ein homogenes Konzept für die ersten drei Stufen des H1-Triggersystems entworfen. Dieses Konzept stellt höchste Ansprüche an Datentransferraten und Datenverarbeitungsgeschwindigkeit. Um einen Eindruck von der Komplexität zu vermitteln, werden nach einem Überblick über das Gesamtkonzept des FTT die einzelnen Komponenten in Aufbau und Funktion vorgestellt.

## 3.1 Gesamtkonzept

Ziel bei der Planung des FTTs war es, für die dritte Stufe des H1-Triggers spurbasierte Informationen zur Identifikation von Teilchenzerfällen aufgrund der charakteristischen, invarianten Masse des Endzustandes zu liefern. Dazu ist er notwendig, die Spurparameter der Teilchenbahnen in der CJC zu messen. Da diese Information bisher frühstens auf der vierten Triggerstufe zur Verfügung stand, muß es Aufgabe der ersten beiden Stufen des FTT sein, diese Spuren in Echtzeit zu identifizieren und ihre Parameter zu bestimmen.

Durch die erste Stufe werden die analogen Eingangsdaten der CJC digitalisiert. Nachdem ein Suchalgorithmus Signaldrahttreffer identifiziert hat, werden in separaten Bereichen der Kammer Spursegmente rekonstruiert. Anschließend werden diese Segmente zu vollständigen Spuren in der $r$-$\varphi$-Ebene zusammengefügt. Während der L2-Entscheidungszeit wird dies mit höherer Genauigkeit wiederholt und anschließend eine dreidimensionale Spurparameteranpassung an die Daten durchgeführt. Die resultierenden Spurparameter sind die Grundlage für die Rekonstruktion von Teilchenzerfällen durch die dritten Stufe. Tabelle 3.1 faßt die Aufgaben der einzelnen Stufen einschließlich der dafür zu Verfügung stehenden Zeit zusammen. Die Kapazität des FTT limitiert dabei die Zahl der verarbeitbaren Spuren auf 48. Untersuchungen haben gezeigt, daß dies für 98% der interessanten Ereignisse ausreichend ist [Bai-2001b]. Für derartige Untersuchungen steht ein Softwarepaket FTTEMU [Wis-2002] zu Verfügung, das die Funktionalität der Komponenten der ersten beiden Stufen simuliert. Dies ermöglichte schon während der Planungs– und Bauphase detailierte Tests des Systemverhaltens einschließlich der Simulation verschiedener Konfigurationen.

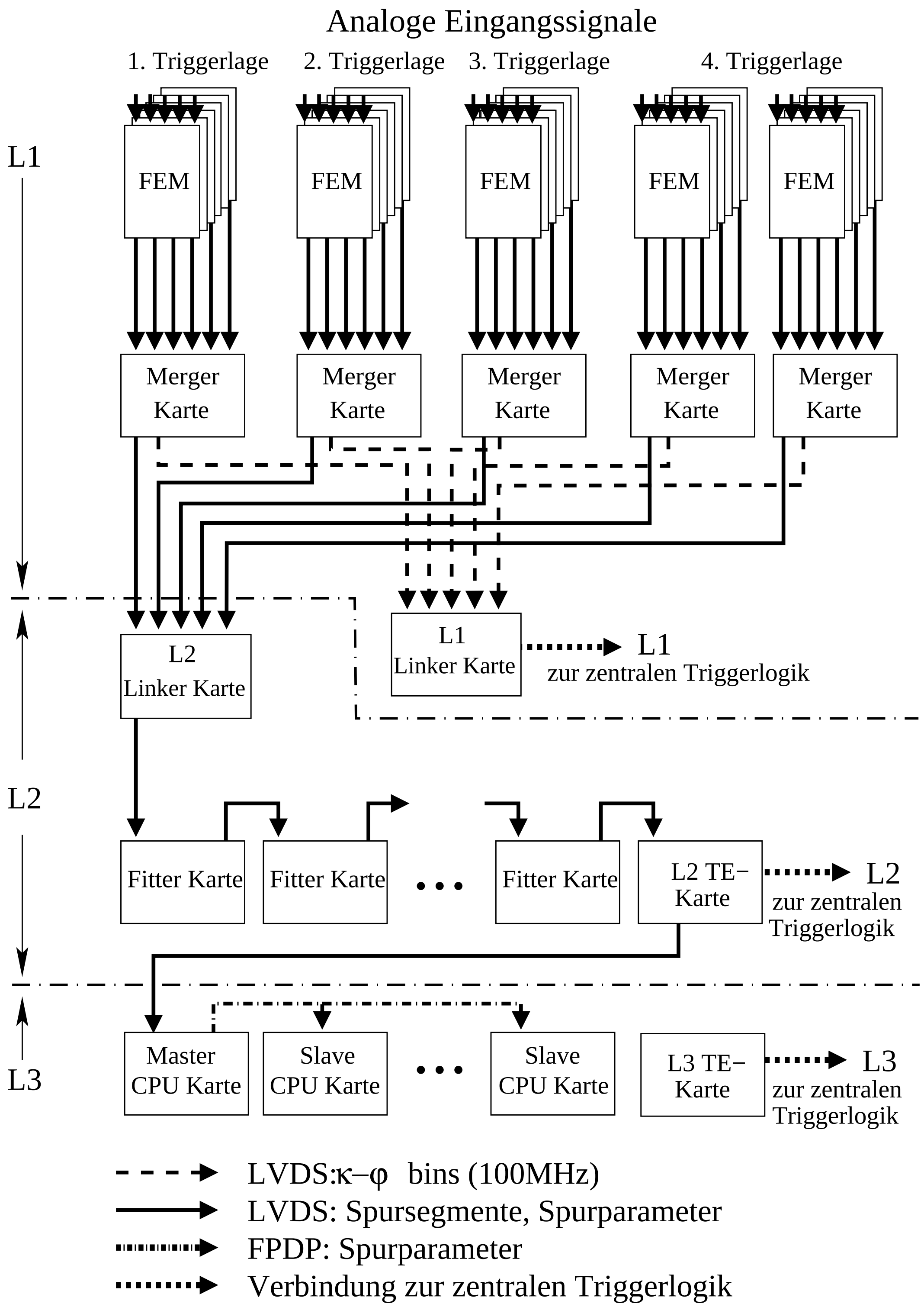



*Abbildung 3.1: Übersicht über die Komponenten des Fast Track Triggers*

| Triggerstufe | Aufgabe | Zeitbedarf [$\mu$s] (aufsummiert) |
|---|---|---|
| L1 | Ionisation und Driftzeit | 1,1 |
| | Analoge Signallaufzeit | 1,3 |
| | Signaldigitalisierung | |
| | Treffersuche | |
| | grobe Spursegmentsuche | 1,5 |
| | Datentransfer zur L1-Linker–Karte | 1,9 |
| | grobe Spursegmentverknüpfung | 2,0 |
| | Übertragung der Triggerinformationen | 2,1 |
| | **L1 Triggerentscheidung** | **2,3** |
| L2 | Verfeinerte Spursegmentsuche | 3,5 |
| | Datentransfer zur L2-Linker–Karte | 3,9 |
| | Verfeinerter Spursegmentverknüpfung | 9,1 |
| | Spurparameteranpassung | 12,9 |
| | Berechnung der Triggerinformation | 21,9 |
| | **L2 Triggerentscheidung** | **23,0** |
| L3 | Übertragung zu FTT-L3 | 30 |
| | L3 Algorithmen | $\lesssim$120 |
| | **L3 Triggerentscheidung** | **$\approx$ 120** |

*Tabelle 3.1: Aufgaben der drei Stufen des FTT. Eine detailierte Beschreibung der einzelnen Schritte findet sich im Text an den Stellen, an denen die entsprechend zugehörigen Karten vorgestellt werden. Als Zeitbedarf ist diejenige Zeit angegeben, die zwischen der Wechselwirkung und dem Abschluß des jeweiligen Schrittes bei der Bearbeitung von 48 Spuren verstrichen ist. Die Bearbeitung nachfolgender Schritte kann für einen Teil der Daten schon vorher beginnen. Werden weniger als 24 Spuren in einem Ereignis gefunden, halbiert sich die Zeit für die verfeinerte Spursegmentverknüpfung und die anschließende Parameteranpassung etwa. Die Zeiten für die Triggerentscheidung sind von der zentralen Triggerlogik vorgegeben. Für die ersten beiden Stufen sind dies feste Zeitpunkte,auf der dritten Stufe kann die Entscheidung nach mindestens 50 µs jederzeit erfolgen, sobald alle benötigten Informationen vorhanden sind. Nähere Angaben zur Bestimmung der einzelnen Zeiten finden sich in [Bai-2001b].*

## 3.2 FTT Komponenten

Nach der Spezifizierung der Eingangssignale im folgenden Abschnitt sollen die Komponenten des FTT detailiert beschrieben werden, die in Abbildung 3.1 im Überblick gezeigt sind. Die Hauptrichtung des Datenflusses ist in dieser Abbildung von oben nach unter.

### 3.2.1 Eingangssignale

Voruntersuchungen haben gezeigt, daß eine ausreichend genaue Bestimmung der Spurparameter mit einer eingeschränkten Auswahl an CJC-Drähten bei gleichzeitig akzeptablem technischen und finanziellen Aufwand möglich ist [H1-1999c]. Wie in Abbildung 3.2 dargestellt, werden in der CJC jeweils drei Drähte einer gemeinsamen Kammerzelle

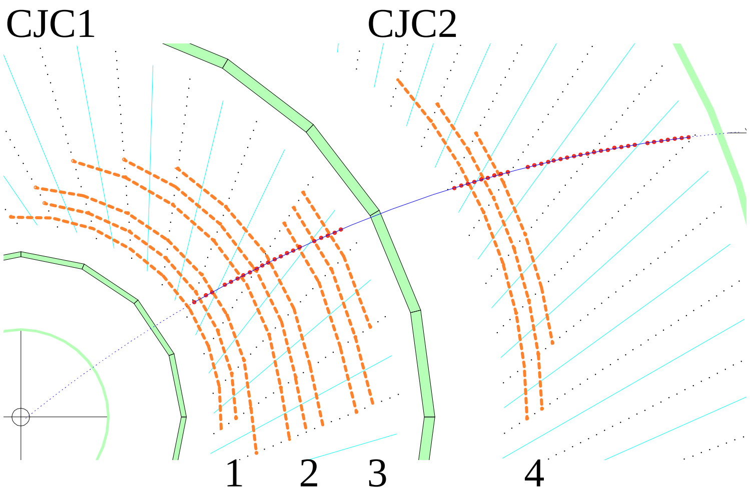



*Abbildung 3.2: FTT-Triggerlagen in der CJC. Jeweils drei Drähte bilden eine Triggerzelle, die wiederum radial zu vier Triggerlagen zusammengefaßt sind.*

zu sogenannten Triggerzellen zusammengefaßt. Radial werden dadurch vier Triggerlagen gebildet, die aus jeweils 30 Triggerzellen in der CJC1, beziehungsweise 60 Triggerzellen in der CJC2 bestehen. Um sensitiver für Spuren zu sein, die Zellgrenzen der CJC schneiden, werden aufgrund geometrischer Überlegungen der innerste Draht der in der $r$-$\varphi$-Ebene links benachbarten und der äußerste Draht der rechts benachbarten Triggerzelle zu einer jeweiligen Triggerzelle assoziiert. Um zusätzlich Informationen über die $z$-Position eines Drahttreffers zu erhalten, werden die Drähte an beiden Enden ausgelesen. Über Adapterkarten werden für die selektierten Drähte analoge Signale aus dem CJC-Auslesesystem ausgekoppelt und den sogenannten Front–End–Modulen zugeführt.

### 3.2.2 Front–End–Modul

Ein Front–End–Modul (FEM) verarbeitet die analogen Signale von fünf benachbarten Triggerzellen und liefert Parameter von Spursegmenten in diesen Zellen. Anhand von Abbildung 3.3 wird die Funktionalität eines FEMs vorgestellt. Für den Transfer aller Spurkammer- und Spursegmentinformation sind auf dem FEM spezielle Punkt-zu-Punkt–Datenleitungen vorgesehen. Die Flußrichtung dieser Daten ist in der Abbildung von links nach rechts. Über einen gemeinsamen Bus können die logischen Bauteile weitere Daten untereinander austauschen. Desweiteren besitzt dieser Bus eine Schnittstelle zum VME-Bus, dessen Eigenschaften in Kapitel 4.2.1.1 beschrieben werden und der die Verbindung zum globalen FTT-System herstellt. Zusätzlich sind alle FEMs innerhalb eines Elektronikgehäuses über einen speziellen FTT-L1-Bus miteinander verbunden, der es erlaubt, Daten zwischen den Karten auszutauschen.

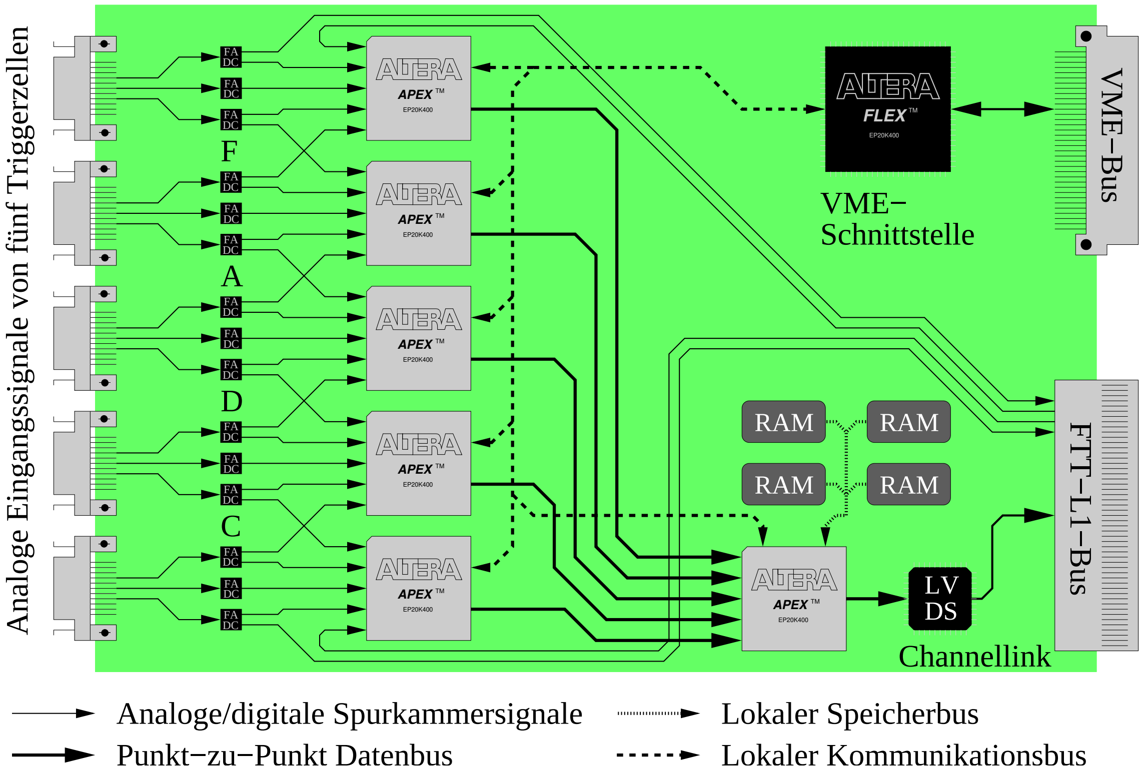



*Abbildung 3.3: Schematische Darstellung eines Front End Moduls (FEM). Der Datenfluß findet von links nach rechts statt und durchläuft folgende Bauteile und Stufen: Stecker für analoge Eingangssignale; Digitalisierung der Signale jeweils beider Drahtenden in FADCs; Treffer- und grobe Spursegmentsuche in fünf parallelen FPGAs (ALTERA APEX); Sammeln der Daten über Punkt-zu-Punkt–Datenbusse und Verifikation der Spursegmente mit feinerer Auflösung für FTT-L2 in einem weiteren APEX FPGA; Ausgabe der Daten über einen LVDS-Treiber zum rückwärtigen FTT-L1-Datenbus. Ein ALTERA FLEX FPGA dient über einen lokalen Bus zur Kommunikation aller Logikbauteile mit dem rückwärtigen VME-Bus.*

#### 3.2.2.1 Aufbereitung der Eingangssignale

Die analogen Signale werden zunächst durch FADCs[1] kontinuierlich mit 80 MHz digitalisiert. An Abbildung 3.3 ist zu erkennen, daß die Signale einer Triggerzelle auf einen nachfolgenden FPGA[2] (hier ALTERA APEX 20k [Alt-2002a]) gegeben werden. Hier findet auch der Austausch der oben erwähnten Signale benachbarter Triggerzellen statt.

Im FPGA wird separat für beide Drahtenden die in Abschnitt 2.1.1 beschriebene $Q$-$t$–Analyse für den FTT durchgeführt. Aus dem Verhältnis der gefundenen Signalhöhen läßt sich mit einer Auflösung von 6 cm die $z$-Koordinate eines Treffers bestimmen [Bai-2001b]. Diese Information wird zunächst nicht benötigt, da sich die Spursegmentsuche auf die $r$-$\varphi$-Ebene beschränkt. Sie kann für spätere Zwecke aber jederzeit wieder dem entspre-

1 **F**lash **A**nalog **D**igital **C**onverter; hier duale, lineare 8-Bits FADCs vom Typ AD9288

2 **F**ield **P**rogrammable **G**ate **A**rray; Elektronikbauteil mit frei programmierbaren Logikgattern und zusätzlichen, fest implementierten Komponenten

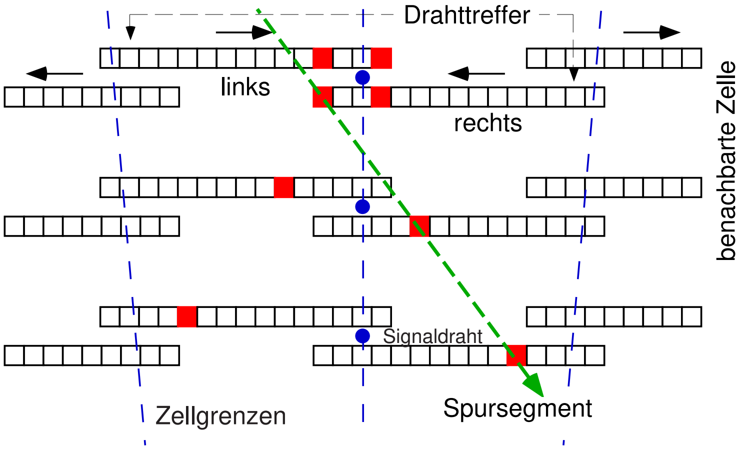



*Abbildung 3.4: Schematische Darstellung der Schieberegister einer Spurkammerzelle. Gefundene Drahttreffer werden durch einen Eintrag an einem Ende eines Schieberegisters repräsentiert. Die Einträge werden alle 12,5 ns verschoben, so daß die Position im Register einer Driftzeitinformation entspricht. Durch eine Mustererkennung werden aus den Bitmustern Spursegmente rekonstruiert. Da ein geometrischer Zusammenhang zwischen Spurkammerzelle und Schieberegister besteht, sind diese zur Verdeutlichung auf eine Zelle "projeziert". Aus Gründen der Übersichtlichkeit wurden hier die beiden Drähte der benachbarten Zellen nicht berücksichtigt. Der gestrichelte Pfeil symbolisiert ein die Kammer durchquerendes Teilchen.*

chenden Treffer zugeordnet werden.

#### 3.2.2.2 Suche nach Spursegmenten

Die Zeitinformation eines Treffers wird durch einen Eintrag in einem mit 80 MHz getakteten Schieberegister repräsentiert. In jedem FPGA sind zehn Schieberegister implementiert. Um Rechts-Links-Ambiguitäten zu berücksichtigen, sind einem Draht jeweils zwei Schieberegister zugeordnet, da a priori nicht bekannt ist, auf welcher Seite des Drahts die Primärionisation stattgefunden hat.

Das Konzept der Schieberegister soll anhand von Abbildung 3.4 erläutert werden. In dieser Abbildung sind aus Gründen der Übersichtlichkeit nur die Schieberegister für die drei Drähte einer Triggerzelle dargestellt. Schematisch ist die Geometrie einer Spurkammerzelle nachgebildet, die von einer Teilchenspur durchquert wird. Die Drahtebene liegt in der Mitte der Abbildung. Für jeden identifizierten Treffer wird je ein Bit an einem Ende der beiden Schieberegister gesetzt. Die Schieberegister werden mit einer Taktfrequentz von 80 MHz betrieben, so daß die Einträge alle 12,5 ns um eine Einheit verschoben werden. Zu einem gegebenen Zeitpunkt sind Einträge für nah am Draht verlaufende Spuren weiter vorgerückt, als für entferntere Spuren. Das Bitmuster ist also ein Abbild der Driftzeiten in der Kammer und somit auch der Teilchenspuren. Mit einem geeigneten Suchalgorithmus ist es möglich, aus dem Bitmuster der Schieberegister Spursegmente in

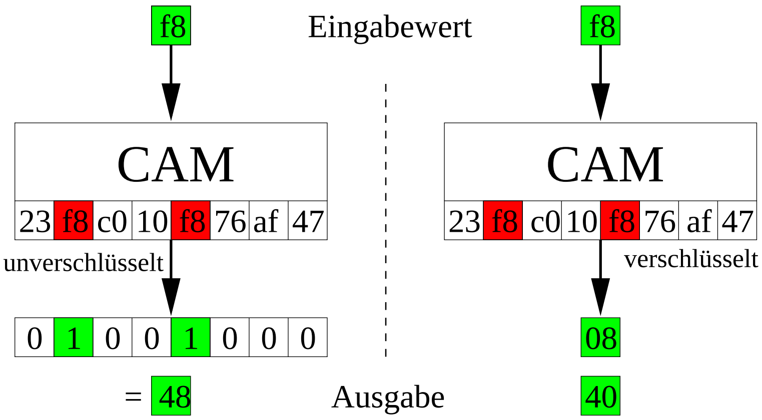



*Abbildung 3.5: Funktionsweise einer CAM. Eingabewerte werden mit vorab gespeicherten Werten verglichen und bei Übereinstimmungen die entsprechenden Speicherstellen als Antwort ausgegeben. Im unverschlüsselten Modus wird der Speicher direkt auf ein Bitmuster abgebildet, im verschlüsselten Modus die Speicherstelle als Adresse kodiert ausgegeben. Wird der Eingabewert mehrmals gefunden, werden alle Adressen seriell ausgegeben.*

einer Triggerzelle zu identifizieren.

Für schnelle Suchzwecke lassen sich CAMs[3] [Alt-2002b] in den FPGAs implementieren, deren Funktionalität invers zu der eines Speicherbausteins RAM[4] anzusehen und in Abbildung 3.5 skizziert ist. Im Falle eines RAMs werden Adressen Werte zugeordnet und diese bei Abfrage einer Adresse ausgegeben. Ein CAM wird ebenfalls zunächst mit Werten beschrieben. Nachfolgende Eingaben werden hier allerdings nicht als Adresse, sondern als Wert interpretiert. Nach bereits einem Taktzyklus erhält man als Antwort, ob dieser Wert ein- oder mehrmals im CAM gespeichert ist. Wird der CAM im "unverschlüsselten Modus" betrieben, repräsentiert jedes Bit des Ausgabewertes eine Speicherstelle. Ein Bit ist gesetzt, falls der Wert an dieser Stelle gespeichert ist. Im "verschlüsselten Modus" wird stattdessen eine Liste mit allen Adressen ausgegeben, an denen der vorgegebene Wert gespeichert ist. Im Vergleich zu einer Implementierung mit RAM-Bausteinen läßt sich somit das zu speichernde Datenvolumen deutlich reduzieren, da nur als gültig bekannte Werte berücksichtigt werden müssen.

CAMs ermöglichen eine effiziente Verifizierung eines Eingabewertes mit einer anschliessenden Verknüpfung mit weiteren Werten. Repräsentieren ein oder mehrere CAMs die vollständige Menge aller gültigen Werte, so erzeugt jeder gültige Eingabewert eine Adresse als Ausgabe. Diese kann als Zeiger in einen parallelen Speicher verwendet werden, in dem weitere, dem Eingabewert assoziierte Ausgabewerte stehen. Durch ihre schnelle Reaktionszeit finden CAMs bisher in zeitkritischen Bereichen von Netzwerk- und Rechnersteuerung ihre Anwendung, wie zum Beispiel bei der Zuordnung von Netzwerknummern zu Rechnern oder bei der rechnerinternen Speicherverwaltung. Im FTT werden CAMs zur schnellen Mustererkennung eingesetzt.

Vor Beginn einer Datennahmeperiode werden die Bitmuster aller gültigen Spuren

[3] **C**ontent **A**ddressable **M**emory
[4] **R**andom **A**ccess **M**emory

für eine bestimmte Kammerkonfiguration berechnet. Eine Spur ist als gültig einzustufen, wenn sie auf den nominellen Ereignisvertex zeigt und die Krümmung unterhalb eines Wertes liegt, der einem Transversalimpuls von $p_t = 100$ MeV entspricht. Um die Zahl der im folgenden Masken genannten Bitmuster auf der ersten Triggerstufe klein zu halten, erfolgt dort die Berechnung nur mit einer Genauigkeit, die der Zeitauflösung eines 20 MHz-Schieberegisters entspricht. Daraus resultieren etwa 2000 verschiedene Masken, so daß bis zu 64 parallele CAMs in einem FPGA notwendig sind, um alle Masken einer Triggerzelle aufnehmen zu können.

Synchron zum HERA-Takt wird alle 96 ns das Bitmuster einer Triggerzelle in den CAMs ausgewertet. Entsprechend der Auflösung der Masken werden vier benachbarte Schieberegisterstellen dabei verodert. In der unverschlüsselten CAM-Ausgabe entspricht jedes gesetzte Bit einem gefundenen Spursegment. Die Position des Bits läßt sich direkt mit einem Punkt im $\kappa\varphi$-Spurparameterraum verknüpfen. In den meisten Fällen kann jeweils nur einer der beiden Einträge eines Treffers mit einer Spur assoziiert werden, so daß an dieser Stelle die meisten Recht-Links-Ambiguitäten aufgelöst werden.

Die bisher beschriebene Datenverarbeitung findet innerhalb der L1-Latenzzeit von 2,2 $\mu$s statt. Um für die erste Triggerstufe Informationen bereitstellen zu können, werden die $\kappa$-, $\varphi$- und $z$-Werte jedes gefundenen Spursegments zu einem weiteren FPGA gesendet, der in Abbildung 3.3 rechts unten zu sehen ist. Dieser dient zunächst nur als Schnittstelle für den weiteren Datentransfer und sendet die Daten an eine Karte, die die Spursegmente zu vollständigen Spuren verbindet. Zahl der gefundenen Spuren und deren Topologie lassen sich in Verbindung mit verschiedenen Schwellen für Transversalimpulse für eine L1-Triggerentscheidungen auswerten. Da viele Bitmuster nur für einen HERA–Takt zu gültigen Spuren führen, läßt sich aus einem Maximum der Zahl der gefundenen Spuren der Zeitpunkt $t_0$, an dem das Ereignis stattgefunden hat, bestimmen. Die hier nur kurz angerissene Spursegmentverknüpfung wird an Hand der analog behandelten Spursegmente mit feinerer Auflösung in den Abschnitten 3.2.3 – 3.2.5 detailiert beschrieben.

#### 3.2.2.3 Verifizierung mit feinerer Auflösung

Nach einer positiven L1-Triggerentscheidung werden die Spursegmentparameter mit höhere Genauigkeit erneut bestimmt. Wieder wird von Spursegmenten ausgegangen, die mit einer effektiven Digitalisierung von 20 MHz gefunden wurden. Für diese wird die volle 80 MHz-Information aus den Schieberegistern zusammen mit den $z$-Koordinaten zu dem FPGA an der rückwärtigen Seite der Karte gesendet. Vor der weiteren Verwendung muß zunächst sichergestellt werden, daß auch diese Spursegmente zu einer vom Ursprung ausgehenden Spur gehören. Da durch die feinere Auflösung die Zahl gültiger Masken stark ansteigt, würde eine Implementierung mit CAMs die Kapazitäten der FPGAs überschreiten. Statt dessen werden herkömmliche Speicherbausteine, in Abbildung 3.3 als RAM bezeichnet, als Look-up-table eingesetzt. In einen Teil des Speichers werden vorab berechnete Masken gültiger Spursegmente mit feiner Auflösung geladen, in den anderen Teil die diesen Spursegmenten entsprechenden $\kappa$- und $\varphi$-Werte. In einem zweistufigen Prozeß wird zunächst überprüft, ob das Bitmuster des gefundenen Spursegments einer vorhandenen Maske entspricht. Ist dieses Kriterium erfüllt, werden aus dem Speicher die zugehörigen Spurparameter ausgelesen.

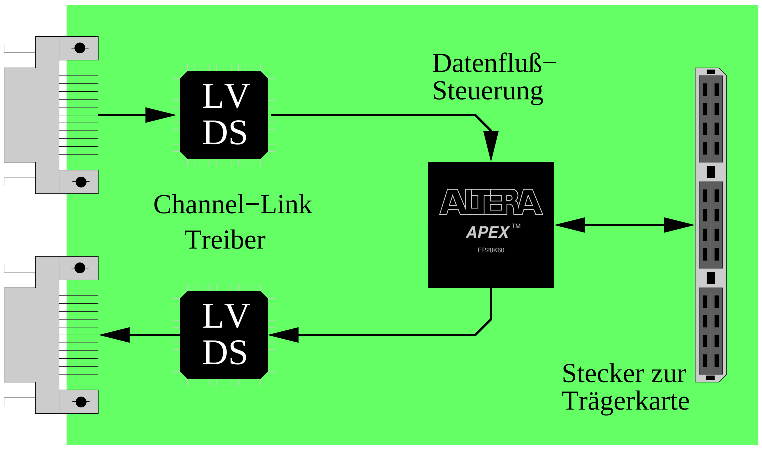



*Abbildung 3.6: Schematische Darstellung der von SCS entwickelten Aufsteckkarte. Gezeigt ist hier die Konfiguration mit je einem Channel-Link-Eingangs- und Ausgangskanal. Alternativ können beide Channel-Link-Kanäle als Eingang geschaltet werden. Ein FPGA steuert den Datenfluß zwischen den beiden Channel-Link-Kanälen und der Trägerkarte, die über einen Stecker mit der Aufsteckkarte verbunden ist.*

Zusammen mit der $z$-Koordinate werden die nun vollständig bestimmten, dreidimensionalen Spursegmentinformationen an die nächste Stufe des FTT geschickt. Die bei diesem Datentransfer eingesetzte Technik wird im nächsten Abschnitt beschrieben.

### 3.2.3 Datentransfer

Das Datentransfersystem des FTTs verbindet hohe Übertragungsraten mit größtmöglicher Flexibilität. Es wird eingesetzt, um zunächst die Daten aller 30 FEMs auf einer Karte zu bündeln, den Datentransfer innerhalb des L2-Systems zu übernehmen und schließlich die Spurdaten zur dritten Stufe zu senden. Für die eigentliche Datenübertragung wird das System "Channel Link" von National Semiconductor [Nat-2002] verwendet, das über 9 parallele LVDS[5]-Verbindungen durch ein Multiplexverfahren 48 Bits breite Datenwörter in Punkt-zu-Punkt-Verbindungen senden kann. Dabei läßt sich bei einer maximalen Taktrate von 112 MHz eine Bandbreite von maximal 5.38 Gbits/s erreichen.

Zur Integration des Channel-Link-Systems in den FTT wurde in Kooperation mit der schweizer Firma SCS[6] eine Aufsteckkarte entwickelt, die über zwei Channel-Link-Anschlüsse und einen Stecker für die Verbindung mit einer Trägerkarte verfügt. Schematisch ist diese Karte in Abbildung 3.6 gezeigt. Zur zentralen Datenflußsteuerung wird ein FPGA verwendet. Beide Channel-Link-Verbindungen können als Dateneingangskanal konfiguriert sein (I/I-Karte), alternativ einer der beiden auch als Ausgang (I/O-Karte). Über den FPGA können Daten aus dem Channel-Link-Datenstrom ausgekoppelt und auf die Trägerkarte gesendet, beziehungsweise von dort eingekoppelt werden.

Für die Belange des FTT wurde ein spezielles, sogenanntes "Message System" ent-

5 **L**ow **V**oltage **D**ifferential **S**ignaling
6 **S**uper **C**omputing **S**ystems, http://www.scs.ch

wickelt. Dabei ist das Ziel jedes Datenwortes in dessen oberen Kennbits kodiert. Entsprechend der Position innerhalb des FTTs wird eine Datenwegtabelle in den FPGA geladen, die alle potentiellen Ziele enthält und hereinkommende Daten dementsprechend weiterleitet. Die konfigurierbaren Tabellen können bei Änderungen der FTT-Komponenten neu geladen werden und bieten somit ein hohes Maß an Flexibilität. Die programmierbaren Komponenten auf den im folgenden beschriebenen Karten der zweiten Stufe unterstützen ebenfalls dieses Datentransferprotokoll, wobei in einem Baustein intern mehrere Ziele implementiert sein können. Insgesamt kann eine Datenwegtabelle daher bis zu 500 Einträge enthalten.

### 3.2.4 Merger-Karte

Die hier und in den Abschnitten 3.2.5 und 3.2.6 beschriebenen Karten sind Bestückungs- und Programmiervarianten eines Grundtyps, der in Abbildung 3.7 gezeigt und als FTT-Mehrzweckkarte oder MPB[7] bezeichnet wird. Ihnen gemein ist, daß alle programmierbaren Komponenten neben der Verbindung über Channel-Link zusätzlich über eine VME-Schnittstelle von außen ansprechbar sind. Detailierte Informationen über Aufbau und Funktion der Karte, die über das in den folgenden Abschnitten beschriebene hinausgehen, können in [Mee-2002] gefunden werden.

Um die Daten von 30 FEMs zu sammeln, werden fünf MPBs als sogenannte Merger-Karten[8] konfiguriert. Diese Karten sind mit der maximal möglichen Zahl von vier Aufsteckkarten bestückt, von denen jeweils zwei als I/I- und zwei als I/O-Karte konfiguriert sind. Damit kann eine Merger-Karte die Daten von 6 FEMs und somit einer CJC1-Triggergruppe oder einer halben CJC2-Triggergruppe empfangen und an zwei Ziele weiterleiten. Synchron zum HERA-Takt werden die groben Spursegmentinformationen zu der L1-Linker-Karte verschickt, nach einer positiven L1-Entscheidung die verfeinerten dreidimensionalen Spursegmentinformationen zur L2-Linker-Karte.

### 3.2.5 Linker-Karte

Die L1- und L2-Linker-Karte[9] sind vom Aufbau her identisch. Die Unterscheidung beruht auf der unterschiedlichen Präzision der Eingangsdaten, der für die Datenverarbeitung zu Verfügung stehenden Zeit und der daraus resultierenden, unterschiedlichen Implementierung des Spurverknüpfungsalgorithmus. Während bei der L1-Karte durch enge Zeitvorgaben nur eine grobe Verknüpfung der Spursegmente stattfinden kann, ist dies auf der L2-Karte mit einem ähnlichen Algorithmus mit feinerer Auflösung möglich. Da die Eingangsdaten für die dritte Stufe des FTT auf der L2-Karte verarbeitet werden, soll nur diese weiter betrachtet werden.

Für jedes Spursegment der vier Triggerlagen wird dessen jeweilige Lage in einem virtuellen $\kappa$-$\varphi$-Histogramm mit $40 \times 640$ Einträgen berechnet. Spursegmente verschiedener Lagen, die zu einer Spur gehören, liegen in diesem Histogramm nahe beieinander. Bei einem Histogramm mit 25600 Einträgen ist Füllen und Suchen sehr zeitaufwendig. Daher werden an dieser Stelle wiederum CAMs verwendet.

[7] **M**ulti **P**urpose **B**oard
[8] to merge = verschmelzen, einfädeln
[9] to link = verknüpfen, verbinden

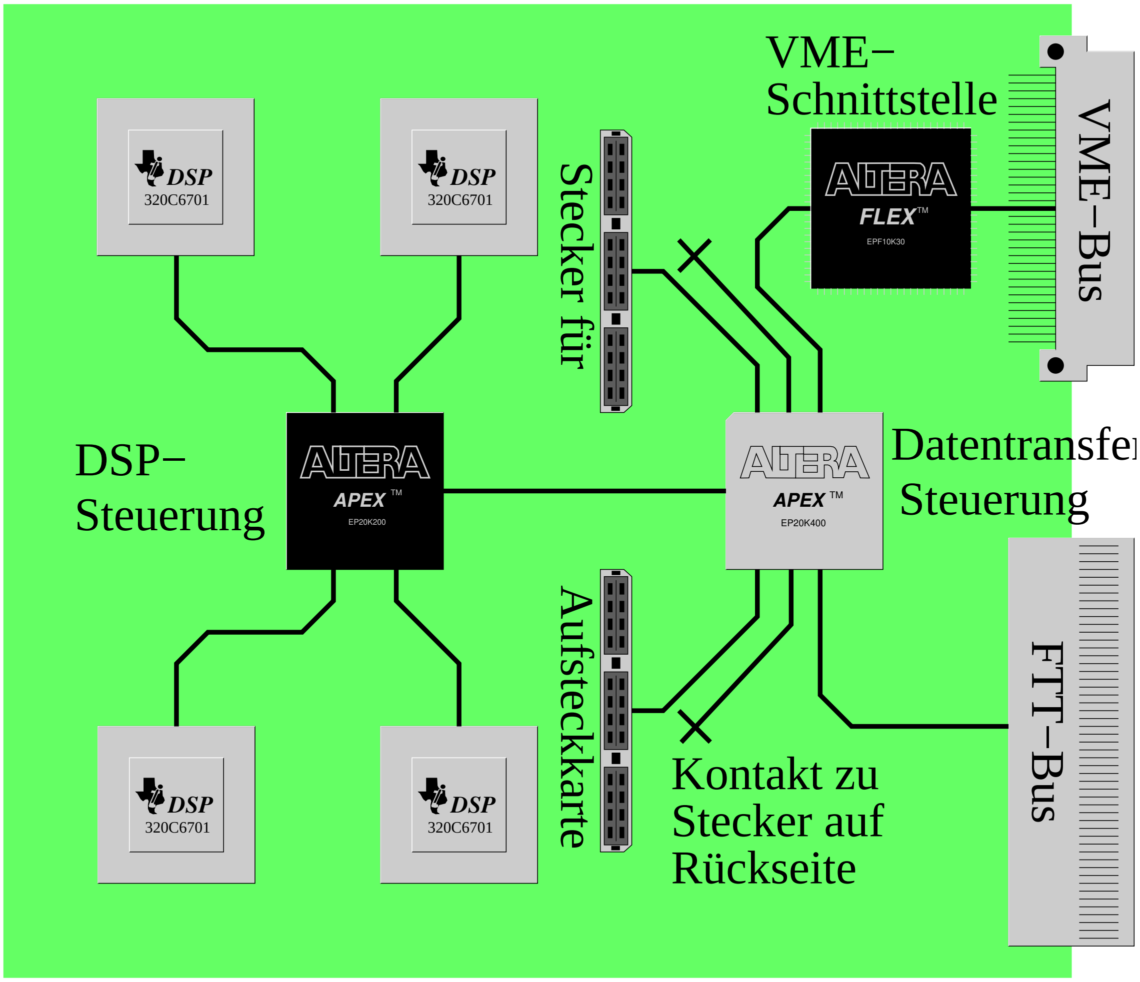



*Abbildung 3.7: Aufbau der FTT-Mehrzweckkarte MPB. Die verschiedenen Konfigurationsmöglichkeiten dieser Karte sind im Text beschrieben.*

Abbildung 3.8 zeigt die Speicherung der L2-Spursegmente auf der Linker-Karte. Neben der eigentlichen Binnummer wird für jedes Spursegment auch eine sogenannte Superbinnummer ermittelt. Diese Struktur setzt sich aus 5×5 Bins zusammen und ist in der Elektronik durch 25 CAMs pro Triggerlage repräsentiert. Insgesamt sind daher 100 CAMs in dem FPGA implementiert, der den Datentransfer kontrolliert und so dimensioniert ist, daß er neben der reinen Datentransfersteuerung noch weitere Aufgaben wahrnehmen kann. Durch den Eintrag der Superbinnummer in den CAM der betrachteten Lage, der der Binnummer innerhalb des Superbins entspricht, ist die Position eines Spursegments im $\kappa$-$\varphi$-Histogramm eindeutig beschrieben. Als Startpunkt für die Suche wird für jedes Spursegment die Superbinnummer entsprechend der jeweiligen Triggerlage in einem von vier RAMs in dem FPGA gespeichert. Aus einem parallelen RAM lassen sich die assoziierten Spurparameter jederzeit wieder auslesen.

Nach aufsteigender Lagenzugehörigkeit werden die Superbinnummern der Startpunkt aus den RAMs ausgelesen und den CAMs präsentiert. Für jeden CAM, der diese

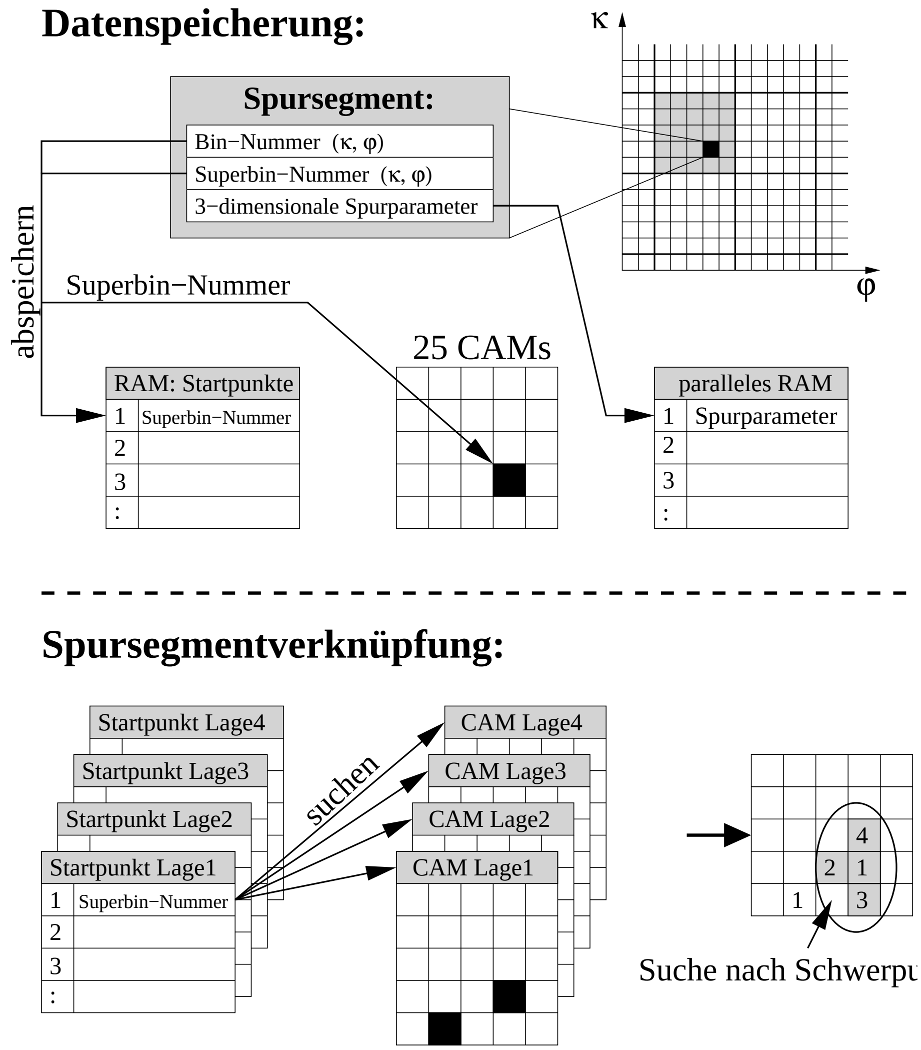



*Abbildung 3.8: Datenverarbeitung auf der L2-Linker-Karte nach [Mee-2002]. Die Kodierung der Spursegmentposition in der κ-φ-Ebene und deren Speicherung, sowie die anschließende Verknüpfung zu ganzen Spuren ist ausführlich im Text beschrieben.*

Superbinnummer enthält, wird entsprechend seiner Position die zugehörige Lagennummer in einem 5×5-Bins großem Array eingetragen. Dadurch steht nach wenigen Taktzyklen eine Projektion aller Triggerlagen in einen kleinen Ausschnitt der $\kappa$-$\varphi$-Ebene zu Verfügung, wie es in Abbildung 3.9 gezeigt ist. Innerhalb dieses Arrays sollen nun in einem 3×3-Bins großen Fenster Spursegmente verschiedener Lagen verknüpft werden, indem jedem Spursegment entsprechend seiner Position im Fenster die in Abbildung 3.9 gezeigten Wichtungsfaktoren zugeordnet und diese anschließend aufsummiert werden. Aufgrund beschränkter Detektorauflösung entstehende Migrationen in benachbarte Bins berücksichtigt man, indem das $3 \times 3$-Fenster überlappend über den $5 \times 5$ Bins großen Ausschnitt verschoben und neunmal die Summe der Wichtungsfaktoren gebildet wird. Die Position des Schiebefensters, an der die Summe der Wichtungsfaktoren ma-

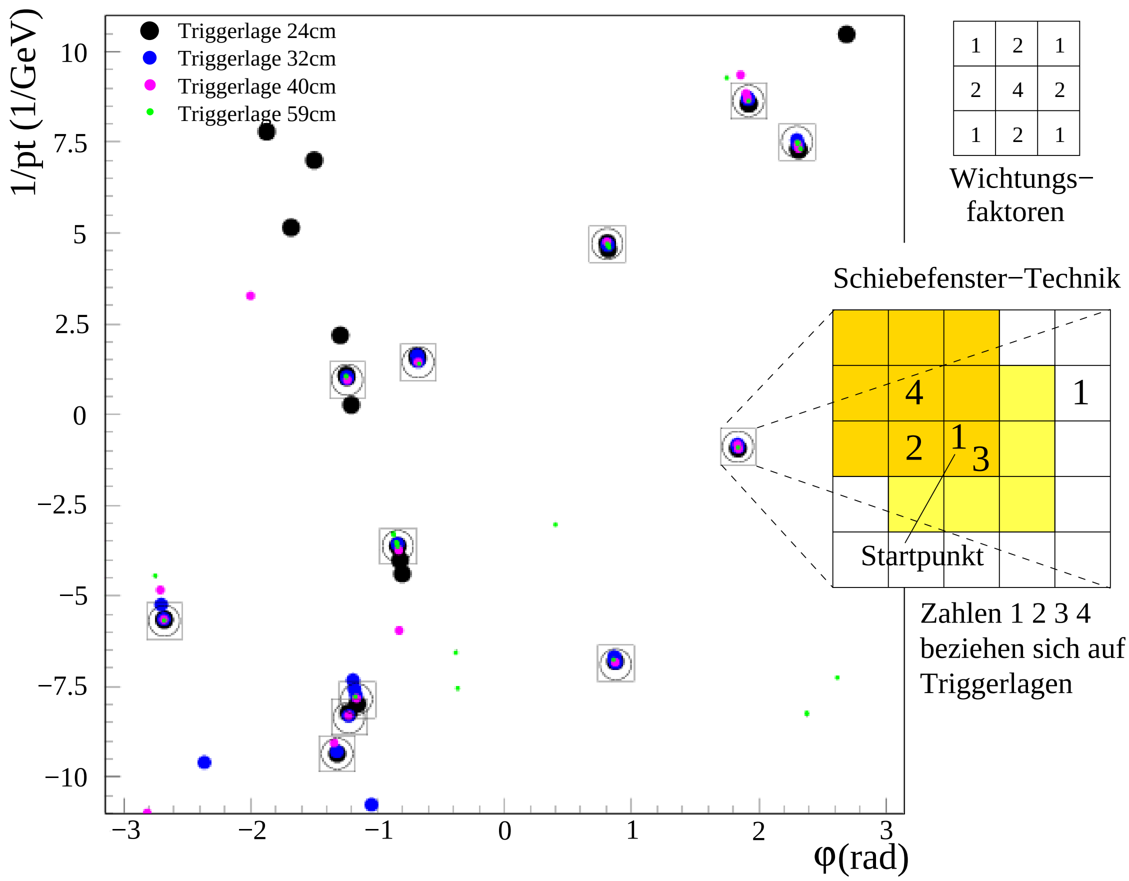



*Abbildung 3.9: Geometrische Veranschaulichung der Spurverknüpfung in der Kappa-Phi-Ebene. Zum einen sind die Einträge für ein Ereignis in der gesamten Ebene gezeigt, zum anderen ein vergrößerter Ausschnitt daraus. Durch eine Schiebefenstertechnik wird in diesem Ausschnitt mit Hilfe von Wichtungsfaktoren nach dem gemeinsamen Schwerpunkt aller Spursegmente bestimmt.*

ximal wird und sich aus mindestens zwei Spursegmenten verschiedener Lagen zusammensetzt, definiert eine gefundene Spur. Eine Markierung verhindert die Benutzung der verwendeten Spursegmente für weitere Spuren. Aus dem parallelen Speicher werden die Spursegment-Parameter für die Spur ausgelesen und an die im folgenden beschriebenen Fitter-Karten weitergeleitet.

### 3.2.6 Fitter-Karte

Über eine Kette von Aufsteckkarten sind die Fitter-Karten[10] untereinander verbunden. Ein Verteilungsalgorithmus sorgt für die gleichmäßige Auslastung der Karten und ihrer Ressourcen.

Die MPBs, die als Fitter-Karten dienen, sind mit der maximalen Zahl von vier möglichen DSPs[11] [Tex-2000] bestückt. Jeder DSP berechnet für ein oder zwei Spuren pro Ereignis deren präzise Spurparameter aus den Spursegmentinformationen. Ein nicht–iterativer Anpassungsalgorithmus für Kreisbahnen [Kar-1991] liefert die Spurparameter in der $r\varphi$-Ebene, $\kappa$ und $\varphi^*$. Eine anschließende lineare Ausgleichsrechnung der $z$-Koordinaten unter Einbeziehung des Ereignisvertex liefert den Kotangens des Winkels $\theta$ als dritte Raumkomponente. Die $z$-Koordinate des Vertex kann dabei zum einen über den L2L3-Bus von dem $z$-Vertex-Trigger bezogen werden, zum anderen haben Untersuchungen gezeigt, daß auch durch FTT-L1 diese schon mit ausreichender Genauigkeit bestimmt werden kann [Ber-2002]. Details zur Programmierung der DSPs sind in [Wis-2002] beschrieben.

Interessant für spätere Überlegungen ist die Bedeutung von $\varphi^*$. Der Anpassungsalgorithmus liefert als Ergebnis zunächst $\sqrt{2} \cdot \cos(\varphi)$ und $\sqrt{2} \cdot \sin(\varphi)$. Der Winkel $\varphi$ selbst wird in den DSPs nicht berechnet, da die Ausführung trigonometrischer Funktionen in diesen sehr zeitaufwendig ist. Statt dessen wird aus den beiden Ausdrücken zunächst der Oktant, in den die Spur zeigt, ermittelt. In jedem Oktanten wird für die weitere Verwendung nun der trigonometrische Ausdruck gewählt, der dort kein Extremum aufweist. Da in diesem Bereich der Betrag der Steigung groß ist, bietet der jeweils gewählte Ausdruck im Vergleich zum anderen das bessere Auflösungsvermögen bezüglich $\varphi$. Abbildung 3.10 zeigt, wie trigonometrischer Ausdruck und Oktanteninformation so zu der Variablen $\varphi^*$ kombiniert werden, daß ein eindeutiger Zusammenhang zwischen dieser und dem wahren Winkel $\varphi$ besteht. Mit einer Abweichung von maximal 5% ist dabei ein linearer Zusammenhang $\varphi = \frac{\pi}{4}\varphi^*$ erfüllt [Sch-2002b].

Ein Vergleich der FTT-Spurparameter mit den durch die vollständige H1-Rekonstruktionskette aus dem vollen CJC-Datensatz gewonnenen ist in Abbildung 3.11 gezeigt. In der $\kappa$-$\varphi$-Ebene werden Auflösungen von $\sigma(1/p_t) = 0,03\ \mathrm{GeV}^{-1}$ und $\sigma(\varphi) = 4,6$ mrad erreicht. Paßt man die $\theta$-Verteilung durch die Summe zweier Gauß-Verteilungen an, so ergibt sich eine Auflösung von $\sigma(\theta) \approx 50$ mrad.

### 3.2.7 L2-Entscheidungskarte

Über die Kette der Aufsteckkarten werden die berechneten Spurparameter an die L2-Entscheidungskarte weitergeleitet, die mit den drei möglichen FPGAs bestückt ist. In dem FPGA zur Datensteuerung werden die Spurparameter zum einen für die Weiterleitung zur dritten Stufe gesammelt, zum anderen werden Informationen für eine L2-Triggerentscheidung extrahiert. Der verfügbare Zeitrahmen erlaubt es, neben Ereignistopologien auch Impulssummen und für Ereignisse mit wenigen Spuren sogar invariante Massen von Spurpaaren zu berechnen. Im Falle einer positiven Level-2-Entscheidung des

[10] to fit = anpassen

[11] **D**igital **S**ignal **P**rocessor, hier Texas Instruments TMS320C6701 Floating Point DSPs

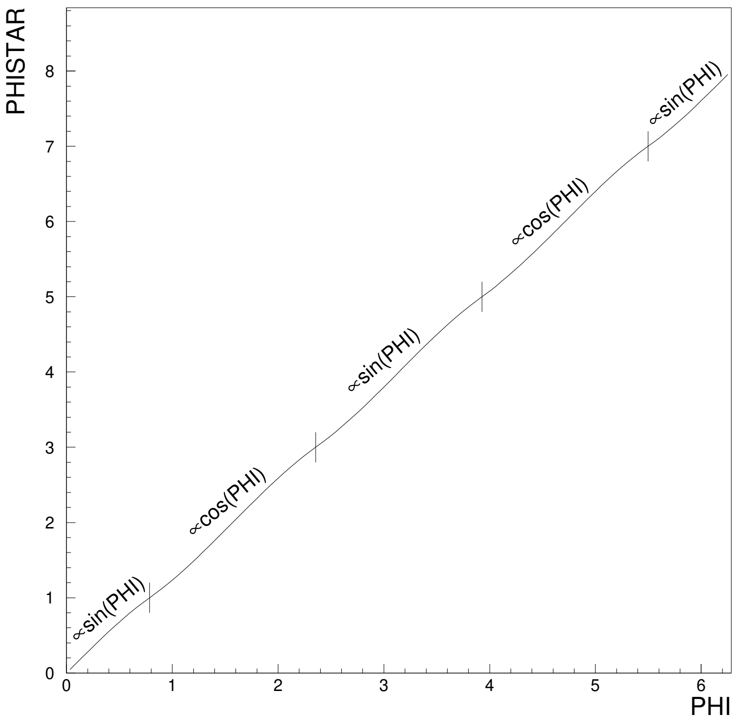


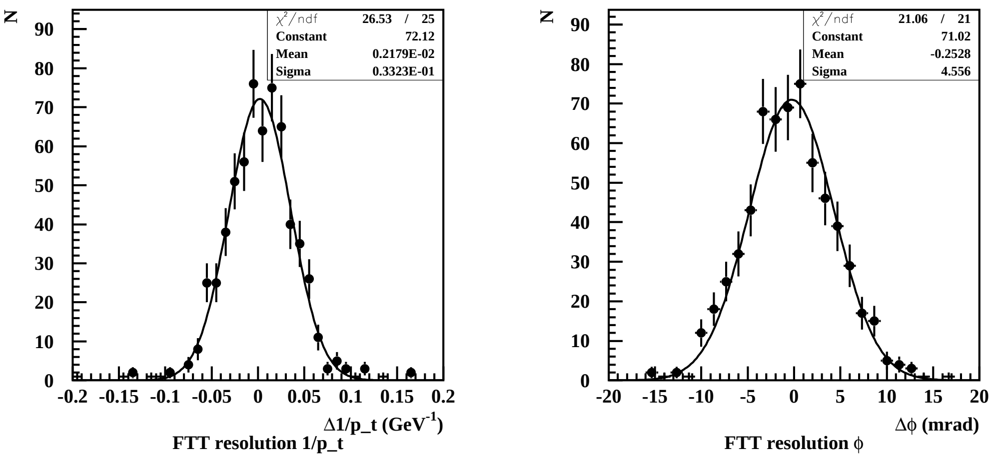

*Abbildung 3.10: Zusammenhang zwischen $\varphi$ und $\varphi^*$.*

zentralen Triggers werden die gefundenen Spuren schließlich über eine Channel-Link-Verbindung an die dritte Stufe gesendet.

### 3.2.8 Prozessorkarte

Neben speziell für Dateneingang und Kommunikation mit dem zentralen Trigger entwickelten Karten kommen bei der dritten Stufe des FTT ausschließlich kommerziell erhältliche Komponenten zum Einsatz. Kernstück ist eine Gruppe von Prozessorkarten. Auf jeder Karte soll der Selektionsalgorithmus für einen ausgewählten Teilchenzerfall oder eine ausgewählte Ereignissignatur implementiert werden. Ein Echtzeitbetriebssystem erlaubt schnelle Reaktionszeiten und die Abarbeitung der Algorithmen in etwa 100 $\mu$s. Es stellt außerdem die Plattform für die einfache Integration von durch Dritte entwickelte Selektionsprogramme bereit. Im folgenden Kapitel schließt sich eine ausführliche Darstellung des Aufbaus dieser Stufe an.

Zusätzlich werden die Prozessorkarten im gesamten FTT für dessen Datenauslese verwendet. Informationen hierzu finden sich im Anhang A.



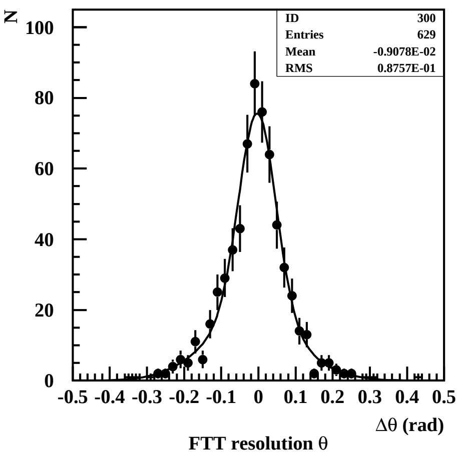




*Abbildung 3.11: Präzision der durch den FTT ermittelten Spurparameter im Vergleich zur Rekonstruktion durch die volle H1-Analysekette [H1-1999c]. Dargestellt sind die Abweichungen für $\kappa \propto 1/p_t$, $\varphi$ und $\theta$. Die Resultate des FTTs wurden mit Hilfe von FTTEMU simuliert.*

# Kapitel 4

# Dritte Stufe des Fast Track Triggers

Zunächst soll in diesem Kapitel das generelle Konzept vorgestellt werden, das der dritten Stufe des FTT (FTT-L3) zugrunde liegt. Die sich daraus ableitende Auswahl elektronischer Komponenten zur Umsetzung des Konzepts wird mit Angabe der bei der Entscheidungsfindung betrachteten Alternativen nachfolgend erläutert. Ausführlich wird anschließend die Implementierung von Algorithmen zur Datentransfersteuerung und Datenselektion beschrieben. Abschließend werden die Ergebnisse einiger Systemtests präsentiert.

## 4.1 Konzept

Primäre Aufgabe von FTT-L3 ist es, Teilchenzerfälle durch ihre charakteristische invariante Masse mit Hilfe der Spurparameter der zweiten Stufe zu identifizieren. Zusätzlich wurde im Laufe der Entwicklung klar, daß häufig ausreichend Rechenzeit zu Verfügung steht, weitere Daten des L2L3-Busses verarbeiten zu können. Generell ist für jede zu untersuchende physikalische Signatur eine Prozessorkarte (CPU[1]-Karte) vorzusehen, die ein Triggerelement (TE) an die zentrale H1-Triggerlogik CTL liefert. Allerdings können zum Teil auch schon aus Zwischenergebnissen weitere TEs extrahiert werden. Ergänzend muß eine Infrastruktur geschaffen werden, die die CPU-Karten in die FTT-Datenübertragungs- , Auslese- und Kontrollsysteme einbindet, sowie Verbindungen zur CTL zum Empfangen von Kontrollsignalen und Senden der TEs bereitstellt. Hauptaugenmerk muß dabei auf eine parallele Übertragung der Eingangsdaten zu den CPU-Karten gerichtet sein, um zu vermeiden, daß der Zeitbedarf während dieser zeitkritischen Phase mit der Zahl der eingebauten CPU-Karten skaliert.

Das daraus resultierende Gesamtsystem ist im Vorgriff in Abbildung 4.1 gezeigt und soll in den folgenden Abschnitten detailliert vorgestellt werden. Neben den Rechenkarten für die Triggeralgorithmen ist eine weitere CPU-Karte als Kontrollkarte zur Steuerung der dritten Stufe vorgesehen. Durch selbst entwickelte Karten werden zum einen die Eingangsdaten und ausgewählte STC-Signale empfangen, zum anderen die TEs in Form von sogenannten Trigger-Bits zur CTL gesendet. Auf die in Abbildung 4.1 als VIC bezeichnete Karte soll in diesem Kapitel nicht näher eingegangen werden, da sie nicht direkter Teil der dritten Stufe ist, sondern als Schnittstelle zu der in Anhang A beschriebenen FTT-Datenauslese dient.

Auf den Rechenkarten sollen letztendlich die Selektionsalgorithmen für bestimmte Ereignisignaturen implementiert werden, wie beispielsweise für den Zerfall

$$D^{+*} \to D^0 \pi^+_{slow} \to K^- \pi^+ \pi^+_{slow}. \tag{4.1}$$

[1] **C**entral **P**rozessing **U**nit

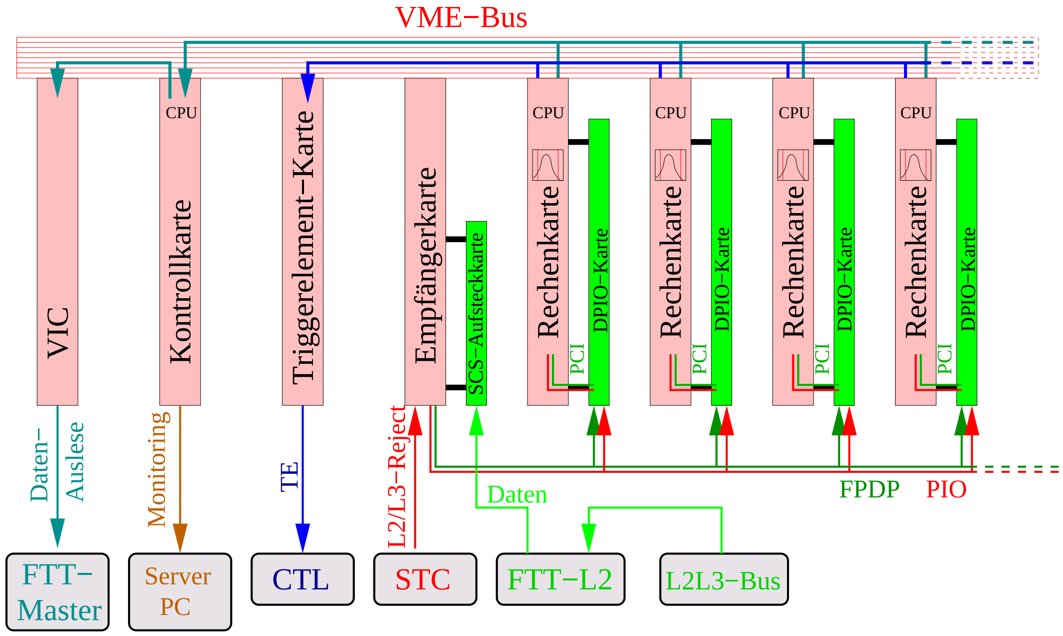



*Abbildung 4.1: Übersicht über die elektronischen Komponenten und Signallaufwege der dritten Stufe des FTT. Neben einer Gruppe von Prozessorkarten wird ergänzende Infrastruktur als Schnittstelle zu anderen Komponenten des FTT und zur CTL benötigt. Funktion und Realisierung der einzelnen Komponenten und Datenübertragungswege sind im Text beschrieben.*

Ereignisse die diesen Zerfall aufweisen, werden typischerweise in zwei Stufen selektiert. Zunächst wird versucht, den $D^0$-Zerfall durch die Berechnung der invarianten Masse[2]

$$\begin{aligned} M_{K\pi} &= \sqrt{(E_K + E_\pi)^2 - (\vec{p}_K + \vec{p}_\pi)^2} \\ &= \sqrt{m_K^2 + m_\pi^2 + 2E_K E_\pi - 2\vec{p}_K \vec{p}_\pi} \end{aligned} \tag{4.2}$$

zu identifizieren, indem man je zwei rekonstruierte Spuren hypothetisch einem Kaon und einem Pion zuordnet und eine kleine Differenz zwischen berechneter ($M_{K\pi}$) und theoretischer $D^0$-Masse ($m_{D^0} = 1864,5$ MeV [Hag-2002]) fordert:

$$|M_{K\pi} - m_{D^0}| < \delta_{D^0}. \tag{4.3}$$

Ein typischer Wert in H1-Analysen für die maximale Differenz ist 80 MeV. Spurpaaren, die dieses Kriterium erfüllen, wird eine dritte Spur mit der Annahme, daß es sich um ein Pion handelt, zugeordnet und die invariante Masse $M_{K\pi\pi_{slow}}$ dieses Systems berechnet. Indem man für die gemessene Massendifferenz

$$\Delta_M = M_{K\pi\pi_{slow}} - M_{K\pi} \tag{4.4}$$

[2] Invariante Massen von Mehrteilchensystemen werden im folgenden immer durch $M$, Ruhemassen einzelner Teilchen durch $m$ bezeichnet

eine obere Schranke nahe des erwarteten Wertes von $m_{D^*} - m_{D^0} = 145,5$ MeV mit $m_{D^*} = 2010$ MeV [Hag-2002] fordert, kann schließlich ein möglicher $D^*$-Zerfall selektiert werden. Da $\Delta_M$ klein ist, steht für den Zerfall $D^* \rightarrow D^0\pi_{slow}$ nur wenig Phasenraum zu Verfügung und der Transversalimpuls des Pions beträgt nur wenigen 100 MeV. Dies stellt große Anforderungen an die Rekonstruktion von niederenergetischen Spuren mit starker Krümmung durch den FTT.

Die Implementierung der Selektionsalgorithmen auf den Rechenkarten wird in Abschnitt 4.3.2 allgemeingültig beschrieben. In Abschnitt 4.4.3 wird auf die Wahl der Parametergrenzen bei speziellen Teilchenselektionen eingegangen.

## 4.2 Elektronische Komponenten

Zunächst sollen die Eigenschaften der einzelnen Komponenten sowie die Kriterien bei ihrer Auswahl beschrieben werden. Teilweise beeinflußte die Wahl einer Komponente auch die Entscheidungsfindung bei einer anderen. Daher werden zunächst die Komponenten beschrieben werden, deren Auswahl bei Entscheidungen bezüglich anderer Komponenten von Bedeutung war.

### 4.2.1 Lokaler Datentransfer

Komplexe Systeme, die aus mehreren Elektronikkarten aufgebaut sind, werden häufig in genormten Gehäusen mit einer durchgehenden Rückwandplatine ("Backplane") untergebracht. In diesen Platinen ist zum einen die zentrale Spannungsversorgung, zum anderen meistens auch ein durchgehender Datenbus zur Kommunikation der verschiedenen Karten untereinander implementiert. Zusätzlich existieren weitere, nicht an die Rückwandplatine gebundene Lösungen zum Datentransfer innerhalb eines Gehäuses und auch über dessen Grenzen hinaus.

#### 4.2.1.1 VME-Bus

Bei der Integration von CPU-Karten in Elektroniksysteme kommen hauptsächlich zwei standardisierte Rückwandsysteme zum Einsatz: VME[3]-Bus [ANS-1987, ANS-1994a] und cPCI[4]-Bus [PIC-1997], die jeweils einen Punkt-zu-Punkt-Datentransfer zwischen zwei Karten unterstützen.

Der VME-Standard wurde in den 70er Jahren durch Motorola aus mehreren Vorläufersystemen entwickelt. Der Datentransfer ist nicht an eine starre Zeitvorgabe gekoppelt, sondern wird durch die Synchronisation der beiden an diesem Transfer beteiligten Karten gesteuert. Bei einer ursprünglichen Datenbusbreite von 16 Bits konnten Übertragungsraten von etwa 10 MBytes/s erreicht werden. Durch zahlreiche Erweiterungen des Standards bezüglich der verwendeten Rückwandplatine und des Transferprotokolls liegt die typische Übertragungsrate inzwischen bei 40–80 MBytes/s, bei speziellen Konfigurationen können sogar Spitzenwerte von 320 MBytes/s erreicht werden [Ryn-2002]. Damit ist der VME-Bus nachwievor zu dem unten beschriebenen cPCI-Bus konkurenzfähig. Für

[3] **V**ERSA**m**odule **E**urocard
[4] **C**ompact **PCI**= **C**ompact **P**eripheral **C**omponent **I**nterconnect

die Elektronik des H1-Experiments wurde der VME-Bus zu Beginn als Quasi-Standard festgelegt. So sind zum Beispiel die Datenauslese und Signalverbindungen zur CTL über VME-Karten realisiert.

Da der VME-Bus vor den Erweiterungen für zahlreiche Anwendungen den Engpaß in der Datentransferkette darstellte, wurde mit dem cPCI-Bus ein weiteres Bussystem für Rückwandplatinen entwickelt. Ausgangspunkt war dabei der PCI-Bus[5] [PCI-1995], der auf CPU-Platinen zur Anbindung von Peripheriegeräten Verwendung findet. Dieser erreicht in seiner Grundkonfiguration bei einer Datenbreite von 32 Bits und einer Taktrate von 33 MHz eine Übertragungsrate von 132 MBytes/s. Die Spezifikationen sehen zwar als Möglichkeit eine Verdopplung sowohl der Datenbreite als auch der Übertragungsrate vor, dies setzt sich aber erst langsam durch.

Beiden Bussystemen ist gemein, daß sie neben dem reinen Datentransfer und der Spannungsversorgung noch weitere Funktionalitäten bieten. So sind zum Beispiel Interrupt[6]-Strukturen implementiert, die es erlauben, an Karten Signale mit verschieden hohen Prioritäten zu senden. Trotz der in der Grundkonfiguration höheren Leistungsfähigkeit konnte sich der cPCI-Bus in vielen Anwendungsbereichen bisher nicht gegenüber dem VME-Bus durchsetzen. Dazu haben die über lange Zeit gesammelten Erfahrungen mit letzterem und die daraus abgeleiteten Erweiterungen beigetragen. Um auf die langjährige Erfahrung bei H1 zurückgreifen zu können und die Kompatibilität zu zentralen Systemen dieses Experiments zu garantieren, wurde auch die Elektronik für FTT-L3 auf Basis eines VME-Busses entwickelt. Die grundlegende Funktionalität des Busses soll im folgenden kurz skizziert werden.

Für den VME-Bus sind zwei Subbusse mit jeweils 96 Leitungen auf der Rückwandplatine vorgesehen, mit denen an 21 Steckplätzen über zwei dreireihige Stecker Karten verbunden werden können. Für den unteren Subbus ist allerdings nur die Pinbelegung der mittleren Reihe durch die Spezifikation vorgegeben. Die Nutzung der beiden anderen Reihen bleibt dem Anwender überlassen. Für eine leicht eingeschränkte Nutzung des Busses reicht allein der obere Subbus aus. Dieser inkorporiert alle Kontrollsignale, die einer Karte die Kontrolle über den Datentransfer zuweisen und diesen dann steuern. Für den Datentransfer sind auf dem oberen Subbus 24 Adreß- und 16 Datenleitungen vorgesehen. Durch den unteren Subbus werden beide Kanäle auf 32 Leitungen erweitert. Zur Adressierung stehen drei separate Adreßräume A16, A24 und A32 (die Zahl bezeichnet jeweils die Anzahl der verwendeten Adressleitungen) zur Verfügung. Auf jeder Karte kann in einer VME-Kontrolleinheit konfiguriert werden, in welchem Adreßbereich und in welchem Adreßraum lokaler Speicher zum Lesen oder Beschreiben durch den Bus freigegeben wird. Es ist Aufgabe des Programmierers, Adreßbereiche mehrerer Karten in einem Adreßraum nicht überschneiden zu lassen. Die Signalabfolge bei Schreib- und Lesezugriffe ist in [ANS-1987] definiert. Durch Kontrolleitungen wird die Zahl gültiger Bits im Datenwort angezeigt. Daraus ergeben sich die Übertragungsarten D08, D16, D32 und im Blockmodus auch D64. Bei dem sogenannten Block-Modus wird nur vorab eine Adresse übertragen, alle folgenden Datenworte werden als aufsteigend im Speicher sortiert interpretiert. Dadurch kann für große, zusammenhängende Datenblöcke die Übertragungsrate verdoppelt werden.

[5] **P**eripheral **C**omponent **I**nterconnect
[6] to interrupt = unterbrechen

Um einer Karte ein Signal zukommen zu lassen, sind sieben Interruptleitungen mit verschiedenen Prioritäten in den oberen Subbus integriert. Der VME-Bus muß so konfiguriert sein, daß pro Priorität nur eine Karte auf den Interrupt reagieren darf. Üblicherweise unterbricht diese Karte daraufhin alle laufenden Aktivitäten und verarbeiten zunächst den Interrupt. Durch einen sogenannten Interruptvektor, den die den Interrupt auslösende Karte zusätzlich über die Datenleitungen sendet, werden der Empfängerkarte weitere Informationen mitgeteilt, die unterschiedliche Aktivitäten auf dieser Karte auslösen können.

Der VME-Bus wird bei FTT-L3 dazu benutzt, die weiter unten besprochenen, selbst entwickelten Karten zu konfigurieren, sowie Daten aus den Rechenkarten auszulesen und der Kontrollkarte zur weiteren Verarbeitung zur Verfügung zu stellen. Zwischen den CPU-Karten werden dabei A32/D32-Zyklen, zum Teil auch im Blockmodus, verwendet. Die selbst entwickelten Karten werden durch A24/D16-Zyklen angesprochen. Durch Interrupts werden der L3-Kontrollkarte Entscheidungen der CTL übermittelt.

#### 4.2.1.2 Front Panel Data Port

Da durch den VME-Bus nur eine Punkt-zu-Punkt-Übertragung möglich ist, können die Eingangsdaten nicht durch diesen übertragen werden, ohne daß die Dauer der Datenübertragung von der Anzahl der Rechenkarten abhängt. Zunächst wurde in Betracht gezogen, jeder Rechenkarte eine Empfängerkarte zuzuordnen, die untereinander analog zu FTT-L2 durch "Channel-Link" verbunden sind und die Daten für die jeweilige Prozessorkarte aus dem Datenstrom in lokalen Speicher kopieren. Damit wären allerdings viele Steckplätze des VME-Busses belegt worden und hätten somit nicht für Rechenkarten zu Verfügung gestanden. Außerdem wäre noch die weitere Übertragung zu der Rechenkarten zu lösen.

Um die Möglichkeit zur Integration von zusätzlichen Datentransferwegen zu CPU-Karten zu haben, sind in deren lokalem PCI-Bus häufig Steckplätze für kleine, standardisierte [IEE-1995b, VIT-1999, ANS-1995] Aufsteckkarten[7] vorgesehen. Der Anbindung der Resourcen dienen dabei entweder die beiden unbenutzen Reihen des unteren VME-Busses, die über direkte Signalleitungen mit der Aufsteckkarte verbunden sind, oder Kabel, die vorne aus der CPU-Karte herausgeführt werden. Die Entwicklung einer solchen Karte gestaltet sich sehr aufwendig, da schnelle Schaltungen zum Empfang der Daten und als Schnittstelle zum PCI-Bus auf kleinem Raum untergebracht werden müssen. Insofern wurde die Entwicklung einer eigenen Aufsteckkarte in Abwandlung der SCS-Aufsteckkarten verworfen. Unter Einbeziehung des ersten Konzepts wurde dann die Möglichkeit untersucht, eine Empfängerkarte zu bauen, die einen Steckplatz für eine SCS-Aufsteckkarte zum Empfang der Daten aufweist. Der Datenstrom sollte allerdings auf der Karte vervielfacht und über Schnittstellen zu einem Bus, der eine Punkt-zu-Punkt-Verbindung zu kommerziellen Aufsteckkarten herstellt, auf die Rechenkarten geleitet werden. Damit wäre eine Empfängerkarte in der Lage, mehrere Rechenkarten mit Daten zu versorgen. Bei der Suche nach geeigneten Aufsteckkarten und Busprotokollen wurden auch zwei gefunden, die das Senden an mehrere Empfänger gleichzeitig ermöglichen, so daß auf der Empfängerkarte die Sendeelektronik nur einmal vorhanden sein muß.

[7] häufig als Piggy-Back- oder Mezzanine-Karten bezeichnet

Kernstück des RACEway-Standards [ANS-1994b] sind Platinen, die auf vier benachbarte untere Stecker auf der Rückseite der Rückwandplatine aufgesteckt werden und die durch den VME-Bus ungenutzten Pins elektrisch verbinden. Eine zwischengeschaltete Logik steuert den Datentransfer zwischen den vier Steckern und weiteren dieser Platinen, die an anderen Stellen des Busses aufgesteckt werden können. Aufsteckkarten nach dem PMC[8]-Standard [VIT-1999] haben Zugriff auf die entsprechenden Pins und schaffen eine Verbindung zum PCI-Bus der CPU-Karte. Über eine Platine können gleichzeitig bis zu vier Datenübertragungen mit einer Breite von 32 Bits und einer maximalen Rate von 267 MBytes/s stattfinden. Außerdem sollte es mit dieser Rate möglich sein, von einem Sender an sieben Empfänger parallel die gleichen Daten zu schreiben. Die große Flexibilität des Systems bedingt allerdings ein komplexes Datentransferprotokoll. Nachteilig ist, daß dabei keine Interruptstruktur vorgesehen ist.

Ein wesentlich einfacheres Protokol ist durch den FPDP[9]-Standard [ANS-1998] festgelegt, der den Transfer von 32 Bit breiten Datenwörtern durch einen festen Sender an einen oder mehrere Empfänger über ein gemeinsames, 80-poliges Flachbandkabel definiert. Die Synchronisation erfolgt durch ein Taktsignal des Senders, gültige Datenworte werden durch ein weiteres, paralleles Signal **DVALID** gekennzeichnet. Da keine Adreßinformationen vorgesehen sind, muß dem Empfänger ein Schema zur weiteren Datenbehandlung bekannt sein. Ist ein Empfänger nicht mehr in der Lage, weitere Daten zu verarbeiten, kann dieser durch das Signal **SUSPEND** die Datenübertragung unterbrechen. Durch den Sender können weitere Signale *RES1–3*[10] zur Gliederung des Datentransfers in Blöcke verwendet werden. Der Standard sieht außerdem zwei Signalleitungen *PIO* vor, die völlig frei konfigurierbar und somit für verschiedene Zwecke nutzbar sind. Typische Übertragungsraten über den FPDP-Bus liegen bei 80 MBytes/s.

#### 4.2.1.3 DPIO-Karte

Die in Abbildung 4.2 gezeigte DPIO[11]-Karte der Firma VMETRO war zum Zeitpunkt der Entscheidungsfindung europaweit die einzige, die den recht neuen FPDP-Standard unterstützte. Über einen PMC-Stecker wird sie mit dem lokalen PCI-Bus einer CPU-Karte verbunden. Neben dem reinen Datentransfer ist sie durch ihren PCI-Kontrollbaustein in der Lage, Interrupts auf dem PCI-Bus auszulösen. Die Karte ist in zwei Varianten als Sender- oder Empfängerkarte erhältlich und für eine Datenbreite von 32 Bits bei einer maximal möglichen Transferrate von 25 MHz ausgelegt. Das aufsteckbare Eingangsmodul ist neben der gewählten Konfiguration für FPDP mit TTL[12]-Signalen, die die Transferrate auf 20 MHz limitiert, auch für andere Signalstandards erhältlich. Das Eingangsmodul der jeweils letzten Empfängerkarte eines FPDP-Busses muß eine geänderte Bestückung mit Widerständen aufweisen, da durch sie der Bus elektrisch zu terminieren ist.

Um die Datentransferzyklen des FPDP- und des PCI-Busses zu entkoppeln, werden die Daten zunächst in FIFOs geschrieben. Ein durch die Prozessorkarte konfigurierbarer

8 **P**CI **M**ezzanine **C**ard
9 **F**ront **P**anel **D**ata **P**ort
10 **Re**served **S**ignal
11 **D**igital **P**arallel **I**nput **O**utput
12 **T**ransistor-**T**ransistor-**L**ogik

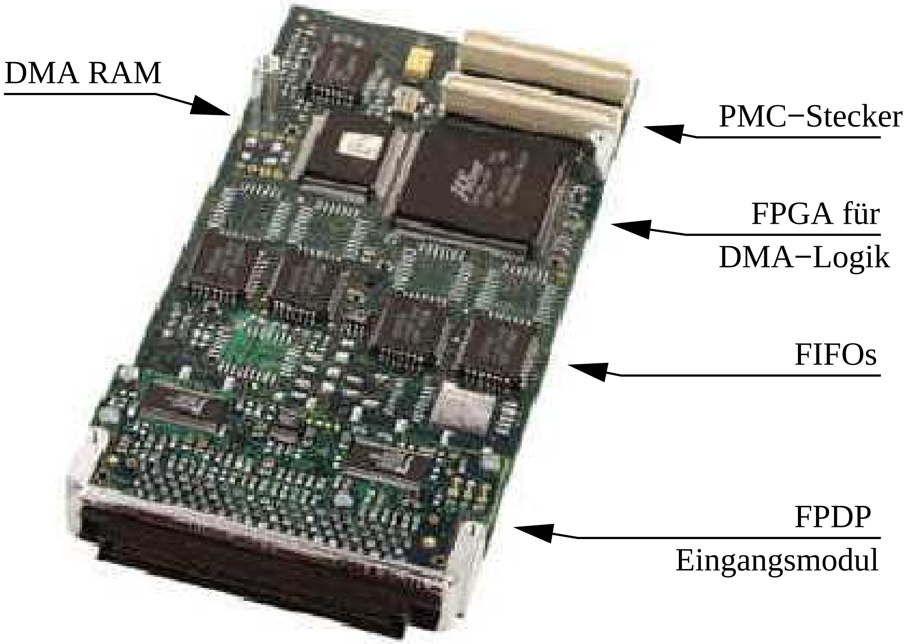



*Abbildung 4.2: DPIO-Karte der Firma VMETRO [VME-2000]. Der 80-polige Stecker für den FPDP-Datentransfer befindet sich auf diesem Bild unten. Die Daten werden mit der Frequenz des FPDP-Busses in FIFOs geschrieben, um von dort durch einen als PCI-Kontrollbaustein programmierten FPGA mit der vom PCI-Bus vorgegebenen Rate wieder ausgelesen zu werden. Die Auslese erfolgt entweder wortweise durch einzelne PCI-Lesezugriffe oder durch DMA-Datentransfer. Um hierfür Speicherzeiger und Datenblocklängen für den PCI-Kontrollbaustein bereitzustellen, ist dem FPGA ein programmierbarer RAM-Baustein (DMA-RAM) angeschlossen. Der elektronische Kontakt zum PCI-Bus der Trägerkarte wird durch einen Stecker nach der PMC-Norm hergestellt.*

FPGA steuert den weiteren Datentransfer. Zum einen ist es möglich, durch einen von der CPU initiierten PCI-Lesezugriff jeweils ein Wort aus den parallelen FIFOs auszulesen. Desweiteren unterstützt die DPIO-Karte für zusammenhängende Datenblöcke DMA[13]-Datentransfer, der ein direktes Beschreiben des Speichers der CPU-Karte unter Umgehung von dessen CPU erlaubt. Wie dies auf den Rechenkarten implementiert ist, wird in Abschnitt 4.2.2 beschrieben. Auf der DPIO-Karte ist zu diesem Zweck ein Speicherbaustein (DMA-RAM) integriert, in dem vor Beginn der Datenübertragung Zeiger in einen reservierten Speicherbereich der CPU-Karte und die Länge der zu übertragenden Datenblöcke gespeichert werden. Durch einen Befehl kann bei einem nicht leeren FIFO der Datentransfer eines DMA-Blocks gestartet werden. Nach Abarbeitung eines Blocks kann ein Interrupt auf dem PCI-Bus ausgelöst und auch automatisch die Abarbeitung eines weiteren Blocks begonnen werden. Ein DMA-Transfer wird beendet, nachdem der letzten

[13] **D**irect **M**emory **A**ccess

Datenblock einer Reihe abgearbeitet wurde oder indem mit einem Datenwort ein **EOT**[14]-Signal aktiviert wird, für das eine der beiden *PIO*-Leitungen oder die *RES1*- oder *RES2*-Leitung entsprechend konfiguriert werden muß. Generell können Interrupts zusätzlich durch Erreichen verschiedener Füllungsgrade der FIFOs (leer / fast leer / halbvoll / fast voll / voll) und asynchron zum Datentransfer durch die beiden *PIO*-Leitungen, wenn sie nicht als **EOT**-Signal konfiguriert sind, ausgelöst werden.

Für FTT-L3 wurden DPIO-Empfängerkarten mit einer FIFO-Tiefe von 8000 Wörtern gewählt. Dies ist die kleinstmögliche Tiefe, die im Vergleich zur maximal erwarteten Datenmenge von etwa 700 Wörtern (inklusive aller L2L3-Daten) je Ereignis zu sehen ist. Es sind maximal acht Empfängerkarten in einem FPDP-Bus vorgesehen. Für die Testphase wurde zusätzlich eine Senderkarte erworben. Treibersoftware zur Konfiguration und zum Betrieb der Karten wird vom Hersteller für die Betriebssysteme WINDOWS NT und VxWorks angeboten.

### 4.2.2 Prozessorkarte

Aus den bisher beschriebenen Entscheidungen resultiert, daß eine VME-Prozessorkarte mit mindestens einem PMC-Steckplatz benötigt wird. Um die Suche in dem umfangreichen Marktangebot einzuschränken, wurde zunächst der gewünschte Prozessortyp festgelegt. Bisher kamen bei H1 hauptsächlich CPUs von Motorola zum Einsatz, die den Typenfamilien 29k, 68k und PowerPC[15] zuzuordnen sind. Alternativ finden sich auch zahlreiche Karten auf dem Markt, die mit einer Intel-Pentium- oder dazu kompatiblen CPU ausgestattet sind, wie sie in normalen PCs eingebaut sind. Neben der Rechengeschwindigkeit waren auch Preis und Betriebssicherheit Kriterien bei der Auswahl. Die endgültige Entscheidung zwischen einer Intel-Pentium- oder PowerPC-Karte wurde im Herbst 2000 getroffen.

Als Kriterium zur Bestimmung der Rechengeschwindigkeit werden Vergleichstests durchgeführt, in denen die Prozessoren eine Reihe von Programmen bearbeiten müssen. Unter Prozessorherstellern ist es üblich, die Leistung ihrer CPU-Karten durch den SPEC[16]-Test messen zu lassen. Die Kriterien, die bei der Auswahl der abzuarbeitenden Programme eine Rolle spielen, sind zum Beispiel in [Dow-1993] beschrieben. Sie sind so ausgewählt, daß in das Resultat die Leistung des Prozessors, der angeschlossenen Busse und Speicher und des Compilers eingeht. Um differenziertere Aussagen machen zu können, ist der Test in zwei Untergruppen eingeteilt: Die eine Gruppe von Testprogrammen führt nur Operationen mit ganzzahligen Werten durch (int), während bei der anderen Gruppe auch Gleitkommaoperationen durchgeführt werden (fp). Bei der Auswahl der CPU-Karten wurde der Test SPEC95 berücksichtigt [SPE-2000], der im Laufe des Jahres 2000 durch einen neu zusammengestellten Test SPEC2000 [SPE-2002] abgelöst wurde. Dieser betont weniger stark die Einflüße der angeschlossenen Busse und insbesondere derer Taktfrequenzen. Der SPEC95-Test hat dadurch jedoch nicht an genereller Aussagekraft eingebüßt und wird nachwievor für neue CPUs durchgeführt. Resultate für Ganz- und Gleitkommazahltests sind für verschiedene Prozessortypen in den Abbildungen 4.3 und 4.4 zusammengefaßt. Diese Abbildungen schließen auch eine Reihe von Prozessor-

[14] **E**nd **O**f **T**ransfer
[15] **P**erformance **O**ptimized **W**ith **E**nhanced **R**ISC microprocessor architecture for **P**ersonal **C**omputers
[16] **S**tandard **P**erformance **E**valuation **C**orporation

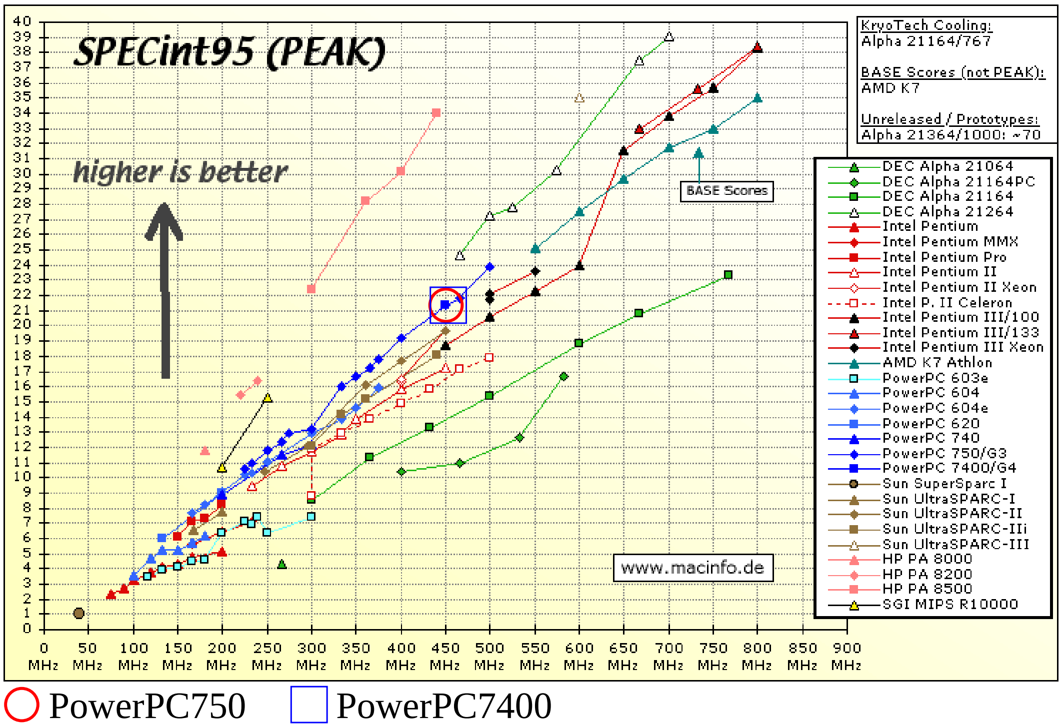



*Abbildung 4.3: SPEC95-Test für verschiedene Prozessoren mit Testprogrammen, die ausschließlich Operationen mit ganzzahligen Werten ausführen [Mac-2002]. Eingekreist ist der Wert für den Prozessor PowerPC 750. Der Nachfolgetyp PowerPC 7400 zeichnet sich nur durch eine Optimierung für Gleitkommaoperationen aus, so daß dieser den gleichen SPEC-Wert wie der PowerPC 750 erreicht.*

typen ein, die nicht für den integrierten Einbau in Elektronik vorgesehen sind, sondern nur bei stationären Rechnern zum Einsatz kommen.

Als genereller Trend für die beiden zur Diskussion stehenden Typen läßt sich für Ganzzahloperationen eine Überlegenheit von PowerPC-Systemen im Vergleich zu Intel-Prozessoren feststellen, die gleiche Rechenleistung erst bei höheren Taktraten erreichen. Die zum damaligen Zeitpunkt verfügbaren schnellsten Prozessoren waren PowerPCs mit 450 MHz und Pentium-Prozessoren mit etwa 550 MHz, die nach dem Diagramm eine vergleichbare Rechenleistung erreichen sollten. Betrachtet man das Verhalten bei Gleitkommaoperationen, zeigt sich kein signifikanter Unterschied zwischen beiden Prozessortypen, sodaß dort Pentiumsysteme durch die höheren verfügbaren Taktraten Vorteile haben.

Zusätzlich wurden eigene Laufzeittests mit einem Algorithmus durchgeführt, der als Prototyp für die später einzusetzenden Selektionsalgorithmen anzusehen ist. Wie in 4.3 vertieft wird, besteht er aus drei verschachtelten Schleifen, in denen Gleitkommaoperationen ausgeführt werden. Der Algorithmus wurde jeweils 10000 Mal auf verfügbaren Rechnern ausgeführt. Durch das frei erhältliche Programm *time* wurde die benötigte Zeit gemessen. Während der Tests liefen auf den Rechnern keine weiteren Programme, die den Prozessor belasteten. Wiederholte Durchläufe auf einem Rechner zeig-

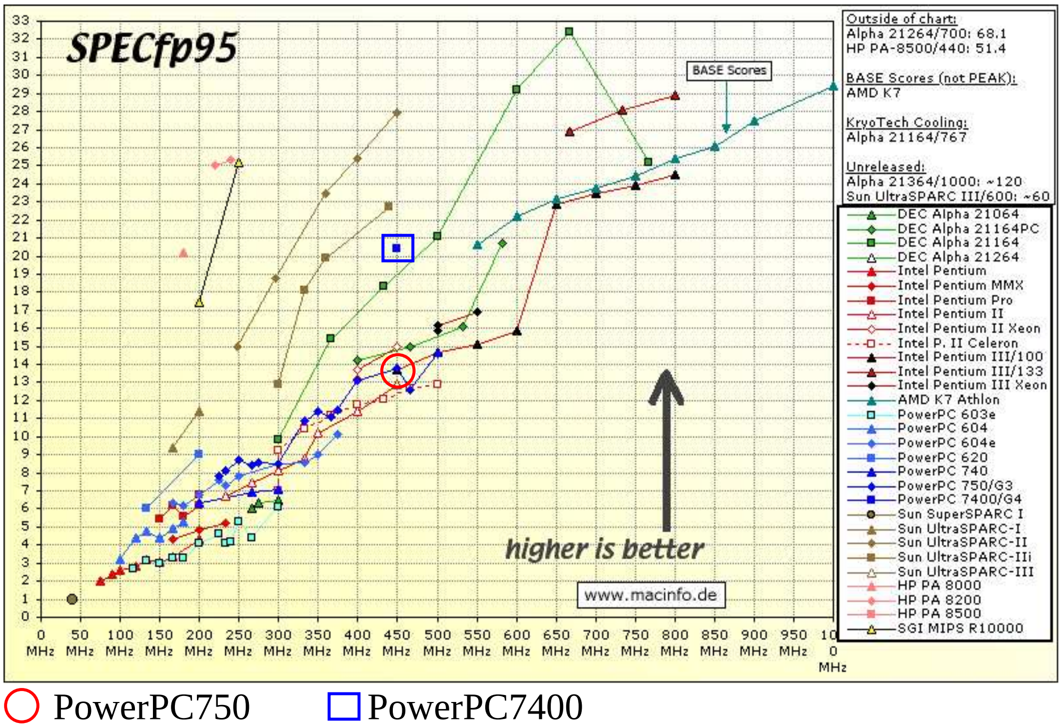



*Abbildung 4.4: SPEC95-Test für verschiedene Prozessoren mit Testprogrammen, die verstärkt Gleitkommaoperationen durchführen [Mac-2002]. Eingekreist ist der Werte für den Prozessor PowerPC 750. Der für Gleitkommaoperationen optimierte Nachfolgetyp PowerPC 7400 ist durch ein Rechteck markiert.*

ten so kleine Abweichungen im Zeitverhalten, daß jede Messung nur fünf Mal durchgeführt werden mußte, um in den meisten Fällen einen ausreichend kleinen statistischen Fehler aus der Mittelwertbildung zu erhalten. Das Programm wurde für Pentium-Rechner mit dem Compiler *gcc* übersetzt und anschließend unter dem Betriebssystem *LINUX* auf Arbeitsplatzrechnern gestartet. Um das Programm auf IBM-Workstations mit PowerPC-Prozessoren unter dem Betriebssystem *AIX* testen zu können, wurde es mit dem IBM-Compiler *xlc* übersetzt. Ein Rechner mit dem später ausgewählten PowerPC750-Prozessor stand für diese Tests nicht zu Verfügung. Für die Testreihe mit einem PowerPC7400-Prozessor wurde ein Macintosh-G4-Rechner mit dem Betriebssystem *LINUX* verwendet. Die Tests wurden jeweils ohne (-O0) und mit (-O2) Optimierung durch den Compiler durchgeführt. Die Ergebnisse des Tests sind in Tabelle 4.1 zusammengefaßt und in Abbildung 4.5 graphisch aufbereitet.

Hier bestätigt sich die Tendenz, daß PowerPC-Prozessoren bei gleicher Taktrate eine höhere Rechenleistung als Pentium-Prozessoren zur Verfügung zu stellen in der Lage sind. Dies liegt im internen Aufbau der Prozessoren als Superskalar-Rechner [Dow-1993] begründet, die in der Lage sind, mehrere Operationen parallel auszuführen. Während ein Pentium-Prozessor nur maximal zwei Operationen gleichzeitig durchführen kann, sind dies beispielsweise bei einem PowerPC 750 drei, wozu sechs verschiedene Recheneinheiten zur Verfügung stehen [Mot-1997]:

| CPU-Typ | Prozessortaktfrequenz [MHz] | Programmlaufzeit [s] | |
|---|---|---|---|
| | | -O0 | -O2 |
| PowerPC 604 | 200 | 27,01 ± 0,16 | 15,53 ± 0,04 |
| Pentium II | 233 | 33,95 ± 0,01 | 19,45 ± 0,01 |
| PowerPC 604 | 233 | 22,80 ± 0,06 | 13,11 ± 0,01 |
| Pentium II | 350 | 22,25 ± 0,04 | 11,90 ± 0,01 |
| PowerPC 604 | 375 | 13,82 ± 0,04 | 7,90 ± 0,01 |
| PowerPC 7400 | 400 | 19,01 ± 0,01 | 12,24 ± 0,01 |
| Pentium II | 400 | 17,60 ± 0,19 | – |
| Pentium III | 450 | 17,05 ± 0,16 | 10,53 ± 0,04 |
| Pentium III | 500 | 14,74 ± 0,04 | 8,44 ± 0,01 |
| Pentium III | 550 | 12,27 ± 0,01 | 7,49 ± 0,01 |

*Tabelle 4.1: Programmlaufzeiten auf verschiedenen Rechnern für 10000 Durchläufe einer Testroutine, wie so bei FTT-L3 zum Einsatz kommen wird. Die Tests wurden mit (-O2) und ohne (-O0) Optimierung durch den Compiler durchgeführt.*

- Integer Units (IUs):
  Zwei Ganzzahl-Einheiten, wobei eine Einheit keine Multiplikationen und Divisionen durchführen kann.

- Floating-Point Unit (FPU):
  Eine Gleitkommazahl-Einheit, die eine Operation in drei Taktzyklen durchführt, wobei jeder Schritt durch eine andere Operation belegt sein kann.

- Load/Store Unit (LSU):
  Eine Datenverwaltungseinheit, die den Daten- und Instruktionentransfer zwischen Speicher und Recheneinheiten kontrolliert.

- System Register Unit (SRU):
  Eine Systemregister-Einheit, um systemnahe Befehle auszuführen und den Datentransfer zu speziellen Registern des Prozessors zu steuern.

- Branch Processing Unit (BPU):
  Ein Register zur Vorhersage von Programmverzweigungen. Die LSU versucht, Daten und Kommandos dem Prozessor unterbrechungsfrei zu Verfügung zu stellen. Diese werden dazu zunächst in Pipelines gespeichert. Treten konditionale Ausdrücke in einem Algorithmus auf, versucht die BPU nach bestimmten Kriterien vorherzusagen, in welchen Pfad das Programm verzweigt. Die zugehörigen Daten und Instruktionen werden dann in die Pipelines geladen. War die Vorhersage falsch, muß die entsprechende Pipeline gelöscht und mit den richtigen Daten und Instruktionen beschrieben werden.

Nicht zu erklären ist allerdings das unerwartet schlechte Abschneiden des PowerPC 7400. Das es sich im eine für Gleitkommaoperationen optimierte Variante des PowerPC 750 mit schnellerere Taktrate handelt, sollte er ein deutlich verbessertes Laufzeitverhalten

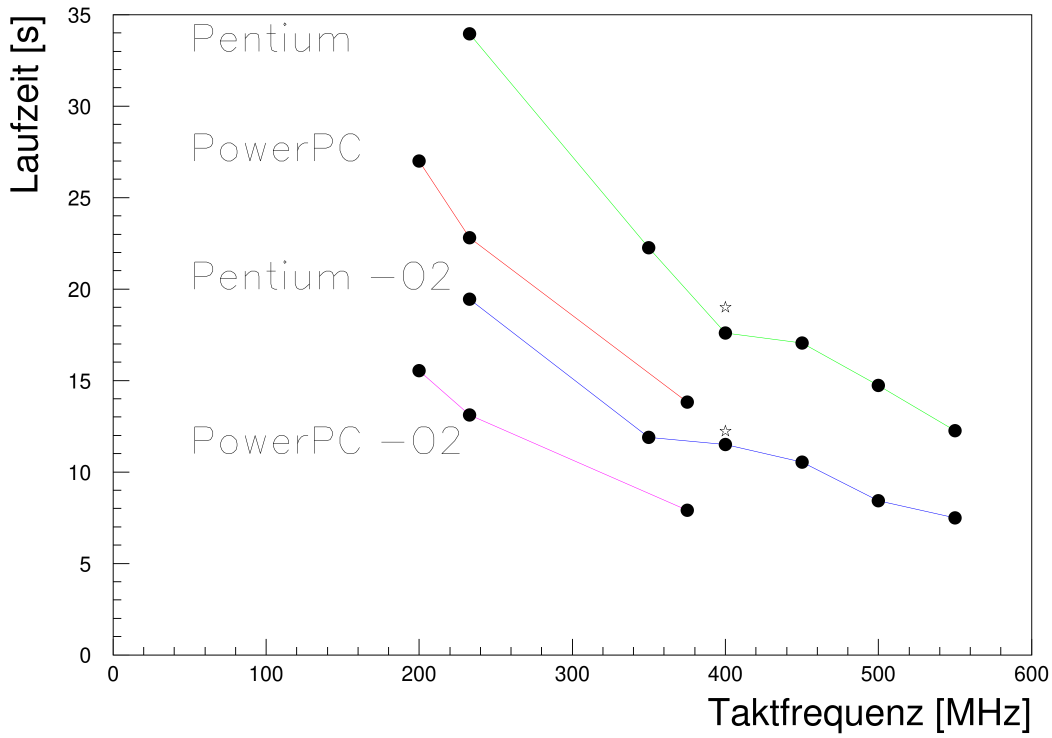



*Abbildung 4.5: Laufzeittest für 10000 Durchläufe eines FTT-L3-Selektionsalgorithmus auf Prozessoren unterschiedlicher Typen und Taktfrequenzen. Die Tests wurden ohne und mit (-O2) Optimierungen durch den Compiler durchgeführt. Vergleicht man die Leistungen bei gleicher Taktrate, sind Power PCs prinzipiell den Pentium-Prozessoren überlegen. Die Lage der beiden einzelnen, durch Sternchen dargestellten Punkte, die jeweils die Ergebnisse für den PowerPC 7400 repräsentieren, ist deutlich höher als erwartet und kann nicht erklärt werden.*

zeigen. Dies konnte auch durch andere Tests bestätigt werden [Hof-2002]. Da der Prozessor zum Zeitpunkt dieses Tests gerade erst auf dem Markt war, könnte ein noch nicht optimaler Compiler die Ergebnisse verfälscht haben.

Wie erwähnt, waren zum Zeitpunkt der Entscheidungsfindung die leistungsstärksten verfügbaren Karten solche mit einem PowerPC 750 Prozessor mit 450 MHz auf der einen, solche mit einem Intel Pentium III mit 550 MHz auf der anderen Seite. Der optimierte PowerPC 7400 wurde zu diesem Zeitpunkt von Motorola gerade auf den Markt gebracht, konnte jedoch erst über ein Jahr später auf VME-Karten erworben werden. Von Intel waren zu diesem Zeitpunkt Pentium-Prozessoren mit einer Taktrate von 800 MHz angekündigt.

Neben der reinen Rechenleistung wurden wie oben beschrieben auch andere Faktoren berücksichtigt. Eine verkürzte Zusammenfassung der Stärken verschiedener Prozessortypen für unterschiedliche Kriterien ist in Tabelle 58 angegeben. Die generelle Lei-

| Prozessortyp | SPARC | PowerPC | PENTIUM |
|---|---|---|---|
| Generelle Leistung | **** | **** | *** |
| SMP[a] | **** | ** | *** |
| Echtzeitverhalten | ** | *** | * |
| Unterstützung von UNIX-Software | **** | * | ** |
| Unterstützung von Echtzeit-Software | ** | *** | ** |
| Unterstützung von Windows-Software | - | - | **** |
| Preis | ** | *** | *** |
| Verhalten bei rauen Einsatzbedingungen | ** | *** | ** |
| Geringe Leistungsaufnahme | *** | **** | ** |
| Zukünftiges Potential | **** | **** | **** |

*Tabelle 4.2: Vereinfachte Übersicht über die Leistungsfähigkeit verschiedener Prozessortypen für die Integration in elektronischen Systemen [Web-1998]. Bei den für den Einsatz in der FTT-L3-Farm wichtigen Punkten schneidet die PowerPC-Architektur am besten ab.*

[a] **S**ymmetric **M**ulti**p**rocessing bezeichnet den gleichberechtigten und gleichzeitigen Einsatz mehrerer Prozessoren zur Lösung eines Rechenproblems unter der Verwaltung einer Instanz eines Betriebssystems.

stungsfähigkeit wurde zuvor schon erörtert. Die als SMP bezeichnete Unterstützung mehrerer Prozessoren unter der gemeinsamen Verwaltung eines Betriebssystems ist eine Funktionalität, die in dem geplanten System nicht benötigt wird. Da der PowerPC als Nachfolger der 68k-Prozessoren, die häufig in Echtzeitanwendungen zum Einsatz kommen, eingeführt wurde, war es Ziel, auch deren günstiges Echtzeitverhalten zu übernehmen und auszubauen. Für die überwiegend in Arbeitsplatzrechnern eingesetzten Pentium-Prozessoren spielt dieses bei der Entwicklung nur eine untergeordnete Rolle. Aus diesem Grund sind zahlreiche Echtzeitbetriebssysteme auf 68k- und PowerPC-Architekturen entwickelt und für diese optimiert worden. Die bisher nicht betrachteten SPARC-Systeme von SUN, die ebenfalls in der Auflistung enthalten sind, finden häufig als Server für Internet-, Datenbank- oder Graphikaufgaben einen Einsatz, da sie im Zusammenspiel mit dem UNIX-Betriebssystem Solaris als Hochverfügbarkeitssystem eingesetzt werden können. Für Echtzeitanwendungen finden sie seltener Verwendung und wurden daher auch nicht als Alternative betrachtet. Preislich liegen SPARC-Systeme auch über Pentium- oder PowerPC-Karten, die beide für unter 5000 Euro erhältlich sind. Problematisch bei Pentium-CPUs ist der relativ hohe Stromverbrauch, der mit einer starken Wärmeentwicklung einher geht. Eine ausreichende Luftzirkulation zur Wärmeabfuhr ist für den sicheren Betrieb einer Pentium-Karte unabdingbar, wohingegen PowerPC-Karten, obwohl mit einem wesentlich kleineren Kühlkörper ausgestattet, nicht so hohe Ansprüche an eine externe Kühlung stellen. Es kann mit Pentium-Karten sogar kritisch sein, mehr als jeden zweiten Steckplatz eines VME-Busses zu bestücken [Biz-2000]. Dies würde die Integration eines geplanten Systems mit 16 Karten in einen VME-Bus unmöglich machen. Dies ist der Hauptgrund, warum PowerPC-Karten als robuster gegenüber rauen Einsatzbedingungen gelten. Das zukünftige Potential wurde zum damaligen Zeitpunkt für alle aufgeführten Prozessortypen generell als positiv bewertet. Aus heutiger Sicht kann angemerkt werden, daß PowerPC 7400 - Prozessoren erst langsam ihre Vorläufermodelle und 68k-Prozessoren verdrängen. Dies ist einerseits auf die späte Verfügbarkeit dieses Typs zum Einbau in Prozessorkarten zurückzuführen, andererseits aber auch darauf, daß die bisher verfügbaren Karten für den Großteil der Anwendungen eine ausreichende Leistung bieten. Nicht unbedingt abzusehen war vor einigen Jahren der rasche Anstieg der Taktleistungen bei Pentium-Prozessoren. Nachwievor nicht zufriedenstellend gelöst ist dabei aber die Temperaturproblematik, die bei steigenden Taktraten noch verstärkt wird [Dow-1993].

Die Wahl fiel schließlich auf eine VME-Karte mit PowerPC 750-Prozessor. Mit der MVME2400-Karte von Motorola wurde eine Karte gewählt, die sich aus folgenden Gründen gegenüber PowerPC-Karten anderer Hersteller (RIO3 von CES, PowerCore 6750 von Force Computers, VG4-Karte mit PowerPC 750 von SBS or-Computers) durchsetzen konnte:

- Die Motorola MVME-Serie hat den höchsten Marktanteil in diesem Segment. Aus den damit verbundenen großen Stückzahlen in der Fertigung resultiert ein günstiger Preis. Außerdem trägt die vielfache Nutzung zur Beseitigung von anfangs vorhandenen Fehlern bei. [Eck-2000]

- Motorola als Hersteller der PowerPC-Prozessoren verfügt als erster Kartenhersteller über die jeweils neuste Generation von Prozessoren. Durch eine enge Zusammenarbeit zwischen Motorola, IBM und Apple und das damit verbundene hohe Potential bei gemeinschaftlichen Entwicklungen bieten die Motorolakarten eine optimierte Architektur, insbesondere bei den integrierten Bussystemen.

- PowerPC-Karten von Motorola gelten als sehr gut dokumentiert.

- Die DPIO-Karten wurden mit MVME-Karten entwickelt und intensiv auf ihnen getestet. [Biz-2000]

- Zwischen Motorola und Windriver, dem Hersteller des in 4.3.1 beschriebenen Betriebssystems, besteht eine enge Zusammenarbeit.

- Über den deutschen Vertriebspartner konnten günstige Preise ausgehandelt werden.

#### 4.2.2.1 Eigenschaften der MVME2400-Karte

Die generellen Eigenschaften der MVME2400-Karte werden in [Mot-1994] beschrieben und sind in Tabelle 4.3 zusammengefaßt. Abbildung 4.6 zeigt eine Übersicht mit den möglichen Datentransferwegen zu der Karte hin oder von der Karte weg.

Der interne Aufbau mit der neu eingeführten PowerPlusII-Architektur soll anhand der schematischen Übersicht in Abbildung 4.7 erläutert werden. Um auf Daten, die häufig benötigt werden, schnell zugreifen zu können, ist neben dem L1-Cache[17], der im Prozessor integriert ist, ein 1 MBytes großer L2-Cache direkt an den Prozessor angeschlossen. Der Hauptarbeitsspeicher RAM und der permanente “FlashROM”-Speicher sind durch

[17] Cache bedeutet wörtlich “Depot, geheimes Lager” und bezeichnet Speicherpuffer zwischen dem Prozessor und dem Arbeitsspeicher, in denen häufig benötigte Daten oder Befehle mit dem Zweck kurzer Zugriffszeiten gespeichert werden. Häufig sind sie in mehreren Stufem (L1, L2) implementiert.

| | |
|---|---|
| Prozessor | PowerPC 750 32-bit RISC[a] mit 450 MHz Taktfrequenz |
| L1 cache | 32 kBytes Daten, 32 kBytes Instruktionen |
| L2 cache | 1 MByte an 180 MHz Bus |
| RAM | 32 MBytes an 100 MHz Bus |
| VME-Anbindung | A16 - A32, D08 - D64, Blocktransfermodus, 7 Interrupts |
| Netzwerkanbindung | 10/100 Mbits/s Ethernet |
| PMC-Anschluß | 2 Steckplätze |
| weitere Anschlüsse | serieller Port |
| getaktete Zählwerke | 1 16-bit-, 4 32-bit-Echtzeituhrzähler,<br>Laufzeitzähler mit Interruptfunktion |
| Stromverbrauch | typisch 3,93 A, maximal 4,31 A bei 5 V |

*Tabelle 4.3: Eigenschaften der MVME2400-Karte. [Mot-1994]*

[a] **R**educed **I**nstruction **S**et **C**omputer

den Prozessorbus (hier PowerPC Bus genannt) und einem Baustein zur Speicherverwaltung (“Hawk”) mit dem Prozessor verbunden. Die Inhalte des “FlashROM”-Speichers bleiben auch bei Stromabschaltung konserviert. 1 MByte dieses Speichers ist in Einstecksockeln untergebracht und enthält die karteneigene Software PPCBug [Mot-2001], um die Komponenten der Karte ohne Betriebssystem konfigurieren und testen zu können. Weitere 8 MBytes dieses Speichers sind fest aufgelötet und können durch den Benutzer beschrieben werden. Hier kann das gesamte Betriebssystem oder der für die Einschaltphase wichtige Teil davon gespeichert werden. Die Instruktionen werden dann automatisch in den Prozessor geladen und ausgeführt. Der Hawk-Baustein dienst außerdem als Brücke zum lokalen PCI-Bus, um diesen vom schnelleren Prozessorbus entkoppelt betreiben zu können. Durch den PCI-Bus werden Komponenten zum Datentransfer auf die Karte integriert. Die beiden PMC-Steckplätze und ein Erweiterungsstecker zur Integration weiterer PCI-Geräte sind direkt an den PCI-Bus angeschlossen. Ein Netzwerkbaustein schafft die Verbindung zu einer mit 10 Bits/s oder 100 Bits/s betreibbaren Ethernetverbindung. Über einen weiteren Brückenbaustein kann ein ISA-Bus mit angeschlossener Echtzeituhr, seriellem Ausgang, Laufzeitzähler und permanentem RAM-Speicher zum Speichern von konfigurierbaren Parametern für die Startphase angesprochen werden. Durch den Universe-Brückenbaustein ist der asynchrone VME-Bus an den synchronen PCI-Bus angebunden.

Teil der PowerPlusII-Architektur ist der extensive Einsatz von Brückenbausteinen zum Trennen von Bussen unterschiedlicher Geschwindigkeit und Übertragungsprotokolle. Zum einen kann so ein Bus schon wieder für den nächsten Datentransfer freigegeben werden, obwohl noch nicht alle Daten ihr Ziel erreicht haben, sondern im Brückenbaustein noch für die Weitergabe über einen langsameren Bus zwischengepuffert sind. Zum anderen können zum Beispiel über den PCI-Bus Daten im DMA-Modus ohne Beteiligung des Prozessors transportiert werden. Dabei kommunizieren zwei Brückenbausteine direkt miteinander. In diesem Modus können zum Beispiel Datenblöcke sehr schnell von einer PMC-Karte oder der VME-Brücke in den RAM-Speicher geschrieben werden, während der Prozessor andere Rechenoperationen ausführt. Dem Prozessor wird über einen Interrupt das Ende des Transfers mitgeteilt.

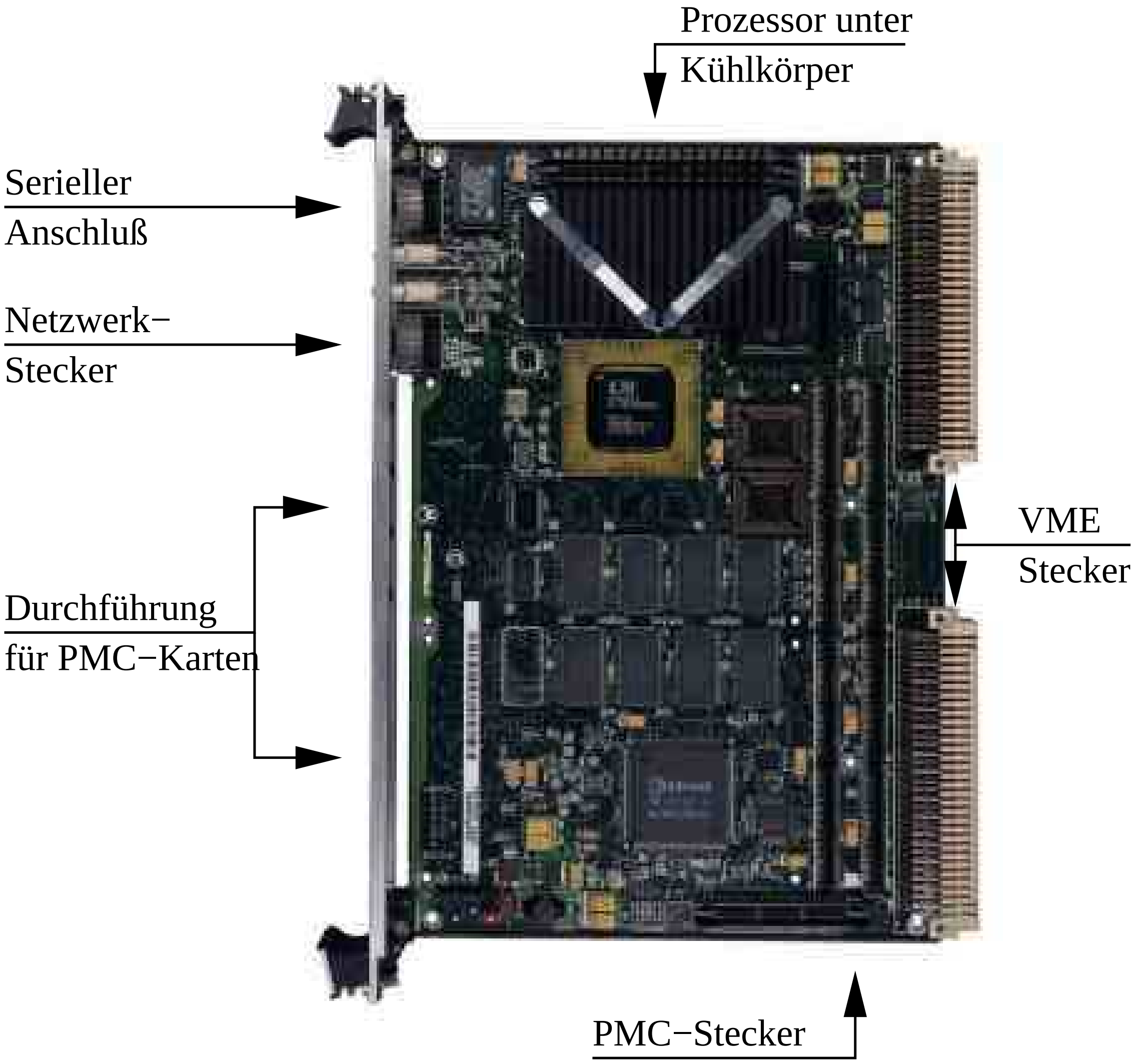



*Abbildung 4.6: MVME2400-Karte von Motorola [Mot-1994]. Im oberen Teil der Karte befindet sich unter einem Kühlkörper der PowerPC 750-Prozessor, in der Mitte befinden sich die Bänke für Speicherbausteine. Für den Datentransfer zu der Karte hin oder von der weg Karte sind ein serieller Anschluß, ein Netzwerkanschluß, eine VME-Schnittstelle und zwei PMC-Steckplätze integriert.*

### 4.2.3 Empfängerkarte

Die L3-Empfängerkarte fungiert als Schnittstelle zwischen der FTT-weiten Punkt-zu-Punkt Datenübertragung über Channel-Link und der parallelen Datenübertragung durch das FPDP-Protokoll innerhalb von FTT-L3. Damit einher geht eine Reduzierung der Datenbreite von 48 Bits auf die durch die FPDP-Verbindung vorgegebenen 32 Bits. Zusätzlich werden die beiden STC-Signale **FastClear** und **L3Reject** empfangen und über FPDP an die Rechenkarten weitergeleitet. Für die zur Zeit laufende Inbetriebnahmephase und spätere in-situ-Kontrollen sind außerdem einige Testmöglichkeiten implementiert

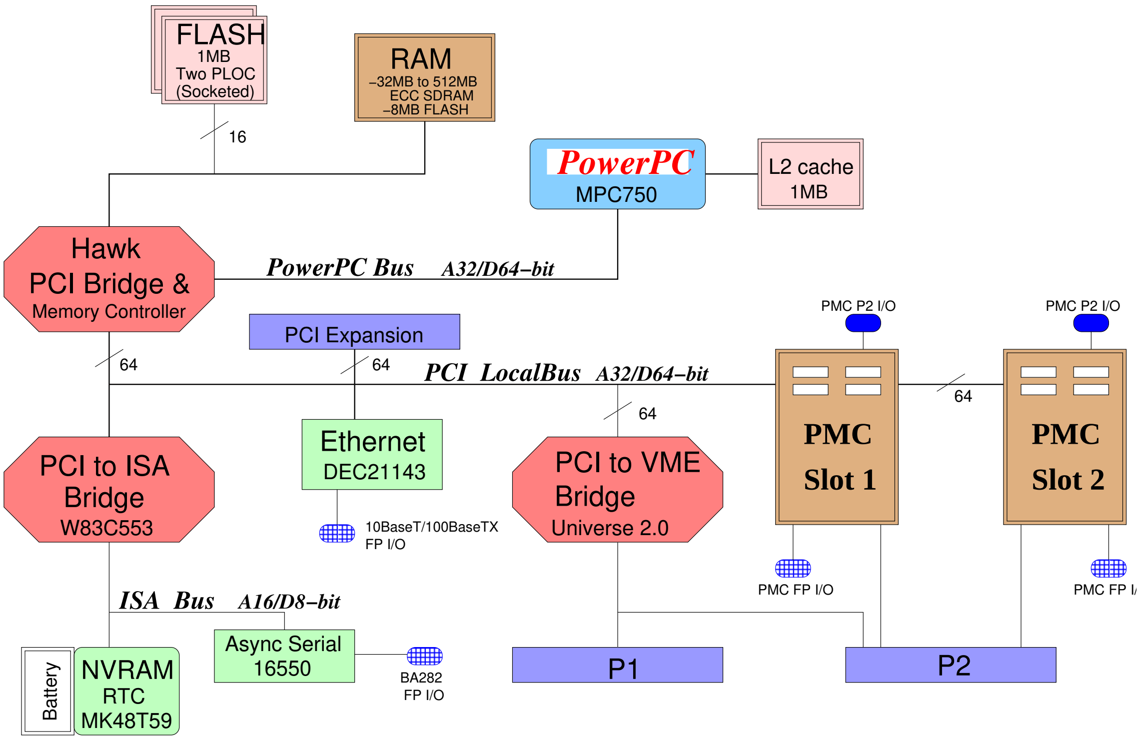



*Abbildung 4.7: Interner Aufbau der MVME2400-Karte [Mot-1994]. Einzelne Komponenten sind durch Datenbusse verbunden. Sogenannte Brücken dienen als Schnittstelle zwischen verschiedenen Bussen. Die gezeigte Busarchitektur wird vom Hersteller als PowerPlusII bezeichnet.*

[Kol-2002]. Eine schematische Übersicht über die Karte zeigt Abbildung 4.8, den Prototypen der Karte Abbildung 4.9.

Um die Empfängerkarte an die Channel-Link-Datenübertragung anzuschließen, ist eine Verbindung für die SCS-Aufsteckkarte auf der Karte integriert. Die Programmierung des FPGAs auf der Aufsteckkarte wird sich an der FTT-L2-Datenkette orientieren. Von FTT-L2 werden die Daten mit einer konfigurierbaren Rate von 50–100 MHz empfangen und in einem FIFO zur Frequenzentkoppelung zwischengespeichert. Die Rate, mit der die Daten aus dem FIFO ausgelesen werden, wird von einem Oszillator auf der Empfängerkarte vorgegeben und beträgt 40 MHz, also das Doppelte der FPDP-Rate. Diese Wahl vereinfacht die Bereitstellung von Taktfrequenzen auf der Empfängerkarte. Die 32 Daten- und 16 Kennbits werden schließlich im LVTTL[18]-Logikstandart auf die Empfängerkarte gesendet. Durch den 80-poligen Stecker wird außerdem die Spannungsversorgung für die Aufsteckkarte hergestellt und zusätzliche Kontrollsignale zur Kommunikation zwischen Empfänger- und Aufsteckkarte ausgetauscht.

Die Aufbereitung der Daten für die Übertragung zu den Rechenkarten wird durch einen zentralen FPGA übernommen. Um ausreichend Ressourcen für Datenspeicher und FIFOs im Baustein zu Verfügung zu haben, wurde ein APEX20K400 von Altera [Alt-2002a] ausgewählt. Die Programmierung des FPGAs [Kol-2002] soll anhand von Ab-

[18] Low Voltage TTL

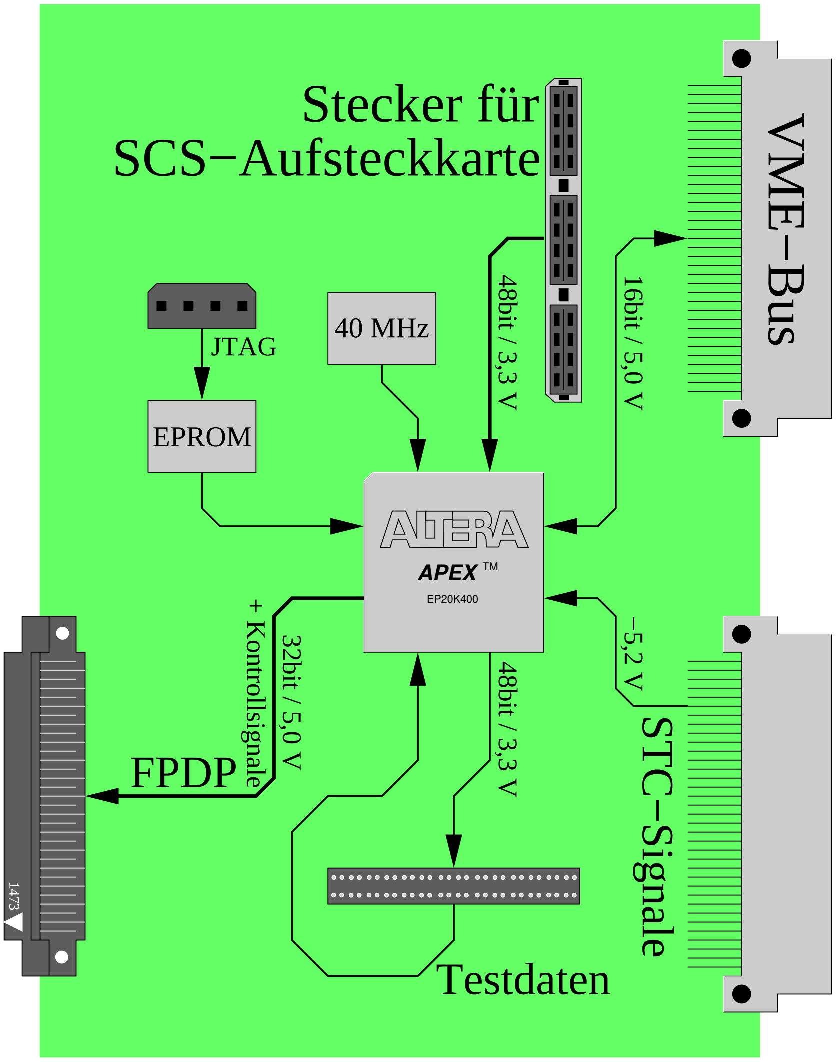



*Abbildung 4.8: Aufbau der FTT-L3 Empfängerkarte. FTT-Daten werden von einer Aufsteckkarte empfangen, durch einen FPGA bearbeitet und für den FPDP-Bus bereitgestellt. Für Konfigurations- und Testzwecke verfügt die Karte über einen Anschluß zum VME-Bus.*

bildung 4.10 erläutert werden. Durch einen Oszillator wird dem FPGA ein 40 MHz-Takt vorgebenen, der durch vier PLLs[19] intern phasenverschoben und auf 20 MHz reduziert werden kann. In einer festen Phasenbeziehung zu dem auf die Aufsteckkarte gegebenen Takt wird mit 40 MHz das Schreiben in den sogenannten LVDS-FIFO erlaubt. Der Schreibzyklus wird allerdings nur dann ausgeführt, wenn ein Kontrollbit von der Aufsteckkarte anzeigt, daß es sich um ein wirkliches Datenwort und keinen Leerlauftakt handelt. Der FIFO dient als zweite Stufe bei der Entkopplung des Channel-Link- und des FPDP-Taktes und gibt die Daten mit 20 MHz wieder aus. Nachdem diese einen Multiplexer (MUX), auf dessen Bedeutung später eingegangen werden wird, durchlaufen

[19] **P**hase **L**ocked **L**oop

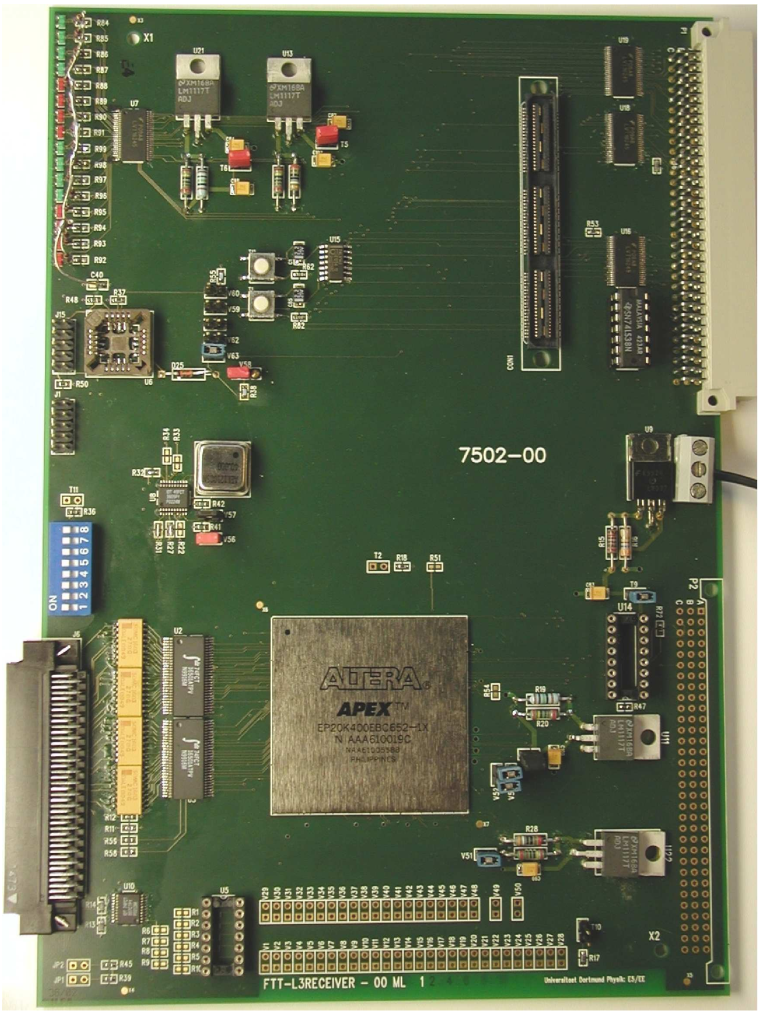



*Abbildung 4.9: Ansicht des Prototypen der L3-Empfängerkarte.*

haben, werden die 16 Kennbits untersucht. Spezielle Bitmuster zeigen dabei jeweils das letzte Wort eines zusammenhängenden Datenblocks bestimmten Typs an. Nach erkennen eines solchen Musters wird mit dem letzten Datenwort die *RES1*-Leitung der FPDP-Verbindung aktiviert, um den Rechenkarten ein **EOT**[20]-Signal zu senden. Von der Firma VMETRO vorgegebene Treiberbausteine speisen alle Daten und Kontrollsignale in das Flachbandkabel der FPDP-Verbindung, über das zusätzlich ein permanenter 20 MHz-Takt übertragen wird. Ein durch die L3-Empfängerkarte aktiviertes **DVALID**-Signal zeigt ein gültiges Datenwort während eines Taktes an. Sollte der FIFO einer DPIO-Karte voll sein, aktiviert die entsprechende Karte das **SUSPEND**-Signal und der FPGA der L3-Empfängerkarte unterbricht die Datenübertragung. Da die FIFO-Tiefe der DPIO-Karten allerdings die Datenmenge eines Ereignisses um ein Vielfaches übersteigt, sollte dieser Fall für FTT-L3 niemals auftreten.

[20] **E**nd **O**f **T**ransfer

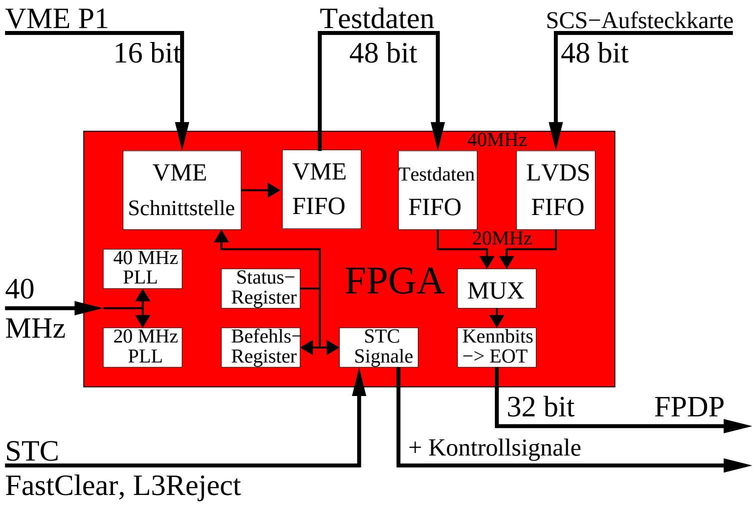



*Abbildung 4.10: Funktionalität im FPGA der L3-Empfängerkarte*

Über unbenutzte Pins des P2-Steckers des VME-Busses (siehe 4.2.1.1) werden die STC-Signale **FastClear** und **L3REJECT** auf die Empfängerkarte gegeben, durch eine Treiberschaltung in ein Logiklevel nach dem LVTTL-Standard transformiert und zum FPGA gesendet. Dort wird daraus ein *PIO1*-, beziehungsweise *PIO2*-Signal nach den FPDP-Spezifikationen generiert und asynchron zum 20 MHz-Takt ausgesendet.

Ergänzend zu dem bisher beschriebenen Datentransfer bietet der FPGA für Testzwecke weitere Möglichkeiten [Kol-2002]. Dazu ist die Empfängerkarte als A24/D16-VME-Karte ausgebildet. Die für eine VME-Schnittstelle benötigte Logik ist ebenfalls in den FPGA implementiert. Diese erlaubt es, in drei Schreibzyklen ein Testdatenwort in einen 48 Bits breiten VME-FIFO zu schreiben, der den FIFO der Aufsteckkarte simulieren soll. Im Unterschied zu diesem werden aber die Daten nicht sofort weitergeleitet, sondern zunächst zwischengespeichert. Ist die gewünschte Anzahl von Testdaten in dem FIFO gespeichert, kann durch ein VME-Schreibzugriff auf ein Befehlsregister der FIFO mit einer Rate von 40 MHz geleert werden. Die Daten werden aus dem FPGA ausgegeben, über eine Sockelleiste geführt und wieder auf den FPGA gegeben. Die Sockelleiste bietet zum einen die Möglichkeit, die Signale abzugreifen und durch einen Logikanalysator zu untersuchen. Zum anderen können über die Sockelleiste Signale eines Bitmustergenerators in die weitere Datenübertragungskette eingeschleift werden. Im folgenden werden die Testdaten analog zu den LVDS-Daten behandelt. Sie werden zunächst in einen weiteren FIFO geschrieben, der hier als Testdaten-FIFO bezeichnet wird. Durch Ändern eines Bits im Befehlsregister wird der Quell-FIFO ausgewählt, aus dem der Multiplexer die Daten ausliest und sie der weiteren Auswertung und Weiterversendung zuführt. Über diesen Weg können über VME-Zyklen oder durch einen Bitmustergenerator eingespeiste Daten zu den Rechenkarten gesendet werden. Durch VME-Schreibzugriffe auf das Befehlsregister können außerdem die *PIO*-Leitungen aktiviert werden. Schließlich ist ein Statusregister in den FPGA integriert, um von außen die aktuelle Konfiguration der Datenübertragung und Füllzustände von FIFOs abfragen zu können.

Um die beschriebene Funktionalität in den FPGA zu laden, ist auf der Karte ein JTAG-

Anschluß mit nachgeschaltetem, beschreibbarem ROM-Baustein vorgesehen. Der JTAG-Bus bietet die Möglichkeit, bitweise Daten von einem Rechner auf einen angeschlossenen Elektronikbaustein zu übertragen. In diesem Fall wird das kompilierte FPGA-Programm in einen EPC8-EPROM-Baustein geschrieben. In diesem Baustein bleiben die Daten auch nach einer Stromunterbrechung weiterhin gespeichert. Nach Einschalten des Stroms werden die Daten des EPROMs automatisch in den FPGA geladen und die ausführbaren Inhalte gestartet.

### 4.2.4 Triggerelementkarte

Die Ergebnisse der Triggeralgorithmen der einzelnen Prozessorkarten müssen der CTL übermittelt werden. Dazu wird für jedes TE zunächst die Entscheidung und dann ein paralleles **STROBE**-Signal gesendet, das anzeigt, daß das übermittelte TE-Signal gültig ist. Zwar steht auf jeder Prozessorkarte ein freier PMC-Steckplatz zu Verfügung, um weiteren Datentranfer zu implementieren, es würde aber für jede Prozessorkarte eine PMC-Karte benötigt werden. Um Kosten zu reduzieren ist daher eine zentrale VME-Karte als Triggerelementkarte vorgesehen, deren Logik wiederum in einem FPGA implementiert ist. In diesem wird eine D16/A24-VME-Schnittstelle integriert werden. Über diese sind 16 unabhängige Register im FPGA ansprechbar, die jeweils einem TE und somit genau einer Prozessorkarte zugeordnet sind. Durch einen VME-Schreibzyklus teilt die Prozessorkarte ihre Triggerentscheidung mit. Daraufhin werden über Treiberbausteine, die auf die Empfängerseite in der CTL abgestimmt sind, zunächst das TE und kurz darauf das **STROBE**-Signal versendet.

Durch Treiberschaltungen analog zu denen auf der Empfängerkarte können außerdem bis zu fünf STC-Signale empfangen und zum FPGA geleitet werden. Dort werden daraus VME-Interrupts mit konfigurierbaren Prioritäten und Interruptvektoren generiert, die durch die Kontrollkarte des Systems ausgewertet werden können.

## 4.3 Programmierung

Da bei FTT-L3 hauptsächlich kommerzielle Elektronikkarten zum Einsatz kommen, lag ein Hauptaugenmerk bei der Entwicklungsarbeit neben der Auswahl der Komponenten auf deren Programmierung. Zunächst soll das verwendete Betriebssystem vorgestellt werden. Anschließend werden die darauf arbeitenden Algorithmen beschrieben.

### 4.3.1 Betriebssystem

Eine Vorraussetzung für einen späteren, effektiven Betrieb ist es, daß jeder potentielle Benutzer von FTT-L3 die Möglichkeit haben muß, einen von ihm geschriebenen Triggeralgorithmus einfach zu integrieren. Dazu muß es möglich sein, die Programmierung in einer Hochsprache wie *FORTRAN*, *C* oder *C++* ausführen zu können. Desweiteren bedingen die strikten Zeitvorgaben eine schnelle Reaktionszeit des Systems auf äußere Ereignisse. Daher kam nur ein Echtzeitbetriebssystem (RTOS[21]) mit Unterstützung einer der genannten Programmiersprachen in Frage. Kriterien bei der Auswahl waren

[21] **R**eal**t**ime **O**perating **S**ystem

- eine gute Dokumentation, um die Einarbeitungszeit klein zu halten.

- eine hohe Verbreitung, um von anderen Anwendern gegebenenfalls Hilfe bekommen zu können.

- schnelles Reaktions- und Programmwechselzeitverhalten. Unter ersterem versteht man die Zeit, die das Betriebssystem benötigt, um auf Ereignisse wie Interrupts reagieren zu können. Letzteres bezeichnet den Zeitbedarf, um die Abarbeitung eines Programmteils zu unterbrechen, dessen aktuellen Zustand abzuspeichern, einen anderen Programmteil in die Register des Prozessors zu laden und dessen Abarbeitung zu beginnen. Beide Zeiten sollten bei einem guten RTOS möglichst kurz sein.

- geringer Bedarf an Arbeitsspeicher, um Prozessorkarten in preisgünstigen Konfigurationen erwerben zu können.

- die Verfügbarkeit von Treibern für die verwendeten Komponenten.

- geringe Kosten.

Die Auswahl der in Frage kommenden Systeme konnte recht schnell eingeschränkt werden. Die zahlreichen neu auf dem Markt befindlichen, preisgünstigen Echtzeitderivate des freien Betriebssystems Linux zeigen noch ein schlechtes Zeitverhalten und vergleichsweise große Lücken bei Treibern für Elektronikkarten. Evaluiert wurden hier zum Beispiel das direkt von Motorola unterstützte "Lineo Realtime Linux" und "OSE", das in hohem Maße objekt-orientierte Programmierung unterstützt.

Eine andere preisgüstige Alternative stellt RTOS-UH[22] dar, das am Institut für Regelungstechnik der Universität Hannover unter anderem auf Karten der MVME-Reihe entwickelt wurde [Ger-1999]. An Lizenzkosten fallen hier nur etwa 100 DM an, die Unterstützung erfolgt direkt durch die Entwickler. Obwohl es auch die Möglichkeit zur Integration von in der Sprache *C* geschriebenen Programmen gibt, wird vorrangig die Hochsprache *PEARL-90*[23] [Giw-1998] unterstützt. Treiber existieren nur für wenige weitere Karten, so daß zum Beispiel ein Treiber für die DPIO-Karte hätte selbst entwickelt werden müssen.

Die Wahl der DPIO-Karte zur Datenübertragung beeinflußte schließlich die Entscheidung zu Gunsten von VxWorks der Firma WindRiver, da nur für dieses RTOS, neben Windows NT, Treiber für diese Karte angeboten werden. Das ebenfalls lange in Betracht gezogene und am H1-Experiment vielfach eingesetzte LynxOS [Lyn-2001] wurde aber auch aus anderen Gründen nicht gewählt:

- VxWorks hat im Vergleich zu LynxOS eine weniger restriktive Speicherverwaltung, die einfachere Zugriffe auf Arbeitsspeicher und Peripherie wie zum Beispiel den VME-Bus ermöglicht [Wat-1994].

[22] **RTOS** **U**ni **H**annover
[23] **P**rocess ans **E**xperiment **A**utomation **R**ealtime **L**anguage

- WindRiver und Motorola haben eine sehr enge Partnerschaft, so daß Motorola-PowerPC-Karten zu den am besten unterstützten VME-Karten auf dem Markt zählen. Die sogenannten BSPs[24], die meistens von den Herstellern der Karten entwickelt werden, kommen hier direkt von WindRiver.

- VxWorks ist das meistbenutze RTOS bei VME-basierten Systemen [VDC-1999], so daß eine große Wissensbasis vorhanden ist. Unter anderem wird VxWorks auch am DESY/Hamburg und am DELTA/Universität Dortmund eingesetzt.

- Für fast jede Elektronikkomponente auf dem Markt ist ein VxWorks-BSP verfügbar, so daß auch eventuelle später Änderungen bei FTT-L3 kein Problem darstellen.

- WindRiver bietet eine intuitive und leicht bedienbare Entwicklungsumgebung für VxWorks an.

- WindRiver bietet umfassende Unterstützung durch gute Dokumentation, Internetseiten[25] und 24-stündigen Telefondienst.

- Preislich ist VxWorks beim Einsatz mehrerer Prozessorkarten etwas günstiger.

Als Basis für die anschließende Beschreibung der Implementierung von Algorithmen für Datentransfer und Datenauswertung sollen das Gesamtsystem VxWorks und typische Programmstrukturen kurz vorgestellt werden. Für eine ausführliche Beschreibung sei auf [Win-1999b] verwiesen.

#### 4.3.1.1 Eigenschaften von VxWorks

Der VxWorks-Betriebssystemkern steuert die Abarbeitung von Programmen inklusive des damit verbundenen Datentransfers auf einer Prozessorkarte. Die Unterstützung verschiedenen Prozessorarchitekturen wird durch BSPs erreicht. Wie in Abbildung 4.11 skizziert, stellt das BSP eine standardisierte Schnittstelle zwischen dem Betriebssystem und der Prozessorkartenelektronik bereit. Darauf aufbauend beinhaltet VxWorks eine große Zahl von Routinen zum Datentransfer, zur Datenmanipulation und Programmsteuerung.

VxWorks ist ein Multitasking-Betriebssystem, das mehrere unabhängige, ausführbare Programmteile (Tasks) zu verwalten in der Lage ist. Aufgabe des Betriebssystems ist es, zu entscheiden, welcher Task durch die CPU bearbeitet wird. Dazu werden den Tasks Prioritäten und Zustände zugeordnet. Wie in Abbildung 4.12 gezeigt, kann ein Task folgende Zustände annehmen:

- **suspended**
  Diesen Zustand nimmt ein Task während der Initialisierungsphase an, solange er für die Ausführung noch nicht zur Verfügung steht. Anschließend wechselt er in einen der anderen Zustände. Darüberhinaus wird dieser Zustand bei der Fehlersuche benutzt, um einen Task nach jedem Programmschritt anhalten zu können.

[24] **B**oard **S**upport **P**ackage
[25] Umfangreiche Unterstützung ist über http://www.windriver.com/windsurf/ zu bekommen

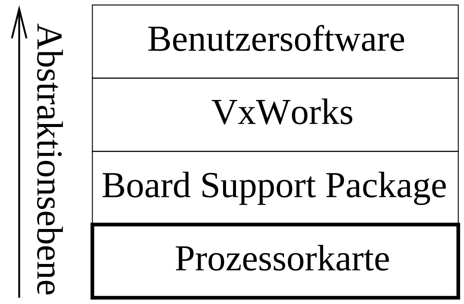



*Abbildung 4.11: Abstraktionsebenen der Software unter VxWorks. Ein Board Support Package stellt durch Bereitstellung von standardisierten Routinen die Schnittstelle zwischen den elektronischen Komponenten einer Prozessorkarte und dem VxWorks-Betriebssystemkern her. Auf diesen setzt wiederum die durch den Benutzer geschriebene Software auf.*

- **ready**
  In diesem Zustand kann ein Task sofort ausgeführt werden, sobald im Rechenzeit der CPU zugeteilt wird. Aus allen Tasks ihm Zustand **ready** wird der mit der höchsten Priorität abgearbeitet. Haben mehrere Tasks diese Priorität, wird ihnen nach einem konfigurierbaren Schema reihum Rechenzeit zugeteilt.

- **pended**
  Ist eine Resource, die ein Task benötigt, nicht verfügbar, geht er in den Zustand **pended** über. Dabei kann es sich zum Beispiel um Semaphoren (siehe Abschnitt 4.3.1.2) oder Daten aus einer bestimmten Quelle handeln.

- **delayed**
  Um die Ausführung eines Tasks zu unterbrechen, kann er für eine bestimmte Zeit in den Zustand **delayed** gebracht werden. Anschließend wechselt er wieder in den Zustand **ready**, wird aber natürlich nur dann ausgeführt, falls kein anderer Task mit höherer Priorität in der Zwischenzeit auch diesen Zustand angenommen hat.

Tasks enthalten häufig eine Endlosschleife, in der eine Programmstelle integriert ist, an der der Task im Zustand **pended** auf die Freigabe einer Resource wartet. Während dieser Zeit können Tasks mit niedrigerer Priorität CPU-Zeit zugeteilt bekommen.

Die Tasks selbst werden entweder direkt mit dem Betriebssystemkern gestartet oder nachträglich auf die Karte geladen. Dies entspringt der Philosopie, daß Entwicklungs- und Laufzeitumgebung zwei getrennte Systeme sind. Beim FTT-L3-Projekt dienen PCs mit dem Betriebssystem Windows NT als Entwicklungsplattform, auf der die graphische Benutzeroberfläche Tornado2 [Win-1999a] installiert ist. Mit dieser wird der Quelltext erstellt und verwaltet. Ein integrierter, auf dem Programm *gcc* [Sta-2002] basierender Compiler übersetzt den Quelltext in ein auf der Zielarchitektur ausführbares Programmformat. Unter dieser Umgebung wird ebenfalls der Betriebssystemkern erstellt, der je nach gewünschter Funktionalität skaliert werden kann. Bei Bedarf können zusätzlich Anwenderprogramme integriert werden. Nach Einschalten der Prozessorkarte wird der Betriebssystemkern über eine Netzwerkverbindung auf diese geladen und gestartet. Weitere ausführbare Programme können später ebenfalls über die Netzwerkverbindung

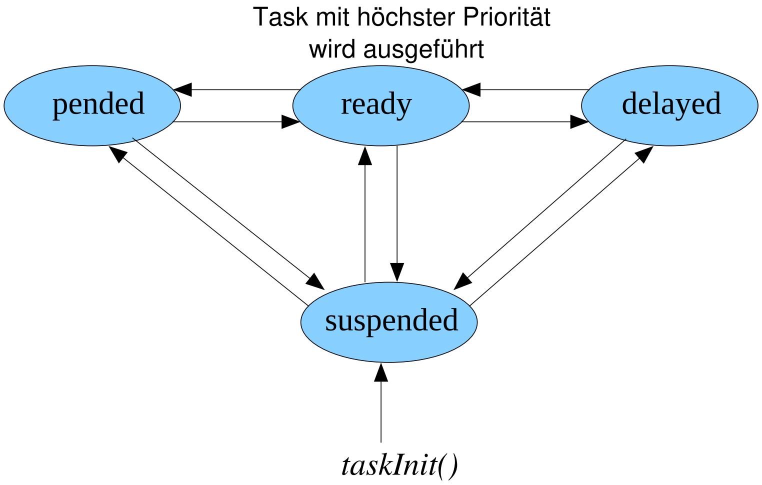



*Abbildung 4.12: Verschiedene Status die ein Task annehmen kann, nach [Win-1999b]. Nach der Initialisierung geht ein Task aus dem Zustand* **suspended** *in den Zustand* **ready** *über. Der Task mit der höchsten Priorität in diesem Zustand wird ausgeführt. Die Ausführung eines Tasks wird unterbrochen, falls er im Zustand* **pended** *auf ein äußeres Ereignis wartet oder im Zustand* **delayed** *für eine bestimmte Zeit von der Ausführbarkeit zurückgestellt wurde. Hauptsächlich zur Fehlersuche kann ein Tasks jederzeit wieder in den Zustand* **suspended** *gebracht werden.*

geladen und manuell gestartet werden. Andere Startkonfigurationen sind möglich und in [Win-1999c] beschrieben.

#### 4.3.1.2 Programmstrukturen

Die im folgenden beschriebenen Programmstrukturen finden sich so oder ähnlich auch in vielen anderen Betriebssystemen, wurden für VxWorks aber unter dem Gesichtspunkt der Anwendung in Echtzeitsystemen optimiert. Zunächst sollen Instrumentarien beschrieben werden, die der Kommunikation und Synchronisation zwischen verschiedenen Tasks auf einer Prozessorkarte dienen.

- Globale Variablen
  Die meisten Betriebssysteme zeichnet eine restriktive Speicherverwaltung aus, in der separate, auf einen abgeschlossenen Speicherbereich zugreifende Programme arbeiten. Diese sind mit Unterroutinen in einer hierarchischen Baumstruktur gegliedert, an der entlang Variablen beim Aufruf von Routinen weitergegeben werden, ohne daß diese Variablen nach außen hin sichtbar sind. Das Konzept zahlreicher Tasks, die zwar zunächst unabhängig arbeiten, aber dennoch miteinander Daten austauschen müssen, führt bei VxWorks zu einer flacheren Struktur, die weniger von Aufrufen von Unterroutinen geprägt ist. Um einen Satz von Variablen für verschiedene Tasks zugreifbar zu machen, werden sie als sogenannte "globale Variablen" außerhalb von Routinen deklariert. Über spezielle Schutzmechanismen muß der Programmierer sicherstellen, daß nicht zwei Tasks oder zwei Inkarnationen einer Unterroutine gleichzeitig auf eine globale Variable zuzugreifen versuchen

und es dadurch zu Inkonsistenzen kommt.

- Semaphoren
  Als Semaphore wird eine Datenstruktur bezeichnet, die im übertragenden Sinne die beiden Zustände **frei** oder **belegt** annehmen kann. Der Zustand einer Semaphore kann durch einen Task geändert werden, indem er sie freigibt oder anfragt. Letzteres kann mit einer Zeitvorgabe geschehen. War es nicht möglich, die Semaphore während der vorgegebenen Zeit zugeteilt zu bekommen, wird der Programmablauf mit einer Fehlerangabe fortgesetzt, die durch konditionale Bedingungen abgefangen werden kann. Eine typische Anwendung mit unendlicher Zeitvorgabe ist es, einen Task auf das Eintreten eines Ereignisses oder die Freigabe eines Resource warten zu lassen. Für unterschiedliche Zwecke unterstützt VxWorks drei Varianten von Semaphoren:

  – Gegenseitige Ausschließungssemaphoren (Mutual Exclusion Semaphores)
    Diese Semaphore ist im Grundzustand **frei** und kann nur von dem Task freigegeben werden, der sie bekommen hat. Sie ist eine typische Lösung, wenn es gilt, eine gemeinsame Resource mehrerer Tasks vor gleichzeitigem Zugriff zu schützen.

  – Binäre Semaphore (Binary Semaphore)
    Die binäre Semaphore ist eine Verallgemeinerung der Ausschließungssemaphore, die jedoch von jedem beliebigen Task freigegeben werden kann. Im Grundzustand ist sie zumeist **belegt**. Eine typische Anwendung ist die Synchronisation von Tasks. Nachdem ein Task eine gegebene Aufgabe abgearbeitet hat, gibt er die Semaphore frei, auf die ein anderer Task wartet und der dadurch aktiviert wird. Beispielhaft sei hier das später beschriebene Aktivieren einer Interruptbehandlungsroutine durch eine Interrupt-Serviceroutine angeführt.

  – Zählende Semaphore (Counting Semaphore)
    Diese Variante der Binären Semaphore soll hier der Vollständigkeit halber erwähnt werden. Sie kann eine mehrmals angefragt und einem Task zugeteilt werden, bevor sie **belegt** ist und somit mehrere Instanzen einer Resource verwalten.

- Nachrichtenwarteschlangen (Message Queues)
  Eine Message Queue kann als Software-Implementation eines FIFOs mit konfigurierbarer Tiefe und Breite interpretiert werden. Sie dient der gepufferten Kommunikation zwischen zwei Tasks, indem ein Task die Message Queue mit Daten beschreibt und ein anderer diese in der gleichen Reihenfolge ausliest. Ist die Message Queue bei einem Lesezugriff leer, kann der ausführende Tasks für einen bestimmten Zeitraum oder bis zum Eintreffen einer Nachricht im Zustand **pended** gehalten werden.

- Signale
  Signale sind mit den unten beschriebenen Interrupts vergleichbar, werden allerdings nicht durch eine Hardwarekomponente, sondern prozessorintern durch

einen anderen Task ausgelöst. Zunächst muß einem Task erlaubt werden, ein bestimmtes Signal empfangen zu können, um dieses anschließend mit einer Signalbehandlungsroutine zu verbinden. Sendet ein Task dem Empfängertask dieses Signal, wird die Signalbehandlungsroutine ausgeführt. Eine typische Anwendung ist es, einen Task zu unterbrechen und in einen definierten Ausgangszustand zu bringen, indem im Programmablauf des Empfängertasks eine Markierung gesetzt und er durch die Signalbehandlungsroutine an diese zurückgesetzt wird. Dabei nehmen die lokalen Variablen wieder die Werte an, die sie zum Zeitpunkt der Markierung hatten.

Ergänzend zu den bisher erwähnten Kommunikationsmöglichkeiten innerhalb eines Prozessors sollen nachfolgend solche über dessen Grenzen hinweg vorgestellt werden:

- Interrupts
  Interrupts werden bei bestimmten Ereignissen oder durch einen Softwarebefehl in einem Baustein ausgelöst und über einen Bus übertragen. Brückenbausteine können konfiguriert werden, die Interruptstrukturen zweier Busse miteinander zu verbinden, so daß zum Beispiel ein VME-Interrupt in einen PCI-Interrupt übersetzt werden kann. Interrupts unterscheiden sich durch ihre Priorität und eine mitgeschicktes Datenwort. Dieser sogenannte Interruptvektor löst bei dem Empfänger weitere Aktionen aus, indem er mit einer Interrupt-Serviceroutine (ISR) verknüpft wird, die mit höchster Priorität abgearbeitet wird. Um die Echtzeitfähigkeit eines Systems nicht zu gefährden, darf in einer Interrupt-Serviceroutine nur ein eingeschränkter Satz von Befehlen verwendet werden. Daher wird dort typischerweise nur der Interrupt deaktiviert, der Quelle der Empfang des Interrupts bestätigt, die ihn daraufhin zurückgesetzt, der Interrupt gegebenenfalls neu aktiviert und eine Interrupt-Behandlungsroutine durch Freigabe einer Semaphore gestartet.

- VME-Bus-Zugriff
  Durch die offene Speicherverwaltung von VxWorks unterscheidet sich die Programmierung eines Zugriffes auf den VME-Bus nicht von der auf lokalen Speicher. Ist auf einer anderen Karte im VME-Bus Speicher freigegeben, kann in diesen einfach durch Zugriff auf eine konfigurierbare Adresse geschrieben, oder von dieser gelesen werden. Im Falle der MVME2400-Karte muß dazu die Abbildung der Adressräume des Prozessor-, PCI- und VME-Busses aufeinander bekannt sein.

- Reservierter Speicherbereich
  Um Daten über den VME-Bus austauschen zu können, ohne Gefahr zu laufen, wichtige Speicherinhalte auf einer anderen Karte zu überschreiben, kann ein Teil des Arbeitsspeichers einer Karte der Verwaltung durch das Betriebssystem entzogen und vom Nutzer verwaltet werden. Bei FTT-L3 sind hierfür 1 MByte auf jeder Karte reserviert. Die gewählte Konfiguration der VME-Schnittstelle erlaubt es nur, in diesen Bereich Schreib- und Lesezugriffe auszuführen. Der Nutzer hat sicherzustellen, daß eine Speicherstelle nicht mehrfach verwendet wird und die Adresse und Struktur von Daten den kommunizierenden Tasks bekannt ist.

- Dateizugriff
  VxWorks bietet in der Hauptsache zwei Möglichkeiten, auf Dateien eines anderen

Rechners über eine Netzwerkverbindung zuzugreifen:

– File Transfer Protocol FTP [Pos-1985]
  FTP ist ein einfaches Protokoll um ganze Dateien über eine Netzwerkverbindung zu transferieren. VxWorks unterstützt lesen, aktualisieren und schreiben einer Datei. Für die ersten beiden Punkte wird zunächst die Datei von einem anderen Rechner gelesen und im lokalen Arbeitsspeicher eine Kopie angelegt, für letzteren wird zunächst nur Arbeitsspeicher reserviert. Auf diese lokale Kopie werden nun Schreib- und Lesezugriffe durchgeführt. Eine zum Schreiben oder Aktualisieren geöffnete Datei wird nach dem Schließen wieder auf den anderen Rechner zurückkopiert, eine zu lesende einfach aus dem lokalen Speicher gelöscht. Nachteil hierbei ist, daß der Arbeitsspeicher die Dateigröße limitiert und es zu großen Datenverlusten kommen kann, falls die Verbindung zwischen Prozessorkarte und Rechner abbricht.
– Network File System NFS [Cal-1995]
  Durch NFS wird keine lokale Kopie der Datei angelegt, sondern Dateizugriffe werden über die Netzwerkverbindung übertragen und direkt auf dem anderen Rechner ausgeführt. Dadurch tritt keine Limitierung der Dateigröße und auch weniger Datenverlust bei Störungen auf. Dateien können auch schon während der Beschreibens durch die Prozessorkarte auf dem anderen Rechner gelesen werden. NFS beansprucht die Netzwerkanbindung der Karte allerdings stärker.

In den folgenden Abschnitten wird die Programmierung der L3-Prozessorkarten unter Verwendung der zuvor beschriebenen Strukturen vorgestellt.

### 4.3.2 Rechnerkarten

Die Datenverarbeitung auf den L3-Rechnerkarten teilt sich in vier unabhängige Blöcke auf:

- Dateneingang
- Datenaufbereitung
- Datenauswertung
- Datenausgabe

Da der erste und letzte Punkt eher technischer Natur sind und durch spätere Anwender nicht modifiziert zu werden brauchen, wurde die dafür benötigte Software mit anderen Hilfsprogrammen in einer gemeinsamen Datei *fttL3SlaveLib* zusammengefaßt. Die Datenaufbereitung und Auswertung muß den jeweiligen Wünschen der Nutzer unter physikalischen Gesichtspunkten angepaßt werden. Alle dabei zu ändernden Programmteile sind in *fttL3PhysLib* zusammengefaßt und sollen später am Beispiel der Suche nach Teilchenzerfällen beschrieben werden.

Vorab soll kurz auf die Initalisierungsphase eingegangen werden. Bei der Inbetriebnahme einer Prozessorkarte wird dieser eine eindeutige Prozessornummer zugeteilt. Ist diese ungleich 0, wird nach dem Einschalten und Starten des Betriebssystems automatisch *fttL3SlaveInit()* aufgerufen. Prozessorkarten mit der Nummer 0 sind der FTT-Datenauslese zugeordnet. Um eine Rechenkarte zur Datenbearbeitung vorzubereiten, führt die Initialisierungsroutine folgende Schritte aus:

- Das FTT-Nachrichtensystem wird gestartet (siehe Abschnitt 4.3.2.5).
- Die DPIO-Karte wird initialisiert.
- Der nicht durch VxWorks verwaltete Speicherbereich wird für die Aufnahme der Auslesedaten vorbereitet (siehe Abschnitt 4.3.2.4).
- Im durch VxWorks verwalteten Speicherbereich wird Platz für die Eingangsdaten zugewiesen.
- Semaphoren werden eingerichtet.
- DMA-Informationen werden in den Speicher der DPIO-Karte geschrieben.
- Tasks werden gestartet.
- Die Selektionsalgorithmen werden durch *fttL3PhysInit()* initialisiert.
- PCI-Interrupts der DPIO-Karte werden mit ISRs verknüpft und aktiviert und die Datennahme der DPIO-Karte gestartet.
- Ist die Initalisierungsphase der Rechenkarte abgeschlossen, wird ein Kontrollwort an einer Adresse im reservierten Speicher überschrieben. Die Kontrollkarte überprüft diese Adressen reihum auf allen Rechenkarten, um eine Liste aller aktiven Rechenkarten erstellen zu können.

Die generelle Struktur des periodischen Programmablaufs während der Bearbeitung eines Ereignisses auf einer L3-Rechnerkarte ist in Abbildung 4.13 skizziert und wird nachfolgend erläutert. Zur Vereinfachung wird dabei angenommen, daß als L2L3-Daten nur ein Block von PQZP-Daten mit fester Länge und anschließend Spurdaten variabler Länge für die Selektionsalgorithmen übertragen werden.

#### 4.3.2.1 Dateneingang

Die Bearbeitung von Eingangsdaten wird durch das Eintreffen eines Wortes in den FIFO der DPIO-Karte ausgelöst. Diese generiert einen PCI-Interrupt mit der Bedeutung **FIFO_NOT_EMPTY** (FNE), der mit der ISR *fttL3SlaveIsrFifoNotEmpty()* verknüpft ist. Durch Freigabe einer Semaphore wird die Routine *fttL3SlaveDmaStart()* aktiviert, die das erste Datenwort direkt aus dem FIFO ausliest und analysiert. Handelt es sich dabei um ein bekanntes Startwort eines Datenpaketes, wird der entsprechende DMA gestartet und der DMA-Typ in einer globalen Variable gekennzeichnet. Anderenfalls wird der Interrupt wieder aktiviert und auf das Eintreffen des nächsten Datenwortes gewartet.

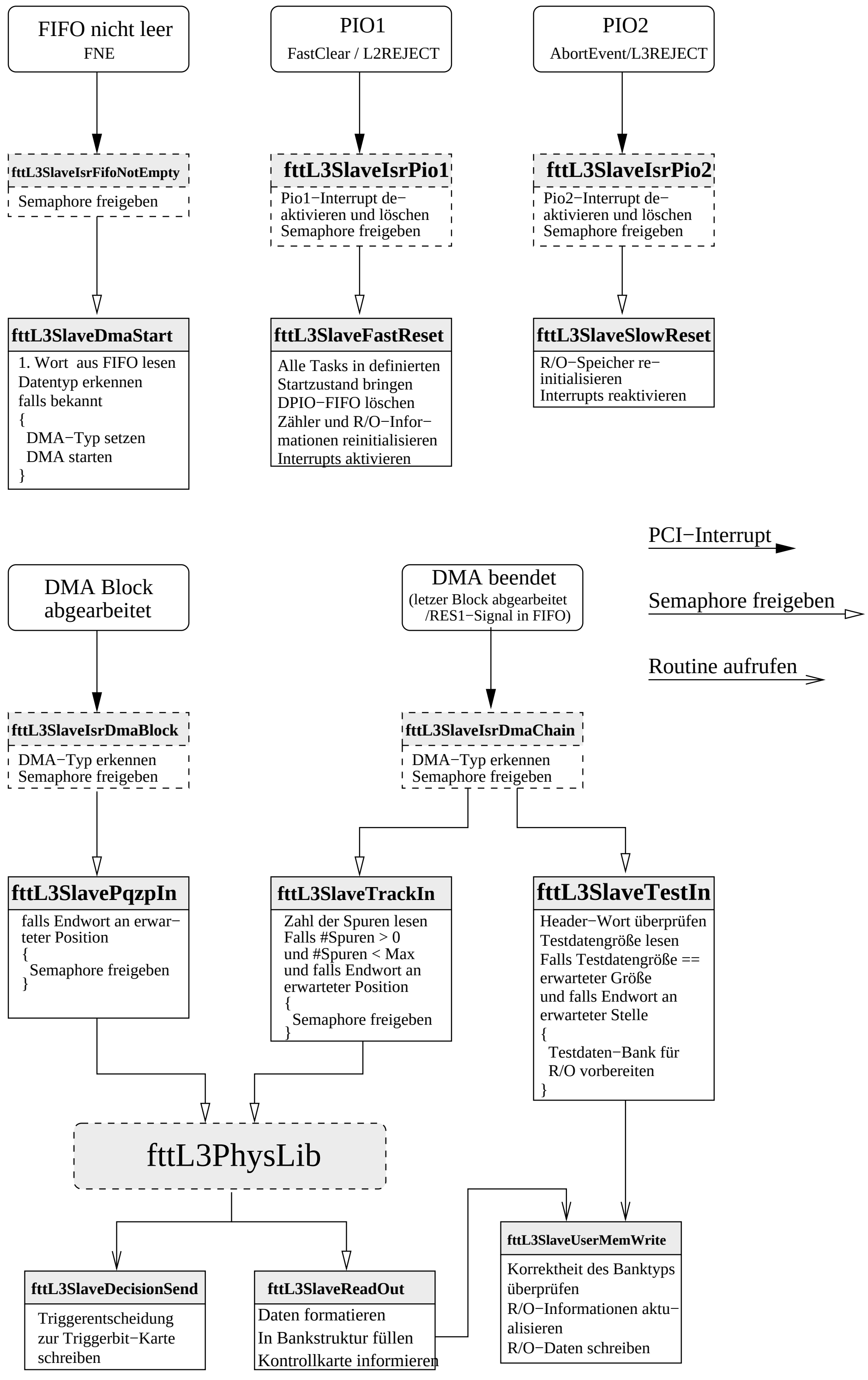



*Abbildung 4.13: Struktur der Hilfsroutinen in der Programmbibliothek* fttL3SlaveLib. *Sie bilden den Rahmen für die Berechnung einer Triggerentscheidung auf einer L3-Prozessorkarte. Der genaue Programmablauf ist im Text beschrieben.*

Der Ablauf eines DMA-Datentransfers aus dem FIFO der DPIO-Karte soll vereinfacht an Hand von Abbildung 4.14 veranschaulicht werden. Im oberen Teil der Abbildung ist die Funktionalität der DPIO-Karte schematisch dargestellt, die durch eine PCI-Brücke mit der Prozessorkarte verbunden ist. Das erste Datenwort löst den beschriebenen FNE-Interrupt aus, wird aus dem FIFO ausgelesen und als PQZP-Startwort identifiziert. Daraufhin wird der unter dem DMA-Index 1 im RAM abgespeicherte Block von drei Speicheraddressen verwendet, um die PQZP-Daten ohne Beteiligung des Prozessors in den RAM-Speicher der Rechenkarte zu schreiben. Ist der Block von Adressen abgearbeitet, werden weitere Informationen ausgewertet, die mit den Adressen abgespeichert wurden. Das **TRUE** (T) im letzten Feld besagt, daß die DPIO-Karte einen **END_OF_BLOCK**-Interupt (EOB) senden soll, um dem Prozessor die Abarbeitung der PQZP-Adressen mitzuteilen. Dieser kann die Daten nun weiter bearbeiten. Da noch die Spurdaten im FIFO erwartet werden, ist als nächster DMA-Index eine 2, also der der Adresse des Spurstartwortes angegeben. Völlig autark startet die DPIO-Karte bei Eintreffen des nächsten Datenwortes einen DMA-Transfer unter der Annahme, daß es sich um das Spurstartwort handelt. Anschließend wird kein Interrupt ausgelöst, sondern automatisch zu dem Addreßblock mit dem DMA-Index 3 gewechselt. Eintreffende Daten werden unter der Annahme, daß es sich um Spurdaten handelt, in den Arbeitsspeicher geschrieben. Unter dieser Annahme aktualisiert der Prozessor bei Beendingung des PQZP-Datentransfers auch die Variable, die den aktuellen DMA-Typ angibt. Sollte zu Beginn keine PQZP-Datenübertragung stattgefunden haben, wird das erste Wort als Spurstartwort erkannt, und gleich der Adreßblock mit dem DMA-Index 3 verwendet. Der DMA-Transfer kann beendet werden, wenn bei Über¨-tragung der maximalen Anzahl von Spuren alle Adressen eines Blocks aufgebraucht sind, oder wenn mit einem Datenwort das *RES1*-Signal mitgeschickt wird. In beiden Fällen löst die DPIO-Karte einen **END_OF_CHAIN**-Interrupt (EOC) aus. Damit ist die gesammte Datenübertragung auf die Prozessorkarte abgeschlossen und die dem FIFO zugeordneten Interrupts bleiben bis zu einem **FastClear** deaktiviert. Werden mehrere Datensätze variabler Länge übertragen oder variiert die Abfolge der Datenblöcke zwischen Ereignissen, muß nach jedem Block ein **RES1**-Signal gesendet werden, um den Typ eines möglichen nächsten Transfers wiederum aus dem ersten Wort bestimmen zu können.

Empfängt der Prozessor einen Interrupt, der das Ende einer DMA-Übertragung anzeigt, wird durch die gestartete ISR der DMA-Typ festgestellt und eine entsprechende Semaphore freigegeben, die einen Task zur Verifikation der Daten ablaufen läßt. In *fttL3SlaveTrackIn()* wird zum Beispiel die Anzahl der zu übertragenden Spurdaten aus dem Startwort extrahiert und auf Konsistenz überprüft. Anschließend wird daraus die hypothetische Stelle des Endwortes berechnet und dessen Vorhandensein kontrolliert.

Für alle Datenblöcke mit physikalischer Relevanz wird nach erfolgreichem Test wiederum eine Semaphore freigegeben, die die Datenaufbereitung und Auswertung in *fttL3PhysLib* startet. Auf die Möglichkeit zur Übertragung von Testdaten wird weiter unten eingegangen.

#### 4.3.2.2 Datenaufbereitung

Die Programmstruktur der für diesen und den nächsten Abschnitt relevanten Routinen ist in Abbildung 4.15 schematisch dargestellt. Durch Unterroutinen werden die Ein-

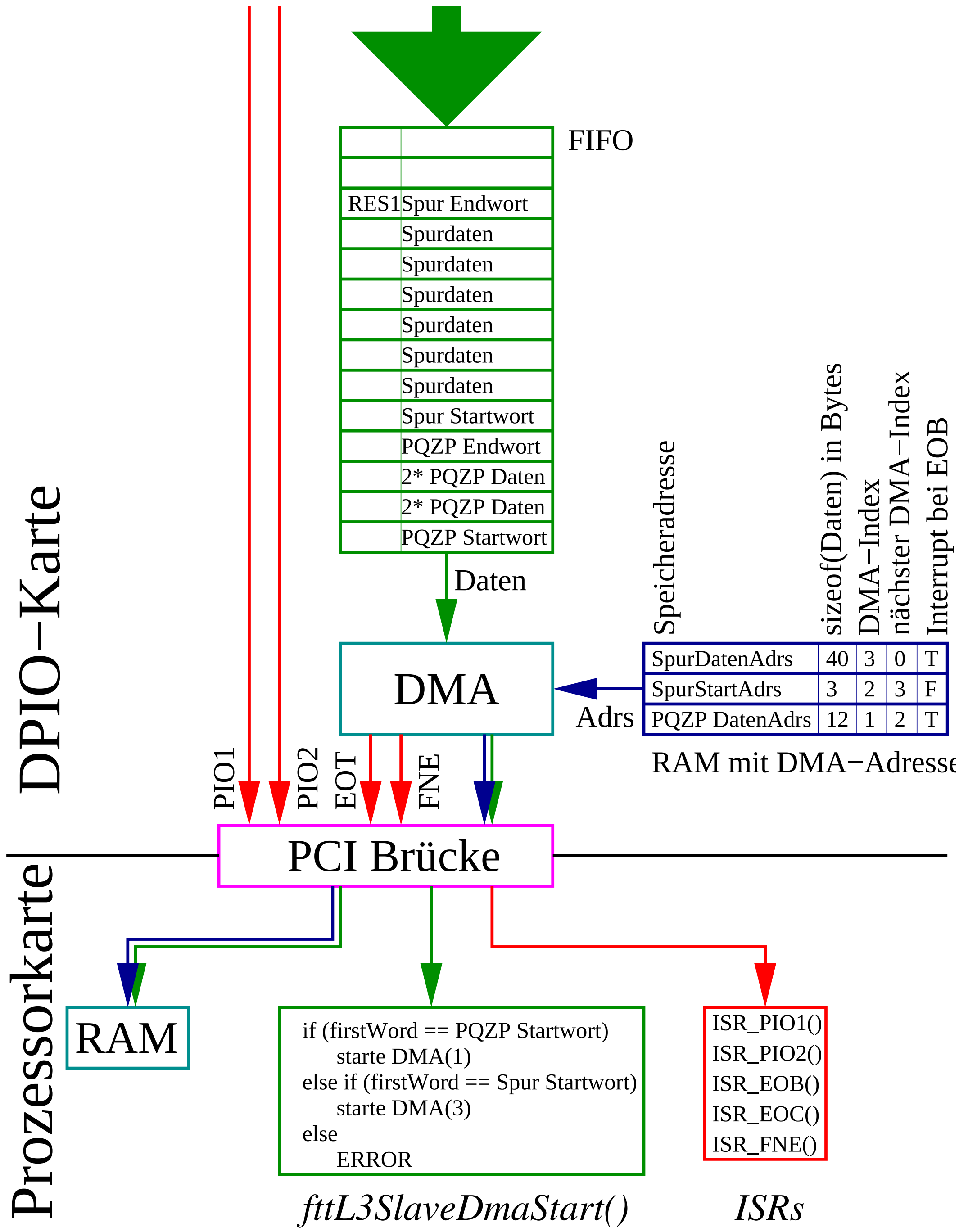



*Abbildung 4.14: Schematische Darstellung der Funktionalität der DPIO-Eingangskarte. Der genaue Ablauf einer Datenübertragung wird in Abschnitt 4.3.2.1 erläutert.*

gangsdaten, die häufig noch nicht physikalischen Größen entsprechen, aufbereitet. In dem Bitmuster der PQZP-Daten sind meistens geometrische Detektorinformationen kodiert, auf die im folgenden aber nicht näher eingegangen werden soll. Wie in Abschnitt 3.2.6 besprochen, werden für jede Teilchenspur die Größen `kappa`=$\kappa$, `phiStar`=$\varphi*$ und `cotTheta`=cot($\theta$) übertragen. Daraus lassen sich eine Reihe von abgeleiteten Größen berechnen und in einer Datenstruktur speichern, wie zum Beispiel:

- Azimuthalwinkelinformation $(\sin(\varphi), \cos(\varphi)) = \mathrm{f}(\varphi^*)$
  In Abschnitt 3.2.6 wurde beschrieben, wie die Azimuthalwinkelinformation in $\varphi^*$ kodiert wird. Der Winkel $\varphi$ selbst wird für die auf FTT-L3 ausgeführten Algorithmen nicht benötigt, statt dessen werden wieder die beiden trigonometrischen Funktionen extrahiert. Unter Berücksichtigung einer eventuell addierten Konstanten $c$ zur Unterscheidung der Oktanten lassen sich aus $\varphi^*$ zwei Größen $a$ und $b$ berechnen, die durch eine vom Oktant abhängigen Fallunterscheidung direkt den beiden gesuchten Größen $\sin(\varphi)$ und $\cos(\varphi)$ zugeordnet werden können.

$$\begin{aligned} a &= \frac{1}{\sqrt{2}}(\varphi^* - c) && (4.5) \\ b &= \pm\cos(\arcsin(a)) \\ &= \pm\sin(\arccos(a)) \\ &= \pm\sqrt{1-a^2} && (4.6) \end{aligned}$$

  Das Vorzeichen ist entsprechend des jeweiligen Oktanten zu wählen. Da $a$ nur Werte im Bereich $a \in [-\pi/4, \pi/4]$ annehmen kann, läßt sich die Wurzel entwickeln durch (siehe zum Beispiel [Bro-1991])

$$\begin{aligned} b &= \pm\sqrt{1-a^2} \\ &\approx \pm\left(1 - \frac{1}{2}\cdot a^2 - \frac{1}{8}\cdot a^2 \cdot a^2 - ...\right). && (4.7) \end{aligned}$$

  Untersuchungen haben gezeigt, daß bei einem Abbruch nach dem dritten Glied eine ausreichende Genauigkeit erzielt wird [Win-2002a]. Bei der Implementierung des Algorithmus wird $a^2$ einmalig berechnet, um die Zahl der Multiplikationen zu reduzieren. Desweiteren sind die Vorfaktoren bei Programmstart zu berechnen, um die Brüche durch einfache Multiplikationen zu ersetzen.

- Polarwinkelinformation $\cot(\theta)$
  Diese Information wird direkt aus den übertragenen Daten in die L3-Datenstruktur gefüllt, da der Winkel $\theta$ selbst später nicht benötigt wird. Soll ein Schnitt auf den Streuwinkel gemacht werden, können dessen Grenzen dann als $cut = \cot(\theta_{\mathrm{cut}})$ angegeben werden. Dies ist ein Beispiel dafür, daß es sich lohnt, Rechenarbeit vorab bei der Berechnung der feststehenden Schnitte für Triggerbedingungen zu leisten, statt Umformungen und Rechnungen mit den Variablen durchzuführen. Dadurch wird Rechenzeit im Prozessor eingespart.

- Transversalimpuls $p_t$
  Zur Berechnung des Transversalimpulses wird die Spurkrümmung $\kappa$ und der Wert

des Magnetfelds benötigt. Dieser wird bisher als zeitlich konstant angenommen, könnte in späteren Versionen aber auch aktuell aus einer Datenbasis ausgelesen und auf die Karte geladen werden.

- Impulskomponenten $p_x, p_y, p_z$
  Die Impulskomponenten werden aus dem Transversalimpuls und den Winkelinformationen berechnet und später zur Addition und Skalarproduktbildung der Impulsvektoren zweier Teilchen benötigt.

- Quadrat des Gesamtimpulses $p_{tot}^2$
  Das Quadrat des Gesamtimpulses wird nachfolgend für die Bestimmung der Energie eines Teilchens benötigt und hier vorab berechnet.

- Ladungsvorzeichen $Q$
  Das Ladungsvorzeichen eines Teilchens ist entgegengesetzt zu dem des $\kappa$ seiner Spur.

- Quadrate der Energiehypothesen $E_\pi^2, E_K^2$
  An dieser Stelle wird das Quadrat der Energie eines Teilchens mit gegebenem Gesamtimpuls unter der Annahme berechnet, daß es sich um ein Pion oder Kaon handelt. Diese Werte werden später beim paarweisen Verknüpfen von Spuren und der Berechnung der resultierenden invarianten Masse häufiger benötigt, können aber schon an dieser Stelle berechnet werden. Dies ist wiederum ein Beispiel für die Vorabberechnung später häufig benutzer Werte, um Rechenzeit zu sparen. Alternativ ist zum Beispiel für die Selektion von $J/\Psi$-Zerfällen hier $E_\mu^2$ zu berechnen.

Die Datenstruktur und die Berechnung der Inhalte muß bei Bedarf durch die Nutzer an die jeweiligen Bedürfnisse angepaßt werden.

#### 4.3.2.3 Datenauswertung

Am Beispiel der $D^*$-Selektion soll die Struktur einer Selektionsroutine erläutert werden. Um die Laufzeit klein zu halten, können dabei einige Programmiertechniken eingesetzt werden, die beispielsweise in [Dow-1993] beschrieben sind:

- Verschiedene Klassen arithmetischer Operationen zeigen deutliche Unterschiede im Zeitbedarf. Nachfolgend sind einige Operationen nach steigendem Zeitbedarf sortiert aufgelistet. Dies kann allerdings nur als grober Anhaltspunkt gelten, da der Compiler oder eine CPU-Karte für eine bestimmte Operation mehr oder weniger gut optimiert sein kann.
  - Addition und Subtraktion
  - Multiplikation
  - Division, Wurzel, Exponentialfunktion, Logarithmus, Trigonometrische Funktionen

  Da zum Beispiel bei der Berechnung der Potenz `z = pow(x, y)` zunächst `x` in einen Logarithmus umgewandelt, multipliziert und anschließend zurückkonvertiert wird, ist es für $y = 2$ zeitgünstiger, `z = x * x` zu schreiben. Ebenso ist `y = x + x + x` der geläufigeren Form `y = 3 * x` vorzuziehen.

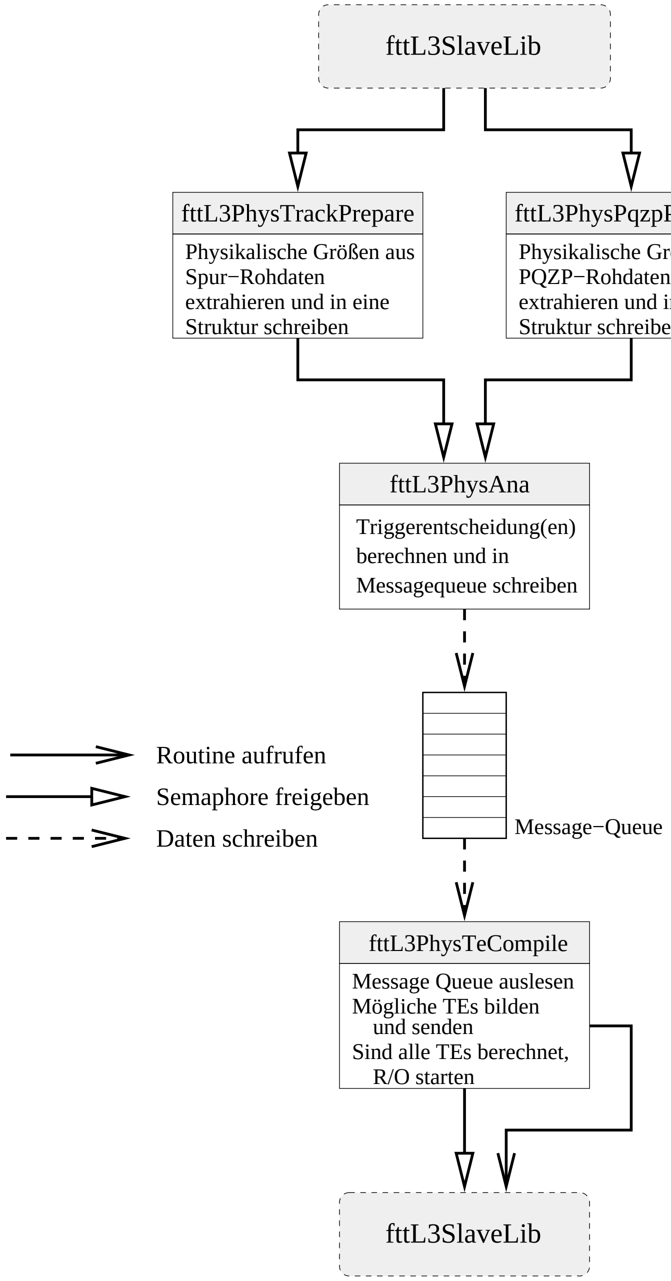



*Abbildung 4.15: Programmstruktur zur Extraktion von Triggerelementen aus der physikalischen Analyse der Eingangsdaten. Sind die Daten auf der Karte gespeichert, wird durch Freigabe entsprechender Semaphore zunächst die Datenaufbereitung und dann die Datenauswertung gestartet. Ein oder in Sonderfällen auch mehrere Ergebnisse für einen Datensatz werden durch eine Messagequeue zu einem Task geleitet, der aus den Einzelentscheidungen Triggerelemente generiert, welche durch eine Hilfsroutine zur Triggerelementkarte geschrieben werden. Sind alle Triggerelemente gebildet worden, werden die Daten für die Auslese aufbereitet, die anschließend durch Routinen aus* fttL3SlaveLib *durchgeführt wird.*

- Mehrmals benötigte Ausdrücke sollten nur einmal berechnet werden. Statt
  ```
  D = C * (A + B)
  E = (A + B) * 3
  ```
  zeigt
  ```
  temp = A + B
  D = C * temp
  E = temp * 3
  ```
  ein wesentlich besseres Laufzeitverhalten.

- Ein Sonderfall des letzten Punktes ist das Entfernen invarianter Ausdrücke aus Schleifen. Tritt in einer Schleife beispielsweise der Ausdruck `a = (b / c) + i` auf, ändert aber nur `i` bei jedem Durchlauf seinen Wert, so kann `b / c` schon außerhalb der Schleife berechnet werden.

- Der Aufruf einer Unterroutine oder ein Taskwechsel benötigt einige Mikrosekunden, da Daten aus den Registern des Prozessor in den Speicher und Daten von dort in die Register geschrieben werden müssen. Besteht eine Unterroutine nur aus wenigen Befehlen, so sollten diese direkt in die aufrufende Routine integriert werden. Wird eine Folge von Befehlen mehrmals an verschiedenen Programmstellen verwendet oder leidet die Übersichtlichkeit des Programmtextes, so können in Programmiersprachen wie *C* sogenannte Makros mit diesen Befehlen definiert werden. Die Makros werden durch den Compiler vor der Übersetzung des Programms in den Programmtext eingefügt und erzeugen bei Ausführung des Programms keine Verzögerungen durch Systemoperationen.

- Konditionale Ausdrücke werden in der Reihenfolge der Programmierung abgearbeitet. Bei dem ersten nicht zutreffenden Ausdruck wird die weitere Berechnung abgebrochen. Demzufolge sollten Ausdrücke, die schnell zu berechnen sind oder häufig zu einem Abbruch führen, als erstes programmiert werden.

Zunächst soll im folgenden die generelle Struktur eines typischen Selektionsalgorithmus vorgestellt werden, wie er bei FTT-L3 zur spurbasierten Identifikation von zweistufigen Teilchenzerfällen $a \to bc \to dec$ verwendet werden wird. Wie in Abbildung 4.16 schematisch dargestellt, ist der Algorithmus prinzipiell durch drei ineinander verschachtelte Schleifen über jeweils alle Spuren charakterisiert. Durch Weglassen der innersten Schleife kann dieser generische Algorithmus auch zur Selektion reiner Zweiteilchenzerfälle $b \to de$ verwendet werden.

Vernachlässigt man die alleinigen Selektionskriterien für die erste Spur, zeigt die Suche nach dem Zerfall $b \to de$ ein Zeitverhalten, das quadratisch von der Anzahl der Spuren im Ereignis abhängt, also $\mathcal{O}(n_{Spur}^2)$ ist. Da die Selektionskriterien bei der Kombination der ersten und zweiten Spur sehr strikt sind, wird die innerste Schleife selten ausgeführt. Daher erwartet man für die Zeitentwicklung nur für den "worst case" bei sehr weichen Selektionskriterien ein Laufzeitverhalten von $\mathcal{O}(n_{Spur}^3)$, typischerweise wird aber bei $k(k \ll n_{Spur}^2)$ gefundenen Zerfällen $b \to de$ eine Abhängigkeit $\mathcal{O}(n_{Spur}^2) + \mathcal{O}(k * n_{Spur}) = \mathcal{O}(n_{Spur}^2)$ erreicht. Hier zeigt sich der Vorteil der zweistufigen Implementierung. Zusätzlich haben Tests gezeigt, daß sich durch Ansprechen der

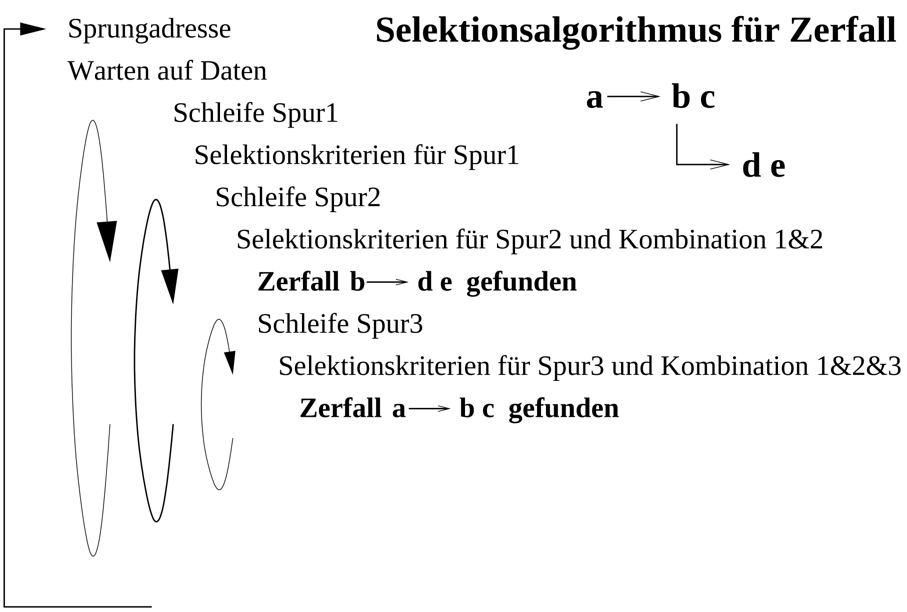



*Abbildung 4.16: Schematische Darstellung eines spurbasierten Selektionsalgorithmus für FTT-L3. Der Algorithmus ist generell zur Selektion eines zweistufigen Zerfalls $a \to bc \to dec$ ausgelegt, kann aber auch durch Weglassen der innersten Schleife für die Selektion reiner Zweiteilchenzerfälle verwendet werden. Beispiele für Selektionskriterien in den einzelnen Schleifen sind im Text gegeben.*

Datenstrukturen durch Zeiger statt durch Indizes in der gegebenen Konfiguration mit dem Compiler *gcc* ein besseres Zeitverhalten erreichen läßt.

Nachfolgend sollen die einzelnen Progammschritte genauer beschrieben werden. Außerhalb der ersten Schleife wird zunächst eine Sprungadresse definiert, an die der Task im Falle eines Abbruchs zurückgesetzt wird. Dort wird auch auf die Freigabe der Semaphoren als Zeichen der Vollständigkeit der Spur- und L2L3-Daten gewartet. Diese liegen dann schon als physikalische Größen in einer Datenstruktur vor, so daß die Selektionsalgorithmen sofort gestartet werden können.

Separat für die erste und zweite Spur kann als typisches Selektionskriterium zunächst ein minimaler Transversalimpuls gefordert werden. Durch Kombination der Spuren können dann deren Ladungsvorzeichen verglichen und Gesamt- und Transversalimpuls dieses Systems berechnet und die Einhaltung vorgegebener Schranken kontrolliert werden. Die Selektion aufgrund der invarianten Masse des Spurpaars ist am rechenaufwendigsten und sollte zum Schluß durchgeführt werden. Die invariante Masse wird dabei unter der Hypothese berechnet, daß die beiden Spuren von den Teilchen $d$ und $e$ stammen:

$$\begin{aligned} M_b &= M_{de} \\ &= \sqrt{(E_d + E_e)^2 - (\vec{p}_d + \vec{p}_e)^2} \\ &= \sqrt{m_d^2 + m_e^2 + 2E_dE_e - 2\vec{p}_d\vec{p}_e}. \end{aligned} \tag{4.8}$$

Die Selektion erfolgt durch die Forderung nach einer oberen Schranke $\delta$ für die Differenz zwischen $M_{de}$ und der theoretischen Masse des Teilchens $b$:

$$|M_{de} - m_b| < \delta \tag{4.9}$$

Die Berechnung dieser Ungleichung kann durch geschicktes Umstellen vereinfacht werden. Insbesondere kann in der Datenaufbereitung und Selektion der Gebrauch von Wurzelfunktionen vermieden werden. Um alle bei Programmstart bekannten Größen zusammenzubringen, wird diese Ungleichung nach Auflösen des Betrags durch eine Fallunterscheidung umgestellt. Durch anschließendes Quadrieren läßt sich dann die Wurzel in Gleichung 4.8 eliminieren:

$$\begin{aligned} & M_{de}^2 \leq (m_b + \delta)^2 \\ \wedge \quad & M_{de}^2 \geq (m_b - \delta)^2. \end{aligned}$$

Einsetzen und umstellen liefert

$$\begin{aligned} & \delta_1 \geq E_d E_e - \vec{p}_d \vec{p}_e \\ \wedge \quad & \delta_2 \leq E_d E_e - \vec{p}_d \vec{p}_e \end{aligned} \tag{4.10}$$

mit

$$\begin{aligned} \delta_1 &= 0.5 \cdot \left( (m_b + \delta)^2 - m_d^2 - m_e^2 \right) \\ \delta_2 &= 0.5 \cdot \left( (m_b - \delta)^2 - m_d^2 - m_e^2 \right). \end{aligned}$$

Alle Größen von $\delta_1$ und $\delta_2$ sind Konstanten, so daß diese beide Werte einmalig während der Initialisierung berechnet werden können.

Da die Berechnung der Teilchenenergien eine Wurzelfunktion erforderlich machen würde, werden die beiden Gleichungen 4.10 nochmals umgestellt und quadriert:

$$\begin{aligned} (\delta_1 + \vec{p}_d \vec{p}_e)^2 &\geq E_d^2 E_e^2 \\ (\delta_2 + \vec{p}_d \vec{p}_e)^2 &\leq E_d^2 E_e^2 \end{aligned} \tag{4.11}$$

Sind alle Selektionskriterien erfüllt, werden die beiden Spuren der äußeren und mittleren Schleife als von den Zerfallsprodukten des Teilchens $b$ stammend behandelt. Bei Bedarf kann an dieser Stelle schon wie unten beschrieben ein Triggerelement generiert werden. Unter Hinzunahme einer dritten Spur, die einem Teilchen $c$ zugeordnet wird, wird nun versucht, den Zerfall $a \to bc$ zu rekonstruieren. Zunächst werden wiederum Selektionskriterien wie die Forderung nach einem minimalen Transversalimpuls und bestimmten Kombinationen von Ladungsvorzeichen angewendet. Der Zerfall selbst wird durch Vergleich der Differenz der invarianten Massen $M_{bc}$ und $M_{de}$ mit einer oberen Schranke $\Delta_M$ detektiert:

$$\Delta_M \geq M_{bc} - M_{de} = M_{dec} - M_{de}. \tag{4.12}$$

Die Schranke $\Delta_M$ wird dabei etwas größer als die theoretische Differenz $m_a - m_b$ gewählt. Zunächst muß die invariante Masse

$$M_{de} = \sqrt{m_d^2 + m_e^2 + 2\sqrt{E_d^2 E_e^2} + 2\vec{p}_d \vec{p}_e} \tag{4.13}$$

berechnet werden. Dazu werden nur vorab bekannte oder schon berechnete Größen benötigt. Da der Ausdruck unter der Wurzel etwa dem Massenquadrat $m_b^2$ entspricht, kann man diese Formel auch schreiben als

$$\begin{aligned} M_b &= \sqrt{m_b^2 + \Delta_{m^2}} \\ &= m_b \sqrt{1 + \frac{\Delta_{m^2}}{m_b^2}} \end{aligned} \tag{4.14}$$

mit

$$\Delta_{m^2} = m_d^2 + m_e^2 + 2\sqrt{E_d^2 E_e^2} + 2\vec{p}_d \vec{p}_e - m_b^2 \tag{4.15}$$

Da typischerweise $\Delta_{m^2} \ll m_b^2$ ist, kann die Wurzelfunktion in Gleichung 4.14 durch eine Taylorentwicklung bis zu dem dritten Glied ersetzt werden, so daß schließlich für jeden Kandidaten eines Teilchens $b$ nur eine Wurzelfunktion auszuführen ist. Weiter oben wurde erwähnt, daß die Zahl der Kandidaten für ein Teilchen $b$ typischerweise kleiner als die Zahl der Spuren in einem Ereignis ist, so daß diese Vorgehensweise günstiger ist, als $E$ schon vorab für jede Spur und beide Teilchenhypothesen durch Wurzeloperationen zu berechnen. Weiterhin sind durch Additionen $E_b$ und $\vec{p}_b$ zu bilden, um schließlich die eigentliche Selektion ähnlich zu der in Formel 4.11 durchführen zu können:

$$(\Delta + \vec{p}_b \vec{p}_c)^2 \geq E_b^2 E_c^2 \tag{4.16}$$

mit

$$\Delta = 0.5 * (\Delta_M^2 - m_c^2) + \Delta_m M_b. \tag{4.17}$$

Sind alle Selektionskriterien erfüllt, terminieren alle Schleifen und die Triggerentscheidung wird mit Kennbits zur Identifikation in eine Messagequeue geschrieben. Zusätzlich kann wie oben beschrieben auch schon eine zusätzliche Triggerentscheidung nur auf Basis eines Zwischenergebnisses erwünscht sein, so daß noch während Ablauf des Selektionsalgorithmus eine Nachricht in die Messagequeue geschrieben wird. Wie in Abbildung 4.15 gezeigt, werden die Nachrichten über berechnete Triggerentscheidungen durch einen Tasks *fttL3PhysTeCompile()* aus der Messagequeue ausgelesen. Dieser schreibt durch eine Hilfsroutine die endgültige Entscheidung in ein zugeordnetes Register der Triggerbitkarte und sendet dadurch ein Triggerelement zur CTL.

#### 4.3.2.4 Datenausgabe

Um die Entscheidungen von FTT-L3 überprüfen zu können, müssen Daten aus den Rechnerkarten über die Kontrollkarte ausgelesen werden. Die Datenstruktur mit den physikalischen Daten ist eine globale Variable und kann damit auch von Tasks aus *fttL3SlaveLib* angesprochen werden. Sollen andere Größen aus *fttL3PhysLib* ausgelesen werden, müssen diese ebenfalls in globalen Variablen gespeichert sein. Sind alle Berechnungen in *fttL3PhysLib* ausgeführt und alle Daten bereitgestellt worden, wird, wie in Abbildung 4.13 gezeigt, eine Semaphore freigegeben, die die Auslese der Rechenkarte startet. Die Daten werden für das bei H1 verwendete Datenformat BOS[26] [Blo-2001] formatiert, das zusam-

[26] **B**ank **O**bject **S**ystem

menhängende Datenstrukturen in sogenannten Bänken gliedert. Im Bankkopf wird deren Name und Länge kodiert. Über ein globales Kontrollwort kann konfiguriert werden, welche Bänke die Rechenkarte für die Auslese bereitstellen soll.

Die vollständigen Bänke werden in den reservierten Bereich des Arbeitsspeichers kopiert. Informationsstrukturen an dessen Beginn zeigen die Existenz einer Bank mit Angaben der Datenlänge und Startaddresse im Speicher an. An einer weiteren Speicherstelle wird zusätzlich die Länge aller Auslesedaten inklusiver der Informationstruktur abgelegt. Eine formatierte Bank wird durch die Routine *fttL3SlaveUserMemWrite()* den Auslesedaten hinzugefügt. Zunächst wird überprüft, ob der Banktyp dem System bekannt und der Bankkopf korrekt gefüllt ist. Nachdem der Bankinhalt komplett kopiert worden ist, wird die Kontrollstruktur und die Gesamtdatenlänge aktualisiert. Sind alle durch das Kontrollwort geforderten Banken vorhanden, wird der Kontrollkarte durch Überschreiben einer bestimmten Speicherstelle im reservierten Bereich die Vollständigkeit der Auslesedaten mitgeteilt.

#### 4.3.2.5 Weitere Funktionalitäten

Zur Überwachung der Prozessorkarten sind weitere Funktionalitäten implementiert, die auf allen im FTT eingesetzten Karten Verwendung finden, an dieser Stelle aber beschrieben werden sollen.

- Zeitmessung
  Um Programmabläufe zeitaufgelöst kontrollieren zu können, wird ein Zählerregister im Hawk-Baustein ausgelesen, dessen Inhalt mit jedem vierten Takt des Prozessorbusses inkrementiert wird. Dies erlaubt eine Zeitauflösung von 40 ns. Durch die Routine *fttUtilTimeBase()* wird eine Referenzzeit festgelegt, auf die sich die Ausgabe von *fttUtilTimeGet()* bezieht. Im Falle der Rechenkarten wird die Referenzzeit beim Eintreffen des ersten Datenwortes eines Ereignisses im DPIO-FIFO aktualisiert. An folgenden Programmstellen wird anschließend die abgelaufene Zeit bestimmt:

  1. Erstes Wort eines Datenblocks im DPIO-FIFO
  2. Alle Worte eines Datenblocks im Arbeitsspeicher
  3. Physikalische Struktur eines Datentyps gefüllt
  4. Triggeralgorithmus für einen Datentyp terminiert
  5. Alle Auslesedaten (außer den Zeitdaten) bereitgestellt

  Die Zeitinformationen werden durch eine eigene BOS-Bank ausgelesen und können online oder zu jedem beliebigen Zeitpunkt ausgewertet werden.

- Nachrichtenprotokollierung
  Da die Textausgabe eines Rechners viel Zeit in Anspruch nimmt, sollte diese bei Echtzeitsystemen von den zeitkritischen Anwendungen entkoppelt erfolgen. VxWorks bietet dazu ein System an, das Nachrichten zunächst in eine Messagequeue schreibt. Ein Task mit niedriger Priorität liest diese aus und leitet die Nachrichten an eine eventuell angeschlossene Textkonsole weiter. Für den FTT wurde ein eigenes System mit erweiterter Funktionalität entwickelt, das in Abbildung 4.17 skizziert ist.

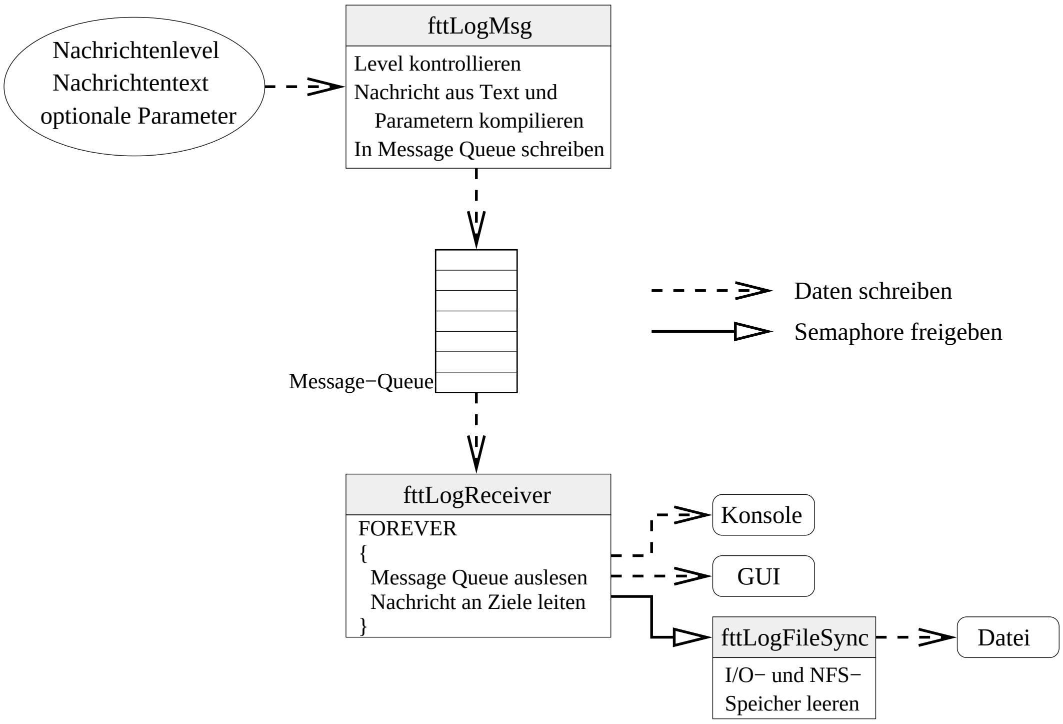



*Abbildung 4.17: Programmstruktur zur Übertragung von Kontrollnachrichten im FTT. Durch verschiedene Nachrichtenlevel wird der Umfang der Datenausgabe geregelt. Eine Messagequeue dient der Entkopplung der Tasks mit Echtzeitanforderung von der weiteren Nachrichtenbehandlung. Die Nachrichten können schließlich an mehrere Ziele geleitet werden.*

Durch die Routine *fttLogMsg()* wird eine Nachricht mit einem bestimmten Level versehen erzeugt. Nur Nachrichten oberhalb eines einstellbaren Levels werden anschließend auch ausgegeben. Folgende Level sind möglich:

- `FTT_LOG_ERROR`
  Nachricht über Fehler, die die Funktionalität des Systems stark beeinträchtigen, so daß ein Neustart der Datennahme oder des gesamten Systems notwendig wird.
- `FTT_LOG_WARNING`
  Nachricht über Fehler, die die Datennahme eines Ereignisse beeinträchtigen, das System aber in einem funktionsfähigen Zustand belassen.
- `FTT_LOG_INFO_1`
  Nachricht über Bearbeitung eines Programmpunktes, der typischerweise außerhalb der Datennahme und damit nicht periodisch erreicht wird. Beispielsweise sei hier die Initialisierungsphase beim Einschalten von Komponenten erwähnt, während der Informationen über die aktuelle Konfiguration ausgegeben werden.
- `FTT_LOG_INFO_2`
  Nachrichten, die periodisch während der Datennahme erzeugt werden. Dies

beinhaltet zum Beispiel die Ausgabe der Zählerstände für STC-Signale bei Beendigung der Datennahme.

– `FTT_LOG_INFO_3`
  Nachrichten, die periodisch für einzelne Ereignisse ausgegeben werden, um den Programmablauf schrittweise nachvollziehen zu können. Diese Nachrichten erzeugen eine umfangreiche Datenausgabe und sollten nur für Testzwecke eingesetzt werden. Um das Zeitverhalten des Systems nicht durch zahlreiche Aufrufe der Unterroutine *fttLogMsg ()* zu beeinträchtigen, wird schon durch den Compiler das Erzeugen von Nachrichten dieses Levels nur in einer speziellen Konfiguration zugelassen.

In *fttLogMsg ()* wird zunächst anhand des Nachrichtenlevels kontrolliert, ob die Eingabe bearbeitet werden soll. Ist dies der Fall, werden optional mitgelieferte Parameter in den Nachrichtentext integriert und dieser mit aktuellem Datum und Uhrzeit markiert, um nachfolgend in eine Messagequeue geschrieben zu werden.

Ein Task mit niedriger Priorität liest die Messagequeue aus und kann die Nachricht an drei Ziele weiterleiten:

– Textkonsole
  WindRiver liefert eine Konsole mit, auf der die Textnachrichten einer Prozessorkarte ausgegeben werden. Diese Konsole bietet nicht die Möglichkeit, die Nachrichten zu speichern.
– Graphische Benutzeroberfläche
  Zur Steuerung und Überwachung des FTT wurde eine graphische Benutzeroberfläche entwickelt [Bau-2002], die auf einem Steuerungsrechner gestartet wird und über eine Netzwerkverbindung mit den Prozessorkarten kommuniziert. Eine integrierte Funktionalität ist das Empfangen und Darstellen der Nachrichten.
– Protokolldatei
  Zur dauerhaften Protokollierung kann die Textausgabe in eine Datei auf einem Steuerungsrechner gelenkt werden, auf die die Prozessorkarte über NFS zugreifen kann. Dieser Datentransfer ist über einen generellen Ausgangspuffer und einen NFS-Puffer zweifach entkoppelt. Um nach dem Absetzen einer Nachricht in den Ausgangspuffer ein zeitnahes Schreiben der Nachrichten in die Datei zu erreichen, werden volle Puffer etwa sekündlich durch einen Task mit niedriger Priorität geleert.

### 4.3.3 Kontrollkarte

Zur zentralen Initalisierung, Überwachung und Auslese von FTT-L3 wird eine weitere Prozessorkarte als Kontrollkarte in das System integriert. Die Verbindung zu den anderen Karten wird ausschließlich über den VME-Bus hergestellt.

#### 4.3.3.1 Initalisierung

Nach dem Einschalten des Systems versucht die Kontrollkarte, auf allen möglichen VME-Adressen Rechenkarten anzusprechen, um daraus die aktuelle Konfiguration ableiten zu können. Gefundene Rechenkarten bekommen die Adresse von einem oder mehreren Registern in der Triggerbit-Karte mitgeteilt, an die sie die jeweilige Triggerentscheidung zu senden haben. Die Bedeutung jedes Registers muß zentral verwaltet und mit dem zentralen Trigger synchronisiert werden. Außerdem werden die Triggerbit- und Empfängerkarten initialisiert. Schließlich wird durch einen im Abschnitt 4.3.3.3 beschriebenen Systemtest die Integrität des L3-Datentransfers überprüft.

#### 4.3.3.2 Auslesen der Prozessorkarten

Wie oben beschrieben, stellt jede Rechenkarte Daten zur Auslese in dem reservierten Speicherbereich bereit und meldet deren Vollständigkeit durch Überschreiben einer bestimmten Speicherstelle in diesem Bereich. Die Kontrollkarte wertet reihum diese Speicherstelle auf jeder Rechenkarte aus. Ist eine Karte bereit, wird die Größe des auszulesenden Datenblocks ermittelt und dieser durch einen VME-DMA-Zugriff in den Arbeitsspeicher der Kontrollkarte gelesen. Sind die Daten aller Karten vorhanden, werden aus ihnen globale FTT-L3-Bänke im BOS-Format erstellt. Die weitere Behandlung der Auslesedaten wird in Anhang A beschrieben.

Durch das H1-Datennahmesystem ist eine Auslesezeit in der Größenordnung von 1 ms vorgegeben, bei deren Überschreitung eine nicht mehr zu tolerierende Totzeit eintreten würde. Die Kontrollkarte überprüft daher die Zeit zwischen einem **L2Keep** und der Vollständigkeit aller Auslesedaten auf jeder Rechenkarte. Überschreitet diese 1 ms, wird auf einem Steuerungsrechner im Kontrollraum eine Warnmeldung erzeugt. Hat eine Rechenkarte nach mehr als 10 ms noch keine Daten bereitgestellt, wird ein schwerwiegender Fehler angenommen und eine entsprechende Meldung generiert. Die Auslese wird zunächst ohne diese Karte fortgesetzt. Zeigt der nachfolgend beschriebene Systemtest einen Fehler, muß die Datennahme komplett gestoppt und FTT-L3 neu gestartet werden.

#### 4.3.3.3 Überwachung des Systems

Um die einwandfreie Funktion von FTT-L3 überprüfen zu können, wurde ein Systemtest entwickelt. Die Kontrollkarte initiiert dazu eine geschlossene Kette von Datenübertragungen innerhalb FTT-L3, deren Ablauf in Abbildung 4.18 gezeigt ist. Zunächst wird ein Testmuster inklusive der 16 Kennbits der FTT-weiten Datenübertragung generiert und in den VME-FIFO der Empfängerkarte geschrieben. Anschließend wird durch einen Schreibbefehl in das Befehlsregister der Karte der Transfer über die FPDP-Verbindung zu den Rechenkarten ausgelöst. Dort muß das erste Wort als Startwort der Testdaten erkannt und ein entsprechender DMA-Transfer gestartet werden. Aus dem Arbeitsspeicher werden die Testdaten in eine BOS-Struktur mit entsprechenden Kontrollwörtern in den reservierten Bereich geschrieben und den Rechenkarten die Bereitstellung der Daten gemeldet. Nachdem die Daten auf die Kontrollkarte zurückgelesen worden sind, findet ein Vergleich mit dem ursprünglichen Testmuster statt. Verläuft dieser für alle Rechenkarten fehlerfrei, gilt der Test als erfolgreich. Anderenfalls wird das Nachrichtensystem eine Fehlermeldung ausgegeben. Der Systemtest wird nach Einschalten des Systems und während angehaltener Datennahme sekündlich durchgeführt. Zusätzlich kann er durchgeführt werden, falls während der Datennahme das Auftreten eines schwerwiegenden Systemsfehlers zu verifizieren ist.

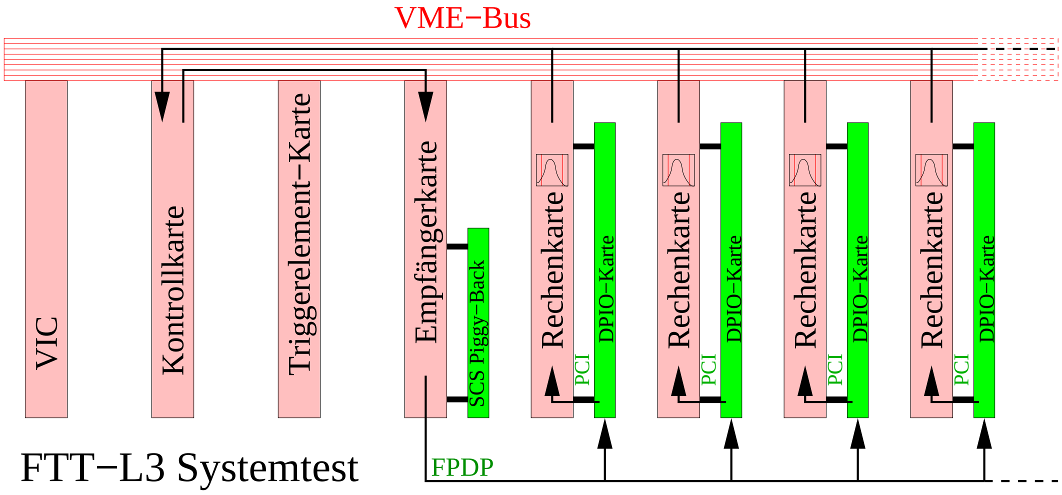



*Abbildung 4.18: Schematische Darstellung des Testdatenflusses innerhalb von FTT-L3. Ein auf der Kontrollkarte generiertes Testmuster wird über die Empfängerkarte und die FPDP-Verbindung zu den Rechenkarten gesendet. Dort werden die Daten für die Auslese durch die Kontrollkarte aufbereitet. Die Kontrollkarte verifiziert nach dem Zurücklesen schließlich deren Übereinstimmung mit dem ursprünglichen Testmuster.*

## 4.4 Systemverhalten

### 4.4.1 Testaufbau

Ein Testaufbau erlaubt die Bestimmung einiger Kenngrößen des Systemverhaltens. Die Programmierung der Kontrollkarte wurde dazu um einige Funktionalitäten erweitert, die es erlauben, Daten aus einer Datei in das System einzulesen und Zyklen von STC-Signalen zu generieren. Um auch ohne Empfängerkarte, die sich Ende 2002 erst in der Inbetriebnahmephase befindet, das L3-System testen zu können, wurde eine Prozessorkarte mit einer DPIO-Ausgangskarte bestückt, die sich im wesentlichen durch die Datenflußrichtung von den ansonsten benutzen Eingangskarten unterscheidet. Diese Prozessorkarte emuliert durch ihre Programmierung die Funktionalität der Empfängerkarte und wird daher im folgenden als Empfängeremulator bezeichnet. Die Kontrollkarte spricht den Empfängeremulator über Routinen an, die leicht modifiziert denen entsprechen, die für die Empfängerkarte vorgesehen sind und das Schreiben von 48 Bits breiten Datenworten und das Auslösen von FPDP-Datentransfers und PIO-Signalen erlauben. Der FPDP-Bus verbindet den Empfängeremulator schließlich mit drei Rechenkarten. Der vollständige Testaufbau ist in Abbildung 4.19 gezeigt.

### 4.4.2 Laufzeitverhalten

Die Stabilität des Gesamtsystems kann mit Hilfe des L3-Systemtests überprüft werden. Dabei hat sich herausgestellt, daß durch ein fehlerhaftes Bauteil auf der DPIO-Ausgangskarte die Bits 8, 10, 12 und 14 der übertragenen Daten dauerhaft gesetzt sind. Dies ist bei der Fehlerauswertung im Testsystem zu berücksichtigen. Die Dauer eines

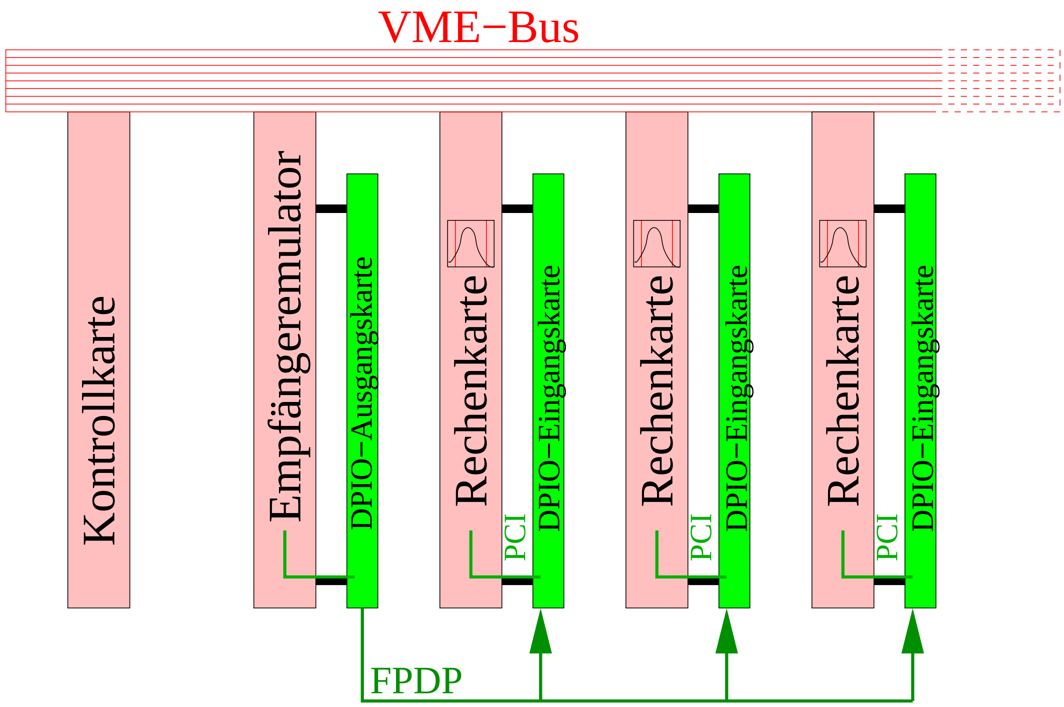



*Abbildung 4.19: FTT-L3-Testaufbau zur Bestimmung von Systemkenngrößen. Da zunächst keine Empfängerkarte zu Verfügung steht, wird diese durch eine Prozessorkarte mit aufgesteckter DPIO-Ausgangskarte und entsprechender Programmierung emuliert. In die Programmierung der Kontrollkarte wurden zusätzliche Datentransfermöglichkeiten zu einem Steuerungsrechner implementiert, um Testdaten in das System ein- und Ergebnisse auslesen zu können.*

Testdurchlaufs beträgt etwa 400 $\mu$s, so daß der Test mit einer Wiederholrate von 1 kHz sicher durchgeführt werden kann. Während verschiedener Testläufe wurde kein Fehler beobachtet. Der längste Test dauerte knapp acht Tage und wurde nach $N_{Test} = 689,32 * 10^6$ fehlerfreien Durchläufen abgebrochen. Damit kann eine Fehlerwahrscheinlichkeit von

$$W_{Fehler} \leq \frac{1}{N_{Test}} = (1,45071 \pm 0,00006) \cdot 10^{-9} \tag{4.18}$$

für die gesamte Datenübertragungskette innerhalb von FTT-L3 angegeben werden.

Desweiteren wurde die Möglichkeit genutzt, Testdaten aus einer Datei über die Kontrollkarte in das System einzuspeisen. Für die Tests wurde ein Rohdatensatz von 20000 Ereignissen verwendet, die am 3.9.2002 während $e$-$p$-Kollisionen mit eingeschalteten Spurkammern aufgezeichnet wurden. Die Bitfehler der FPDP-Übertragung beeinflussen den Wert einer Gleitkommazahl dabei nur in wenig signifikanten Dezimalstellen, so daß das prinzipielle Zeitverhalten der Selektionsalgorithmen hinreichend gut bestimmt werden konnte. Als zeitlicher Bezugspunkt wird der Moment gewählt, in dem das erste Datenwort im FIFO der DPIO-Karte erkannt wird und die benötigte Zeit bis zu den in Abschnitt 4.3.2.5 beschriebenen Punkten 2, 3 und 4 gemessen. Die Tests wurden parallel auf den drei Rechenkarten durchgeführt, die keine signifikante Abweichungen im

Zeitbedarf zeigten. Nacheinander wurden die Selektionsalgorithmen für $J/\Psi$- und $D^*$-Mesonen überprüft. Die Ergebnisse sind in Abbildung 4.20 abhängig von der Zahl der Spuren im Ereignis dargestellt.

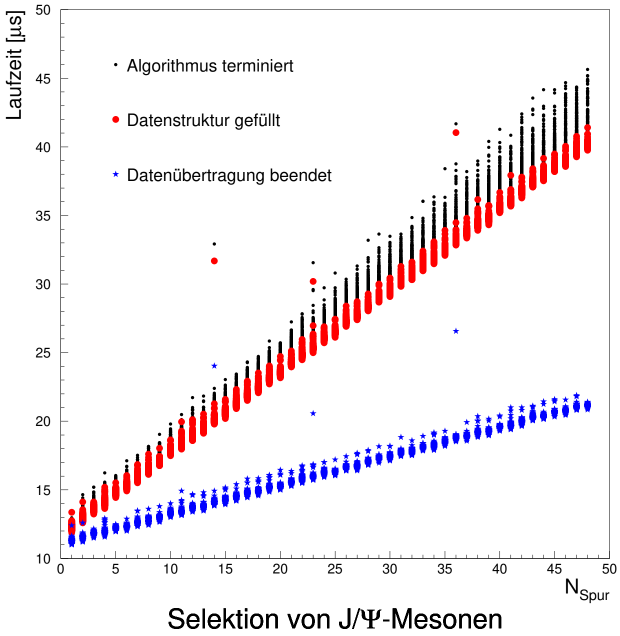
Bei der Suche nach $J/\Psi$-Kandidaten zeigen die Meßpunkte für die Zeit, die Daten in den Speicher zu schreiben und die Daten aufzubereiten, wie erwartet eine lineare Abhängigkeit von der Zahl der Spuren. Der anfängliche Versatz von etwa 10 $\mu$s ist wie in Abschnitt 4.3.2.1 beschrieben dadurch zu erklären, daß bei Beendigung des Datentransfers ein Interrupt bearbeitet werden muß, zwei Taskwechsel auftreten und zusätzlich in dieser Zeit die Eingangsdaten auf ihre Konsistenz überprüft werden. Die Datenaufbereitung nimmt maximal 40 $\mu$s in Anspruch, der gesamte Zeitbedarf bis zu einer Entscheidungsfindung liegt für die maximale Spurzahl bei unter 50 $\mu$s. Da auch bei einer finalen negativen Triggerentscheidung nicht alle Schleifen komplett durchlaufen werden, sondern teilweise durch Abbruchbedingungen vorzeitig terminieren, läßt sich keine quadratische Abhängigkeit von der Spurzahl erkennen. Die theoretische $n^2_{Spur}$-Abhängigkeit ist also als "worst case" anzusehen.

Bei der Suche nach $D^*$-Kanditaten ändert sich der Zeitbedarf für die Datenübertragung und Datenaufbereitung selbstverständlich nicht. Der gesamte anschließende Zeitbedarf nimmt jedoch drastisch zu, da bis zu einer positiven Entscheidung nun eine zusätzliche dritte Schleife mindestens einmal durchlaufen werden muß. Zwar werden dadurch Extremwerte von etwa 250 $\mu$s erreicht, Abbildung 4.21 zeigt allerdings, daß mit der untersuchten Selektion nach dem in Abschnitt 4.4.3 beschriebenen Szenario "FTT(2)" nur 325 Ereignisse die Zeitvorgabe von 100 $\mu$s nicht erfüllen. Dies entspricht $1,6 \pm 0,1$ % der getesteten Ereignisse. Da durch die CTL jedes nach 100 $\mu$s nicht eingegangene TE als "negativ" bewertet wird, brauchen in Wirklichkeit nur die 160 Ereignisse berücksichtigt zu werden, in denen nach mehr als 100 $\mu$s noch ein $D^*$-Kandidat gefunden wird. Somit detektiert man durch Überschreiten der erlaubten Rechenzeit nur $0,80 \pm 0,06$ % der $D^*$-Kandidaten nicht. Es bleibt zu untersuchen, in wievielen dieser Ereignissen auch durch eine vollständige Analyse ein $D^*$-Kandidat gefunden werden würde und wieviele zufällige Untergrundereignisse sind. Desweiteren wird es für die Bestimmung der Triggereffizienzen wichtig werden, das Zeitverhalten der Selektionsalgorithmen auch außerhalb des eigentlichen FTT-L3-Systems an Arbeitsplatzrechnern untersuchen und Überschreitungen des Zeitrahmens exakt messen zu können. Zu diesem Zweck werden Untersuchungen durchgeführt, den Zeitbedarf durch entsprechende Zeitkalibrationen an diesen Rechnern bestimmen zu können [Win-2002b].

Zu Kontrollzwecken wurden zusätzlich die invarianten Massen, respektive deren Differenzen, die zu einer positiven Entscheidung führten, ausgelesen. Abbildung 4.22 zeigt die resultierenden Verteilungen, wobei durch die hier ausgeführten Selektionsschnitte noch keine deutlichen Signaturen für die jeweiligen Zerfälle im Spektrum zu sehen sind. Dies hat mehrere Gründe:

- Die Massen wurden aus der Kombination von Spuren berechnet, die als erste, aber nicht unbedingt als beste die Selektionsschnitte erfüllt, wie es in Analysen gemacht wird.

- Die Testdaten wurden während der Inbetriebnahmephase von HERA-II aufgenommen, als die Datennahme durch starken Untergrund dominiert wurde. Daher ist der Anteil physikalischer Ereignisse in den Daten möglicherweise klein.



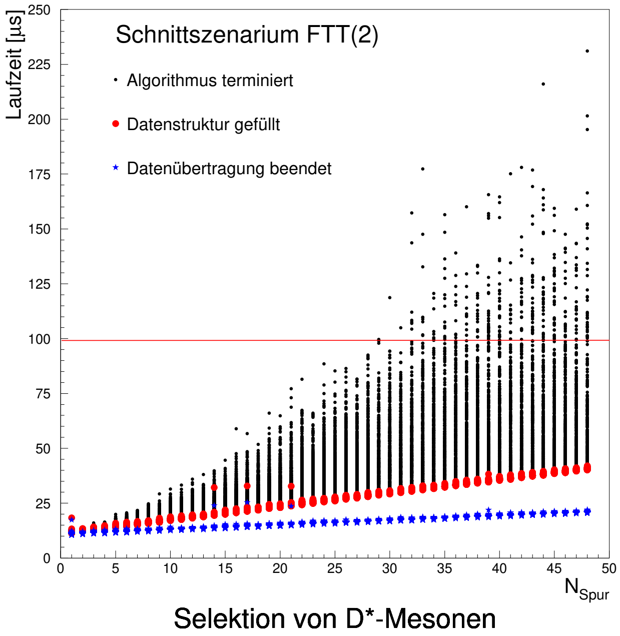




*Abbildung 4.20: Zeitbedarf der Rechenkarte zwischen Erkennen des ersten Datenwortes im FIFO der DPIO-Karte und weiteren Ereignissen während der Datenverarbeitung in Abhängigkeit von der Zahl der Spuren in einem Ereignis. Die Untersuchung wurde für Algorithmen zur Selektion von $J/\Psi$- und $D^*$-Mesonen durchgeführt. Die waagerechte Linie in der untereren Abbildung bezeichnet die durch die CTL vorgegebene, maximale Laufzeit von 100 µs.*

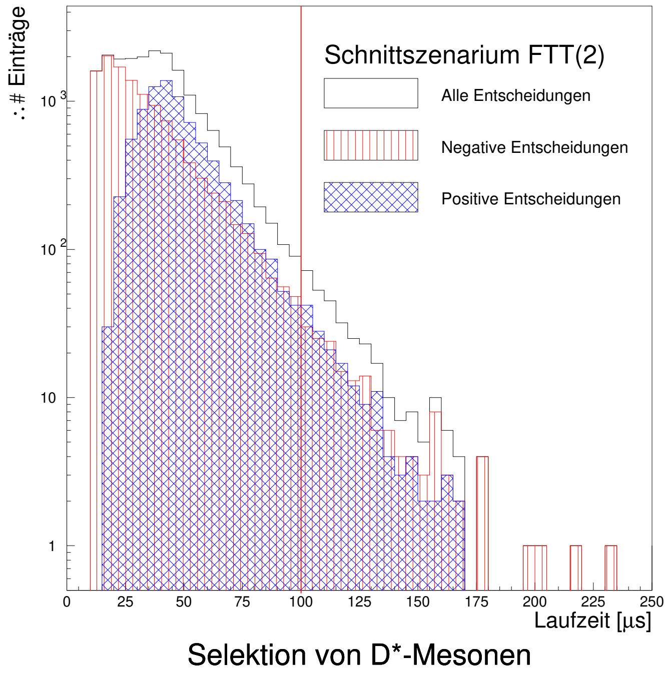



*Abbildung 4.21: Zeitbedarf der Rechenkarte zwischen Erkennen des ersten Datenwortes im FIFO der DPIO-Karte und der Terminierung des Selektionsalgorithmus. Die schraffierten Flächen zeigen gesondert die Verteilungen für Ereignisse mit einer negativen, beziehungsweise positiven Triggerentscheidung. Die senkrechte Linie bezeichnet die durch die CTL vorgegebene, maximale Laufzeit von 100 µs..*

- Der genaue Einfluß der Bitfehler wurde nicht weiter untersucht, da vorrangig prinzipielle Systemuntersuchungen gemacht werden sollten.

### 4.4.3 Selektionsvermögen

Ereignisse mit $D^*$-Mesonen wurden bei HERA-I durch inklusive DCR$\varphi$- und $z$-Vertex-Triggerelemente in Verbindung mit dem Nachweis eines Elektrons selektiert. Für Prozesse der tiefinelastischen Streuung wurde mit einem SpaCal-TE der ST61, für Photoproduktionsprozesse mit Signalen der Elektron-Tagger der ST83 (ET33), ST84 (ET44) und ST108 (ET8) gebildet. Abbildung 4.23 zeigt die L1-Raten dieser Subtrigger (ST) während einer typischen Datennahmeperiode an der HERA-I-Maschine im Jahr 2000. Wie beschrieben, ist bei Erreichen der vollen Leistungsfähigkeit mit einer etwa fünffachen Rate bei HERA-II zu rechen. Vor allem die STs, die eine Triggerbedingung auf Basis der Elektron-Tagger enthalten, führen durch starken, strahlinduzierten Untergrund zu Triggerraten, die einer weiteren Reduzierung bedürfen.

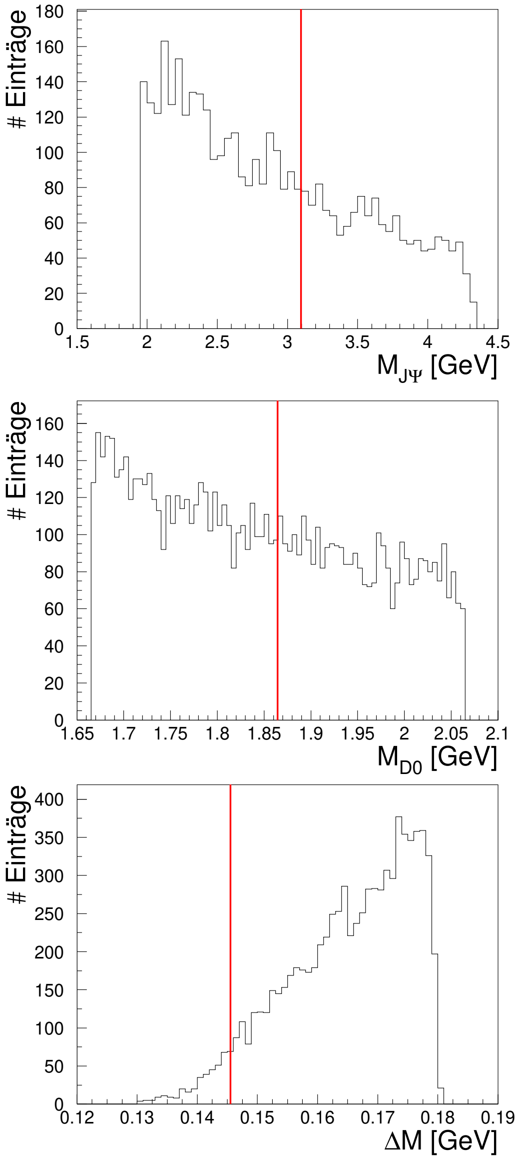



*Abbildung 4.22: Invariante Massen, beziehungsweise deren Differenz, die zur Selektion von Meson-Kandidaten aus Testdaten führten. Oben ist die Verteilung für $J/\Psi$-Kandidaten zu sehen. Unten ist die Massendifferenz $\Delta_M$ zur Selektion von $D^*$-Mesonen gezeigt. Die dabei eingehende invariante Masse des $D^0$-Kandidaten zeigt die mittlere Abbildung. Durch Linien ist jeweils der theoretische Wert markiert.*

Das zusätzliche Selektionsvermögen von FTT-L3 für Ereignisse mit $D^*$-Mesonen wurde daher für verschiedene Szenarien untersucht [Wag-2002]. Tabelle 4.4 faßt die drei untersuchten Kombinationen von Selektionsschnitten zusammen. Als Eingangsdaten dienten reale Daten aus der Datennahmeperiode 2000 sowie durch das Monte-Carlo-Programm “AROMA” generierte. Die ersten beiden Stufen des FTT wurden durch FT-

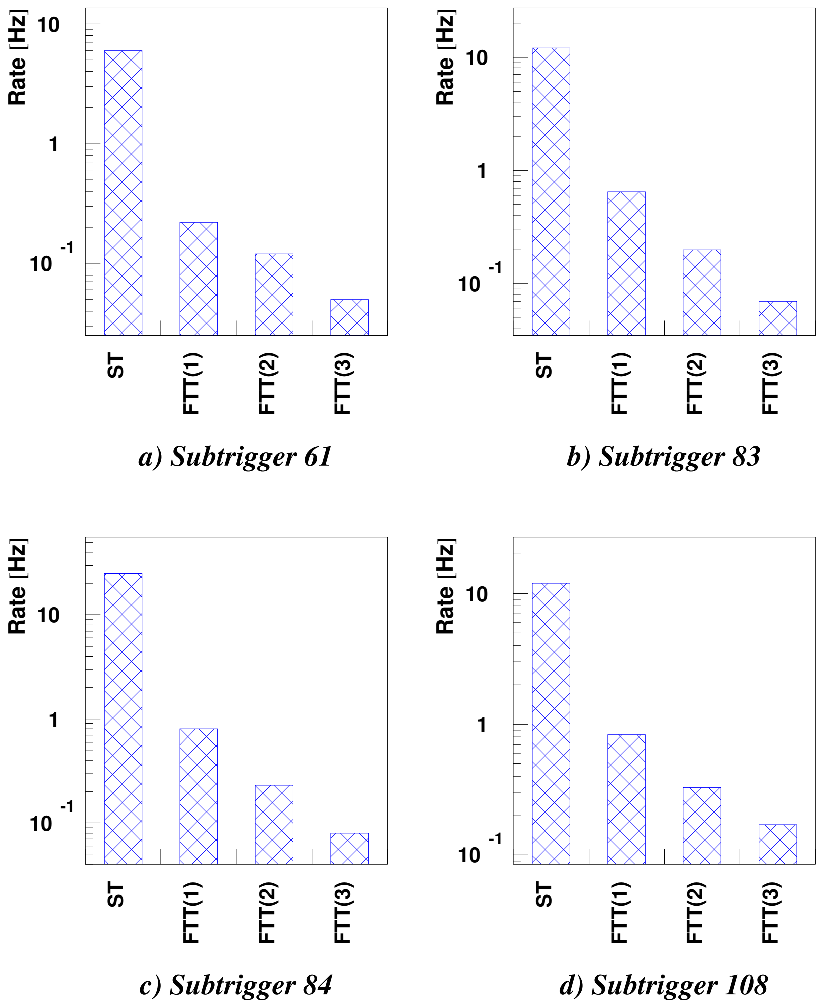



*Abbildung 4.23: Durch die FTT-Simulation erreichbare Raten in Bezug auf im Jahr 2000 bestimmte Subtriggerraten [Wag-2002]. Es handelt sich um inklusive Trigger zur Selektion von DIS- (a)), beziehungsweise Photoproduktionsereignissen (b) - d)). Die verwendeten Schnittszenarien* FTT(i) *durch FTT-L3 sind in Tabelle 4.4 aufgelistet.*

TEMU3 simuliert, die Schnitte der dritten Stufe in die Analysekette integriert.

Aus der bekannten Rate $R_{ST}$ der STs und der Zahl der Ereignisse ohne ($N_{ST}$) und mit ($N_{ST,FTT}$) zusätzlichen FTT-L3-Bedingungen läßt sich die zu erwartende Ereignisrate $R_{ST,FTT}$ für die betrachteten Szenarien abschätzen:

$$R_{ST,FTT} = R_{ST} \frac{N_{ST,FTT}}{N_{ST}} \tag{4.19}$$

Abbildung 4.23 zeigt die deutliche Ratenreduktion, die schon bei dem Szenario FTT(1)

| | FTT(1) | FTT(2) | FTT(3) |
|---|---|---|---|
| $p_{t,D^*}$ | $> 1{,}2$ GeV | $> 1{,}8$ GeV | $> 2{,}2$ GeV |
| $p_{t,K,\pi}$ | $> 0{,}25$ GeV | $> 0{,}3$ GeV | $> 0{,}4$ GeV |
| $p_{t,\pi_{slow}}$ | $> 0{,}1$ GeV | $> 0{,}1$ GeV | $> 0{,}13$ GeV |
| $\theta(D^*)$ | $20^o < \theta(D^*) < 160^o$ | $20^o < \theta(D^*) < 160^o$ | $20^o < \theta(D^*) < 160^o$ |
| $\delta_{D^0}$ | $< 0{,}25$ GeV | $< 0{,}2$ GeV | $< 0{,}2$ GeV |
| $\Delta_M$ | $< 0{,}18$ GeV | $< 0{,}18$ GeV | $< 0{,}17$ GeV |

*Tabelle 4.4: Schnittszenarien bei der Simulation von FTT-L3 geordnet nach steigendem Selektionsvermögen [Wag-2002].*

| | Offline(1) | Offline(2) |
|---|---|---|
| $p_{t,D^*}$ | $> 1{,}5$ GeV | $> 2{,}0$ GeV |
| $p_{t,K,\pi}$ | $> 0{,}3$ GeV | $> 0{,}5$ GeV |
| $p_{t,\pi_{slow}}$ | $> 0{,}12$ GeV | $> 0{,}12$ GeV |
| $\theta(D^*)$ | $20^o < \theta(D^*) < 160^o$ | $20^o < \theta(D^*) < 160^o$ |
| $\delta_{D^0}$ | $< 0{,}06$ GeV | $< 0{,}06$ GeV |
| $\Delta_M$ | $< 0{,}17$ GeV | $< 0{,}17$ GeV |
| $L_{Spur,K,\pi}$ | $> 17$ cm | $> 17$ cm |
| $L_{Spur,\pi_{slow}}$ | $> 10$ cm | $> 10$ cm |

*Tabelle 4.5: Typische Schnittszenarien in der Offline-Analyse zur Selektion von $D^*$-Kandidaten [Wag-2002]. Offline(1) wird zur Bestimmung der Effizienz des FTT bei DIS-Ereignissen, Offline(2) bei Photoproduktionsereignissen verwendet. Die Bedeutung der einzelnen Größen ist im Text erklärt.*

bei allen betrachteten Subtriggern mehr als ein Faktor zehn beträgt. Während bei ST61 schon mit dem Szenario FTT(1) eine Rate von etwa 0,2 Hz zu erreichen ist, führt bei der Selektion von Photoproduktionsereignissen erst das härtere Szenario FTT(2) zu vergleichbaren Ergebnissen, ohne jedoch die Effizienz zu verringern.

In Tabelle 4.5 werden zwei Schnittszenarien vorgestellt, wie sie bei der Offline-Selektion von $D^*$-Zerfällen bei DIS-, beziehungsweise Photoproduktionsereignissen Verwendung finden. Im Vergleich zur Selektion durch FTT-L3 können dort auf Grund der besseren Auflösung der Meßgrößen nach der vollständigen Ereignisrekonstruktion härtere Schnitte durchgeführt werden. Zusätzlich stehen die gemessenen Spurlängen $L_{Spur}$ der identifizierten Teilchen als Selektionskriterium zu Verfügung. Vergleicht man die Zahl $N_{D^*}$ gefundener $D^*$-Kandidaten mit und ohne zwischengeschalteter FTT-Selektion, läßt sich eine Effizienz für den FTT angeben:

$$\epsilon(FTT(i)) = \frac{N_{D^*}(ST, FTT(i), Offline(i))}{N_{D^*}(ST, Offline(i))} \tag{4.20}$$

In Abbildung 4.24 sind die resultierenden Effizienzen für die verschiedenen Datensätze und Selektionsszenarien angegeben. Trotz der deutlichen Reduktion der Ereignisrate liegt die Effizienz des FTT durch die präzisen Selektionsmöglichkeiten bei 80–90% bezo-

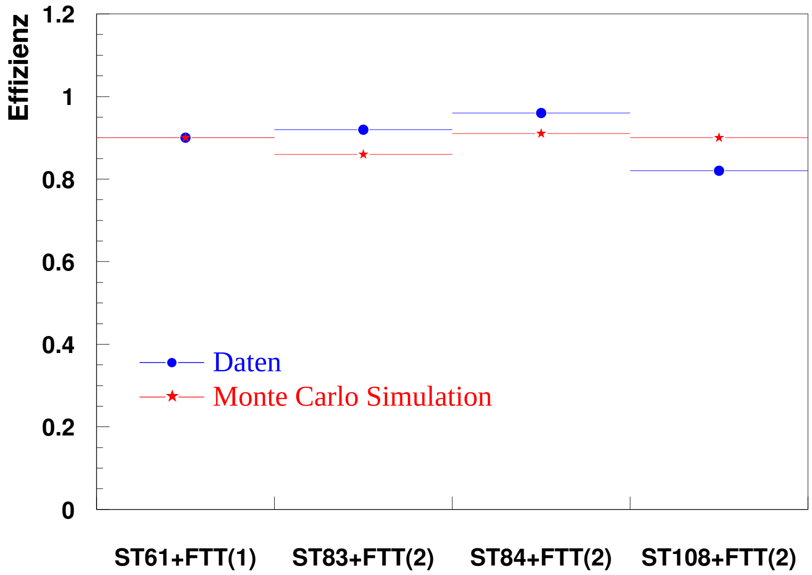



*Abbildung 4.24: Effizienzbestimmung durch Vergleich der Zahl von $D^*$-Kandidaten aus einer vollständigen Analysekette mit und ohne zwischengeschalteter FTT-Simulation [Wag-2002]. Gezeigt ist die Effizienz für reale und simulierte DIS- (ST61) und Photoproduktionsereignisse (ST83, ST84, ST108).*

gen auf auf die aufwendige Selektion durch die vollständige Offline-Analysekette..

### 4.4.4 Ansätze zur Optimierung und Weiterentwicklung

Während der Entwicklung und des Aufbaus des bisher beschriebenen FTT-L3-Systems wurden schon Möglichkeiten erkannt, wie Funktionalität und Leistung weiter zu steigern sind. Zum einen ist dies durch neue Entwicklungen in der verfügbaren Elektronik möglich. Zum anderen zeigte sich recht früh, daß für bestimmte Selektionskanäle ausreichend Zeit bleibt, neben den Spurdaten weitere Daten zu bearbeiten. Dies floß direkt in die Entwicklung ein, indem der Empfang der L2L3-Daten vorgesehen wurde.

#### 4.4.4.1 Verarbeitung weiterer Daten

Welche weiteren Möglichkeiten der Datenanalyse sich durch die L2L3-Daten bieten, soll nachfolgend an Hand von drei Beispielen kurz beschrieben werden.

- Bestimmung der Mandelstam-Variablen $t$ aus SpaCal-Triggerdaten
  Vernachlässigt man in Formel 1.5 die Protonmasse und fordert, daß die longitudinale Impulskomponente des gestreuten Protons sehr viel größer als die transversale

ist, gilt näherungsweise:

$$t \approx -\left(\vec{P}_t'\right)^2 = -\left(\vec{p}_{t,e'}' + \vec{p}_{t,J/\Psi}'\right)^2. \quad (4.21)$$

Für Photoproduktionsereignisse kann der Transversalimpuls des unter einem sehr kleinen Winkel gestreuten Elektrons vernachlässigt werden, so daß es möglich ist, $t$ allein aus dem Transversalimpuls der $J/\Psi$-Zerfallsprodukte, in diesem Fall zwei Myonen, mit ausreichender Genauigkeit zu berechnen. Dies kann durch die Abbildungen 4.25 (a und b) belegt werden, in denen eine generierte $t$-Verteilung mit einer durch FTTEMU, beziehungsweise der vollen H1-Rekonstruktionskette ermittelten verglichen wird [Beh-2002].

Die Abbildungen 4.25 (c und d) zeigen die Ergebnisse für DIS-Ereignisse, bei denen SpaCal-IET-Informationen verwendet werden, um den Streuwinkel $\theta_{e'}$ des Elektrons zu ermitteln. Der Transversalimpuls des Elektrons läßt sich dann mit zusätzlichen Energie- und Impulsinformationen der beiden Myonkandidaten näherungsweise berechnen durch

$$p_{t,e'} = E_{e'} \sin\theta_{e'} \approx \frac{\sin\theta_{e'}}{1-\cos\theta_{e'}} \left(2E_e - \left(E_{\mu^+} - p_{z,\mu^+}\right) - \left(E_{\mu^-} - p_{z,\mu^-}\right)\right), \quad (4.22)$$

wobei $E_e$ die Energie des Elektronstrahls bezeichnet. Die Fluktuationen zu großen Werten von $-t$ lassen sich durch Auflösungseffekte im steil abfallenden Spektrum erklären.

Für Photoproduktionsereignisse ist es möglich, die Rate um einen Faktor 10 mit einer Selektionseffizienz von etwa 95% im Bereich von $0,1 < -t_{cut,FTT} < 1,0$ GeV zu reduzieren [Beh-2002]. Aufgrund der schlechteren Auslösung fällt die Selektionseffizienz bei DIS-Ereignissen für große $-t_{cut,FTT}$ auf etwa 75% ab. Dort ist allerdings eine Ratenreduktion um den Faktor 100 möglich.

- Spurkorrelation zwischen FTT und zentralem Myonsystem
  Zur Generierung von L1-Subtriggern wird das zentrale Myonsystem nur grob in 4 räumliche Bereiche unterteilt. Die PQZP-Daten beinhalten hingegen eine feinere Granulatität mit Trefferinformationen von 64 Modulen. Zur effektiven Selektion von Teilchenzerfällen mit assoziierter Produktion von Myonen ist es wünschenswert, Spuren, die als Myonkandidat behandelt werden, in lokale Koinzidenz mit Treffern im Myonsystem zu bringen. Dazu wird aus den Parametern einer Spur der wahrscheinliche Auftreffort im zentralen Myonsystem berechnet und dort innerhalb eines Suchfensters, das die Krümmung der Spur im Magnetfeld berücksichtigt, nach Treffern gesucht [Mer-2002]. In Abbildung 4.26 sind die Effizienzen für zwei Selektionsszenarien und verschiedene L1-Subtrigger, die alle auf Spurinformationen im zentralen Myonsystem basieren, dargestellt. Die als "FTT" bezeichnete Selektion fordert für Myonkandidaten einen maximalen Transversalimpuls von $p_{t,max} = 0,7$ GeV und ein invariante Masse in einem Fenster von $\pm 1,1$ GeV um

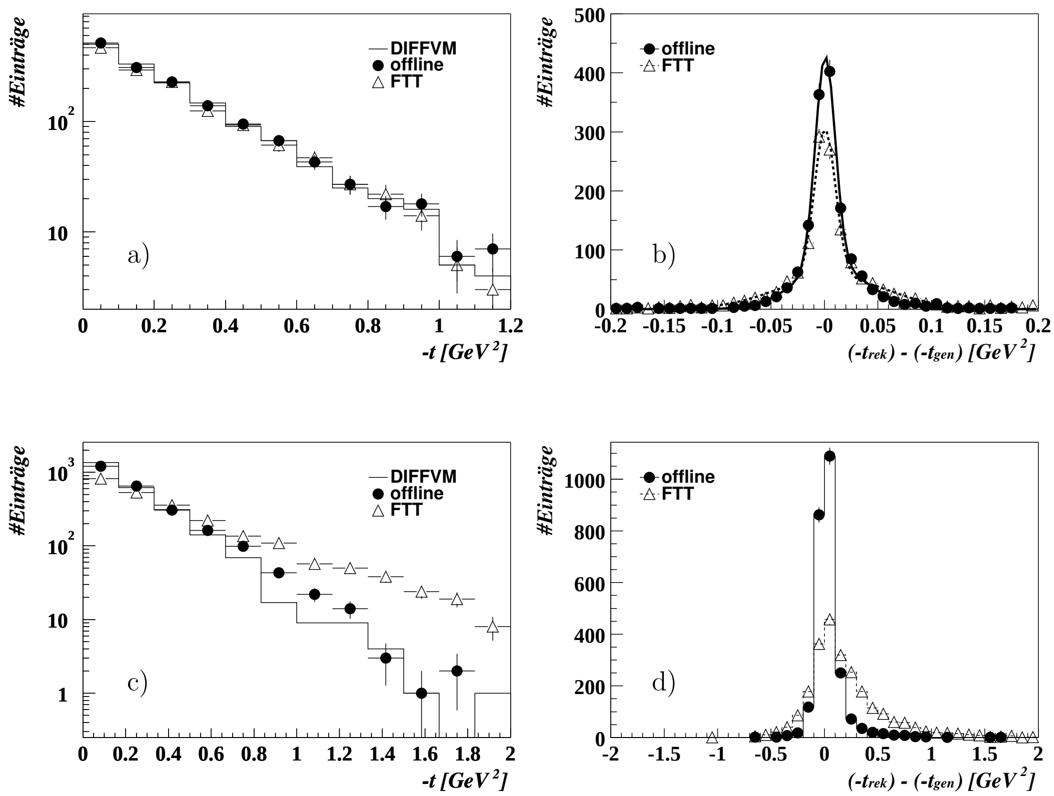



*Abbildung 4.25: Bestimmung der Mandelstam-Variablen $t$ durch FTT-L3 für Monte-Carlo-Ereignisse mit einem Zerfall $J/\Psi \rightarrow \mu\mu$ [Beh-2002]. Bei Photoproduktionsereignissen (a,b) kann $t$ mit ausreichender Genauigkeit aus dem Transversalimpuls der beiden Myonspuren bestimmt werden, während für DIS-Ereignisse (c,d) zusätzlich der Transversalimpuls des gestreuten Elektrons berücksichtigt werden muß. Dieser wird näherungsweise aus den SpaCal-Triggerdaten berechnet. Gezeigt sind jeweils die generierten und durch die volle Rekonstruktion und FTT-L3 rekonstruierten Spektren, sowie die erreichte Auflösung in Bezug auf den generierten Wert.*

die nominelle $J/\Psi$-Masse. "FTT & Muon" enthält zusätzlich die Forderung einer Verifikation eines Myonkanditaten durch das zentrale Myonsystem. In Verbindung mit Tabelle 4.6 läßt sich erkennen, daß bei geringem Effizienzverlust eine deutliche, zusätzliche Ratenreduktion zu erreichen ist.

- Identifikation von Jets im zentralen Detektorbereich
  Zur Bestimmung der Energie von Jets im zentralen Detektorbereich verwendet die volle H1-Rekonstruktion zum einen Informationen des Flüssigargon-Kalorimeters, für niederenergetische Teilchen aber auch Impulsinformationen aus der Spurkammer. Der geplante Jet-Trigger generiert seine Entscheidung nur auf Grundlage von Kalorimeterinformationen, wobei die Energiedeposition in einem lokalen Bereich zur Rauschunterdrückung allerdings 5 GeV übersteigen muß. Daher wurden Studien zur ergänzenden Identifikation von Jets durch den FTT aufgrund ihrer Spursignatur durchgeführt [See-2002, Jun-2002]. Dem Nachteil einer fehlenden Sensitivi-

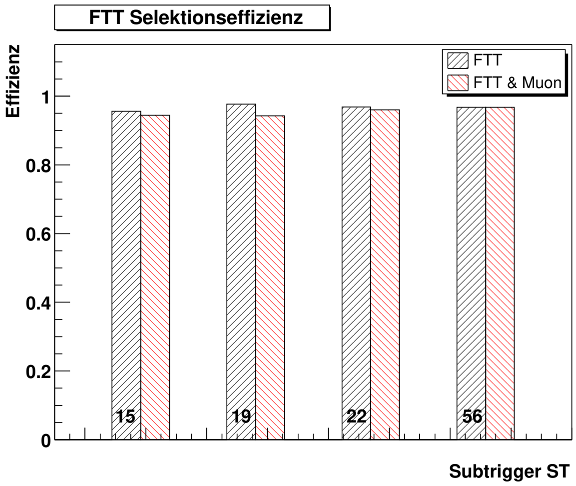



*Abbildung 4.26: Effizienz von $J/\Psi$-Triggern mit zusätzlichen Selektionskriterien auf Basis von Informationen des FTTs und des zentralen Myonsystems in Bezug auf die vollständige H1-Analysekette[Mer-2002].*

| | HERA-I Raten [Hz] | | | HERA-II Raten [Hz] | |
|---|---|---|---|---|---|
| ST | L1 | FTT | FTT & Myon | FTT | FTT & Myon |
| 15 | 1,5 | 0,3 | 0,16 | 1,5 | 0,8 |
| 19 | 0,5 | 0,08 | 0,01 | 0,4 | 0,05 |
| 22 | 0,5 | 0,13 | 0,06 | 0,65 | 0,3 |
| 56 | 1,0 | 0,3 | 0,06 | 1,5 | 0,3 |

*Tabelle 4.6: Triggerraten verschiedener Subtrigger, die auf Informationen des zentralen Myonsystems basieren [Mer-2002]. Die FTT-Raten wurden nach Gleichung 4.19 aus den L1-Raten berechnet. Die HERA-II-Raten ergeben sich aus einer Skalierung mit dem Faktor fünf.*

tät für neutrale Teilchen steht als Vorteil die deutlich höhere Sensitivität für niederenergetische Teilchen gegenüber. Abschließende Ergebnisse für Algorithmen, die diese Suche in der vorgegebenen Zeit effizient durchführen können, liegen noch nicht vor.

#### 4.4.4.2 Neue Elektronikkomponenten

Für beide gewählten, kommerziellen Elektronikomponenten sind inzwischen Modelle mit verbesserter Leistung auf dem Markt erhältlich. FTT-L3 ermöglicht den Einbau beider Komponenten mit geringfügigen Modifikationen am bestehenden System. Zusätzlich müssten allerdings BSPs für VxWorks erworben werden.

- Rechenkarten mit Prozessor der PowerPC7400-Serie
  Mit Karten der MVME5100-Serie [Mot-1999] bietet Motorola inzwischen eine Plattform für PowerPC7410-Prozessoren an. Wie den Abbildung 4.3 und 4.4 zu entnehmen ist, erreichen Prozessoren der PowerPC7400-Serie bei gleicher Rechenleistung bezüglich Ganzzahloperationen eine deutlich bessere Leistung bei Gleitkommaoperationen, die durch konzeptionelle Verbesserungen der dabei verwendeten Recheneinheit erreicht wurde. Zusätzlich konnte die Taktrate auf 500 MHz gesteigert werden. Die resultierende Leistungssteigerung kann mit etwa 25% angegeben werden [Gor-1999]. Eine zusätzliche Leistungssteigerung ist durch die sogenannte AltiVec-Technologie [Ful-1998] möglich, die Operationen mit Daten in 128 Bits breiten Registern durchführt. Dadurch können zum Beispiel vier Gleitkommaadditionen, -Subtraktionen oder -Multiplikationen parallel ausgeführt werden. Damit können häufig auftretende, voneinander unabhängige Berechnungen, wie sie beispielsweise bei der Bildung von Vektorsummen und -Produkten auftreten, durch Parallelisierung beschleunigt werden.

  Eine weitere Leistungssteigerung wird mit Prozessorkarten der von Motorola angekündigten MVME5500-Serie [Mot-2002] möglich sein. Diese wird vor allem durch den Einsatz von PowerPC7455-Prozessoren mit Taktraten von über 1 GHz, aber auch durch eine schnelle Busarchitektur erreicht. So ist zum Beispiel ein 64 Bits breiter, mit 66 MHz getakteter PCI-Bus integriert, der über zwei PMC-Steckplätze verfügt. Durch die Weiterentwicklung im Bezug auf Gleitkommaoperationen und die Fähigkeit zum parallelen Rechnen sind die neueren Prozessorkarten für den Einsatz als L3-Rechenkarten interessant [Hof-2002].

- DPIO2-Karten für FPDP-II
  Um höhere Datenübertragungsraten erreichen zu können, wurde auf Basis des FPDP-Standards die Erweiterung FPDP-II eingeführt. Durch Erhöhung des zu Grunde liegenden Taktes auf 50 MHz und Ausnutzung beider Taktflanken zur Synchronisation lassen sich bei gleichbleibender Datenbreite Transferraten von bis zu 400 MBytes/s erreichen. Die DPIO2-Karte unterstützt FPDP-II, kann aber auch in einem herkömmlichen FPDP-Bus betrieben werden. Um von dem beschleunigten FPDP-II-Datentransfer zu profitieren, ist die Karte mit einer 64 Bits breiten PCI-Schnittstelle ausgestattet, über die die Daten in der 64 MHz-Version mit der vollen FPDP-Bandbreite durch DMA in den Arbeitsspeicher einer Prozessorkarte ge-

schrieben werden können. Effektiv verringert sich damit die Datenübertragungszeit etwa um einen Faktor fünf. Die Programmierung des FPGAs der Empfängerkarte kann durch geringfügige Modifikationen dem FPDP-II-Standard angepaßt werden. Da der Transfer der Spurdaten allerdings weniger als 10% der verfügbaren Rechenzeit in Anspruch nimmt und gegebenenfalls schon andere Berechnungen parallel dazu durchgeführt werden können, bietet das System auch mit den DPIO-Karten eine ausreichende Leistung.

# Kapitel 5

# Zusammenfassung

Nach dem Umbau des HERA-Speicherrings müssen die Experimente H1 und ZEUS in der Lage sein, durch ihre Auslese- und Triggerelektronik eine etwa fünffach höhere Eingangsdatenrate effizient verarbeiten zu können. Ereignisse mit einem hohen Impulsübertrag $Q^2$ zwischen Elektron und Proton sind nach wie vor durch den Nachweis eines Elektrons im Zentralbereich des Detektors bei relativ kleinen Raten leicht zu selektieren. Für kleine $Q^2$ würde die Aufzeichnung aller detektierten Ereignisse die Bandbreite des Auslesesystems übersteigen. Interessante Ereignisse in diesem kinematischen Bereich können jedoch zusätzlich durch charakteristische Topologien ihrer Teilchenspuren identifiziert werden. Für Teilchenzerfälle läßt sich aus den Spurparametern beispielsweise die invariante Masse bestimmen.

Um die Selektionseffizienz für spurbasierte Ereignistopologien zu verbessern, baut die H1-Kollaboration einen schnellen Spurtrigger, FTT, der die Datenrate innerhalb der ersten drei Triggerstufen reduzieren soll. Innerhalb von 2,3 $\mu$s werden auf Basis von Spurkammerinformationen in vier radialen Bereichen Segmente von auf den nominellen Ereignisvertex zeigenden Spuren bis zu einem minimalen Transversalimpuls von 100 MeV identifiziert. Der zweiten Stufe stehen 21 $\mu$s zu Verfügung, um daraus vollständige Spuren zu bilden und eine Spurparameteranpassung durchzuführen. Dies ist der Ausgangspunkt, um durch die dritte Stufe Teilchenzerfällen an Hand der invarianten Masse der Zerfallsprodukte zu identifizieren. Dafür stehen etwa 100 $\mu$s zu Verfügung. Zusätzliche können die Triggerdaten anderer Subdetektorsysteme zur Entscheidungsfindung herangezogen werden.

In der vorliegenden Arbeit wurde ein Konzept für Aufbau und Programmierung der dritten Stufe des FTT entwickelt. Die Umsetzung war nur durch den Einsatz der schnellsten, verfügbaren Elektronikkomponenten möglich. Die Eingangsdaten werden durch eine an der Universität Dortmund entwickelte Elektronikkarte, deren Logik in einem FPGA implementiert ist, empfangen und über einen FPDP-Bus parallel an bis zu acht angeschlossene Prozessorkarten weitergeleitet. Kernstück dieser für den Datenempfang mit einer DPIO-Aufsteckkarte bestückten Rechenkarten ist ein PowerPC-Prozessor, der unter dem Echtzeitbetriebssystem VxWorks betrieben wird. Jede Karte führt Selektionsalgorithmen für eine zugeordnete Ereignistopologie aus. Über eine weitere, noch zu bauende Karte wird die endgültige Entscheidung an die zentrale Triggerlogik gesendet. Konzeptionell ist der Betrieb von maximal 16 Rechenkarten vorgesehen. Eine weitere Prozessorkarte kontrolliert das L3-System, indem sie die einzelnen Karten initialisiert, Kontrolldaten aus den Rechenkarten ausliest und periodisch die Datentransferkette durch einen Systemtest überprüft.

Durch diesen Systemtest konnte an einem Testaufbau die fehlerfreie Funktion der gesamten Datenübertragungskette in FTT-L3 verifiziert werden. Desweiteren wurde die

Laufzeit von Selektionsalgorithmen untersucht. Bei der Selektion reiner Zweiteilchenzerfälle wird die Zeitvorgabe von 100 $\mu$s deutlich unterschritten, bei der aufwendigeren Rekonstruktion zweistufiger Zerfälle wird dieser Zeitrahmen von mehr als 99% der zu selektierenden Ereigniss eingehalten. Mit einfach zu integrierenden Prozessorkarten der neuesten Generation wird die Bearbeitung der Algorithmen noch deutlich beschleunigt werden können.

Weiterhin konnte im Rahmen dieser Arbeit ein System zur Datennahme für den gesamten FTT entwickelt werden. In einem mehrstufigen Prozeß werden dabei die Daten durch Prozessorkarten aus den Elektronikkarten der einzelnen Stufen ausgelesen, gebündelt und an das H1-Datennahmesystem weitergeleitet. Außerdem wird durch dieses System der FTT an die zentrale Triggerlogik des H1-Experiments angebunden.

# Anhang A

# Datennahmesystem des Fast Track Triggers

Zur Kontrolle der Funktionsweise des FTT ist es notwendig, während des laufenden Betriebs Daten aus den einzelnen Komponenten auszulesen. Hier sind zwei Systeme zu unterscheiden: Zum einen das Auslesesystem, das Daten dem H1-Datenstrom hinzufügt, um diese dann dauerhaft zu speichern und jederzeit analysieren zu können, zum anderen ein Kontrollsystem, welches Daten zu einem Kontroll-PC zur sofortigen graphischen Anzeige schickt. Ersteres ist im Rahmen diese Arbeit entstanden und soll im Folgenden ausführlicher vorgestellt werden.

Der durch das H1-Datennahmesystem vorgegebene Zeitrahmen wurde bereits in Abschnitt 2.2 vorgestellt. An dieser Stelle soll nun die Funktionalität der Komponenten beschrieben werden, die als Schnittstellen bei der technischen Umsetzung dienen. Außerdem werden die für das FTT-Datenauslesesystem FTT-R/O[1] spezifischen Komponenten eingeführt, um abschließend beschreiben zu können, wie sich der FTT-R/O in das Gesamtsystem einfügt.

## A.1 Elektronische Komponenten

Um eine Detektor- oder Triggerkomponente in das H1-Datennahmesystem integrieren zu können, muß sie logisch mit der zentralen Triggerlogik CTL und dem zentralen Datenauslesesystem CDAQ[2] verbunden sein. Zur Synchronisation mit ersterem dienen verschiedene STC-Karten, die Verbindung zu letzterem wird durch die sogenannte TAXI-Karte hergestellt. Teilweise werden Informationen redundant durch beide Kanäle übertragen. Der FTT-R/O stellt durch Prozessorkarten eine programmierbare Logik bereit, die auf Signale von außen reagiert, die Datenauslese innerhalb der FTT steuert und schließlich die Daten in der TAXI-Karte bereitstellt. Um Daten und Signale zwischen den verschiedenen FTT-Subsystemen übertragen zu können, kommen VIC[3]-Karten zum Einsatz.

### A.1.1 STC Fast-Karte

Die von der CTL kommenden Signale werden bei H1 als "einlaufende Signale" bezeichnet und in zwei Klassen unterteilt. "Fast"-Signale bezeichnet die zur Synchronisation eines Ereignisses dienenden Signale bis einschließlich **L2Keep**. Sie werden als differentielles Signal übertragen und aufgrund ihrer schnellen Abfolge hauptsächlich

[1] Read-**O**ut
[2] **C**entral **D**ata **Acq**uisition
[3] Vertical **I**nter**c**onnection

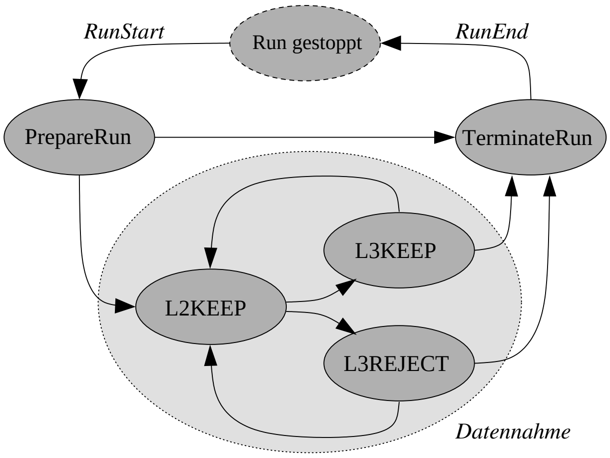



*Abbildung A.1:* Abfolge der durch die Slow-Karte übertragenen Signale.

von festverdrahteten Elektronikschaltungen ausgewertet. Da die Datenauslese erst mit dem Empfang des **L2Keep**-Signals beginnt, soll nur dieses weiter betrachtet werden. Als Schnittstelle für "Fast"-Signale dient in jedem H1-Subsystem eine sogenannte Fast-Karte [Kre-1989], die als A24/D08-VME-Karte ausgeführt ist. Über den dadurch nicht benötigten unteren Stecker werden Signale zwischen der Fast-Karte, CTL und weiteren STC-Karten ausgetauscht. So wird zum Beispiel das **L2Keep**-Signal zu der in Abschnitt A.1.2 beschriebenen Slow-Karte weitergeleitet.

Nach einem Run-Start oder Beendigung der Auslese wird der CTL durch einen Schreibbefehl in ein spezielles Register der Fast-Karte durch das **FrontEndReady**(FER)-Signal mitgeteilt, daß das System zur Bearbeitung des nächsten Ereignisses bereit ist. Weiterhin sind Register implementiert, in denen Zähler in Abhängigkeit der Teilchenpakete in HERA nach verschiedenen Kriterien inkrementiert werden. Diese sind für jedes Ereignis auszulesen und dem Datenstrom hinzuzufügen, um an Hand dieser Daten den Synchronismus aller H1-Subsysteme kontrollieren zu können.

### A.1.2 STC Slow-Karte

Die zweite Klasse von Signalen umfaßt die sogenannten "Slow"-Signale. Hierzu gehören alle Signale nach einschließlich **L2Keep**, sowie **PrepareRun** und **TerminateRun**, die Beginn und Ende eines Runs markieren. **L2Keep** ist somit als "Fast-" und "Slow Signal" zu betrachten. Als Schnittstelle dient in diesem Fall die als A24/D16-VME-Karte ausgeführte Slow-Karte [Ols-1990], die die Signale teilweise direkt von der CTL, teilweise von der Fast-Karte empfängt und als konfigurierbare VME-Interrupts ausgibt. Die mögliche Abfolge der Signale ist in Abbildung A.1 gezeigt. Jedes Signal ist einzeln abschaltbar, Interrupt-Vektor und Level (zwischen 4 und 6) sind frei konfigurierbar. Der Empfang eines Interrupts muß durch die diesen auswertende Logik durch entsprechende Schreibzugriffe auf die Slow-Karte quittiert werden. Auch in der Slow-Karte sind Zähler implementiert, die durch VME-Zugriffe ausgelesen werden können und die Zahl von **L1Keep**- und **L2Keep**-Entscheidungen im laufenden Run liefern.

### A.1.3 STC Fanout-Karte

Der Vollständigkeit halber soll die Fanout-Karte [Kre-1991] erwähnt werden, die Signale von der Fast- und Slow-Karte abgreift und an mehrere Empfänger weiterleiten kann. Von dort empfangene Signale können wiederum in verschiedenen Konfigurationen verknüpft und an die CTL gesendet werden. Da zum Betrieb der FEM- und MPB-Karten des FTT einige STC-Signale notwendig sind, werden diese über Fanout-Karten und den von der Universität Manchester entwickelten Service-Modulen zu der L1- und L2-Elektronik des FTT gesendet.

### A.1.4 TAXI-Karte

Um die verschiedenen H1-Subsysteme mit der CDAQ zu verbinden, wird in der Elektronik des H1-Experiments die sogenannte “VMExi-” oder TAXI-Karte eingesetzt [Pie-1991]. Dabei handelt es sich um eine A32/D32-VME-Karte mit einem 68k-Prozessor zur Steuerung und TAXI-Bausteinen zur Datenübertragung über optische Fasern. Wie diese Karte für den Einsatz im H1-Experiment konfiguriert ist und zahlreiche Karten in der H1-CDAQ zusammenarbeiten, ist in [Hay-1993] beschrieben.

Für den Anwender sind hauptsächlich drei Funktionalitäten der lokalen Karte von Interesse:

- Ein Bitmuster in einem bestimmten Register zeigt an, welche H1-Subsysteme an der zentralen Datennahme teilnehmen. Die CTL verlangt nur für diese ein Quittieren der STC-Signale.

- Durch Auslesen des sogenannten *SYSMODE*-Registers ist periodisch der aktuelle Status der CDAQ abzufragen und entsprechend darauf zu reagieren. Abbildung A.2 zeigt die möglichen Abfolgen der Status.

  - **Run gestoppt**
    In diesem Zustand findet keine zentrale Datenauslese statt. Es muß einzig in kurzen Zeitintervallen das *SYSMODE*-Register auf eine mögliche Änderung hin kontrolliert werden.
  - **PrepareRun**
    Wird im Kontrollraum ein Runstart ausgelöst, wird dies zunächst durch eine Änderung im *SYSMODE*-Register angezeigt. Ausgehend davon sendet die CTL einen **PrepareRun**-Interrupt als redundante Information durch die Slow-Karte. Jedes ausgewählte H1-Subsystem muß nun bereit sein, entsprechend der STC-Signale an der Datennahme teilzunehmen.
  - **Warten auf Runstart Record**
    Ist ein bestimmtes Bit im *SYSMODE*-Register gesetzt, erwartet die CDAQ vor der eigentlichen Datennahme das Senden einer sogenannten *Runstart Record*. Darin können Informationen über die aktuelle Systemkonfiguration enthalten sein, die damit dauerhaft gespeichert und jederzeit den nachfolgenden Daten zugeordnet werden können.

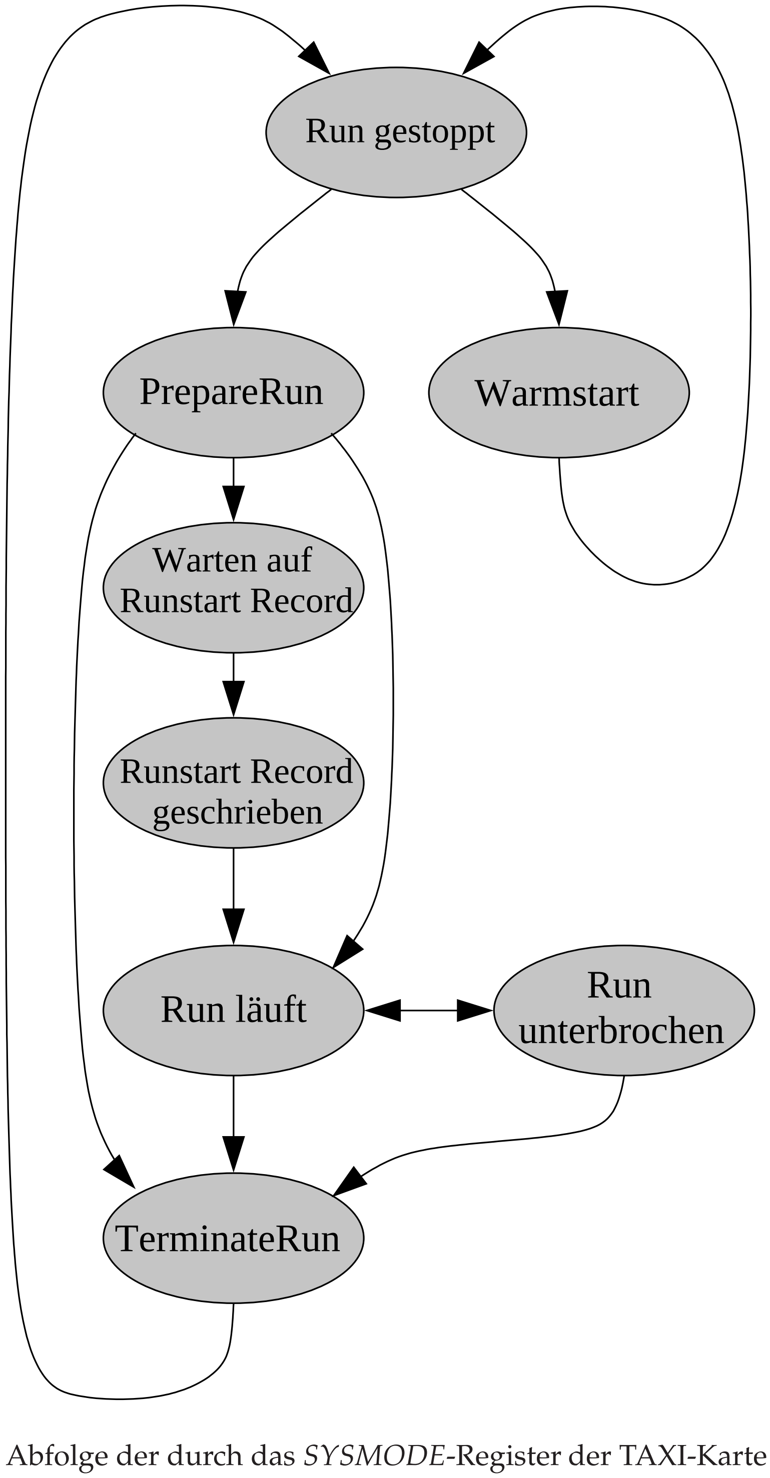



*Abbildung A.2:* Abfolge der durch das *SYSMODE*-Register der TAXI-Karte übertragenen Runstatus.

- **Runstart Record geschrieben**
  Erst wenn jedes an der Datenauslese beteiligte H1-Subsystem einen möglicherweise auch leeren *Runstart Record* gesendet hat, kann ein Run endgültig gestartet werden.
- **Run läuft**
  In diesem Status können wie unten beschrieben Ereignisdaten durch die TAXI-Karte an die CDAQ gesendet werden. Für die Synchronisation der H1-Subsysteme ist nun die CTL verantwortlich.
- **Run unterbrochen**
  Ein Run kann jederzeit unterbrochen und nachfolgend reaktiviert werden, ohne daß die gesamte Kette von Stopp- und Startprozeduren durchlaufen werden muß. Diese Möglichkeit wird allerdings in der Realität selten genutzt.
- **Terminate Run**
  Bei Änderungen der experimentellen Bedingungen wird die Datennahme angehalten. Dies wird wiederum durch eine Bedienhandlung im Kontrollraum ausgelöst, durch den Inhalt des *SYSMODE*-Registers angezeigt und zusätzlich durch einen **TerminateRun**-Interrupt der Slow-Karte übermittelt. Jedes H1-Subsystem muß daraufhin seine Datennahme mit einer speziellen *Runend Record* beschließen.
- **Warmstart**
  Eine besondere Form eines Runs ist der sogenannte Warmstart. Dabei werden keine Daten ausgelesen, sondern Komponenten der H1-Subsysteme neu konfiguriert. Ein Warmstart ist häufig notwendig, um ein Subsystem in die Datenauslese zu reintegrieren. Haben alle an der Auslese beteiligten Subsysteme den Warmstart beendet, kann ein neuer Run gestartet werden.

• Für die Datenübertragung selbst werden sogenannte "Multi Eventbuffer" MEB bereitgestellt, die im Speicher der TAXI-Karte durch den darauf befindlichen Prozessor verwaltet werden. Während eines Runstarts ist dem Prozessor Zahl und Größe der benötigten MEBs mitzuteilen, um anschließend für jedes Ereignis einen MEB anfordern zu können. Durch den Prozessor wird daraufhin die Adresse, laufende Nummer und tatsächlich mögliche Größe des MEBs ermittelt und der Ausleselogik des Subsystems zu Verfügung gestellt. Sind alle Auslesedaten als BOS-Strukturen in den MEB geschrieben worden, wird dies der TAXI-Karte unter Angabe der Adresse, MEB-Nummer, Ereignisnummer und wirklichem Datenvolumen mitgeteilt. Dieser Schritt findet typischerweise erst nach der synchronen Phase der Datenauslese für ein Ereignis statt. Die MEBs dienen auch der Übertragung von *Runstart-* und *Runend-Records*. Der Prozessor der TAXI-Karte sorgt abschließend dafür, daß die Inhalte eines vollständigen MEBs zur CDAQ übertragen werden, um sie dort mit den MEBs der anderen Subsysteme für jedes Ereignis zu vereinen.

### A.1.5 Prozessorkarten

Im FTT-R/O kommt mit den MVME2400-0321-Karten von Motorola der gleiche Typ wie bei FTT-L3 zum Einsatz. Aufbau und Eigenschaften der Karten wurden bereits ausführlich in Abschnitt 4.2.2.1 beschrieben. Für deren Betrieb wird ebenfalls das Echtzeitbetriebssystem VxWorks verwendet, das in Abschnitt 4.3.1.1 vorgestellt wurde.

### A.1.6 VIC-Karten

Für die interne Kommunikation sind die einzelnen FTT-Subsysteme durch den sogenannten VMV-Bus über Flachbandkabel miteinander verbunden. Als Schnittstelle zu den VME-Bussen dienen dabei jeweils VIC-8250-Karten [CES-1992], von denen bis zu 15 in einem VMV-Bus betrieben werden können. Sie erlauben es, Daten und Interrupts zwischen den FTT-Subsystemen auszutauschen. Darüberhinaus bietet jede Karte 512 kBytes Speicherplatz, der über beide angeschlossenen Busse angesprochen werden kann. Durch einen Wahlschalter muß der Karte eine eindeutige Nummer im VMV-Bus zugeordnet werden.

Zur Datenübertragung müssen in der lokalen VIC-Karte sogenannte Pages angelegt werden, durch die VME-Zugriffe an bestimmte Adreßbereiche zu einem anderen VME-Bus weitergeleitet werden. Dazu muß im lokalen VME-Bus ein freier Adreßbereich ausgewählt werden und die zugehörige Page mit der Nummer der empfangenden VIC-Karte und der VME-Zieladresse konfiguriert werden, die im entfernten Bus angesprochen werden soll. Die Zugriffe können dabei im A24- oder A32-Adreßraum stattfinden.

Eine Sonderform sind "Broadcall"-Zyklen, bei denen jeder VIC-Karte eines VMV-Busses ein spezielles Bit im Datenwort zugeordnet ist. Dies erlaubt ein paralleles antworten aller Karten. Auf diese Weise läßt sich über drei spezielle Register in den VIC-Karten abfragen, ob

- die Karte eingeschaltet ist.
- auf der Karte ein Fehler aufgetreten ist.
- durch das Register *Mailbox 0* der Karte ein Interrupt aktiviert wurde (siehe unten).

Damit steht ein Instrumentarium zur schnellen Überprüfung des Zustands des Systems zu Verfügung.

Zusätzlich zur Datenübertragung stellt der VMV-Bus eine eigene Interruptstruktur mit 14 möglichen Interrupts bereit. Jede VIC-Karte kann als Sender eines oder zweier Interrupts konfiguriert werden, die von allen anderen Karten im VMV-Bus ausgewertet werden können. Beide einer VIC-Karte zugeordneten Interrupts werden durch verschiedene Aktionen aktiviert.

- *Mailbox 0*-Interrupt
  Durch einen Schreibzugriff auf das sogenannten *Mailbox 0*-Register wird ein VMV-Interruptzyklus initiiert, der in anderen angeschlossenen Systemen über deren jeweilige VIC-Karte einen VME-Interrupt auslöst, dessen Priorität und Interruptvektor konfigurierbar sind. Eine dort in einem Prozessor aktivierte ISR muß zunächst einen Schreibzugriff auf das *Mailbox 0*-Register in der sendenden VIC-Karte ausführen, um anschließend den Interrupt in der lokalen VIC-Karte löschen zu können.
- *VME-VMV*-Interrupt
  Die VIC-Karten können konfiguriert werden, einen VME-Interrupt einer bestimmten Priorität zu empfangen und daraufhin einen VMV-Interruptzyklus auszulösen. In den angeschlossenen Systemen kann dadurch ein VME-Interrupt mit der

gleichen Priorität wie ein durch einen *Mailbox 0*-Zugriff ausgelöster Interrupt, jedoch mit verschiedenem Interruptvektor aktiviert werden. Die Empfänger müssen durch spezielle VMV-Schreibzugriffe die auslösende VIC-Karte veranlassen, den VME-Interrupt-Zyklus zu beenden, um daraufhin den Interrupt auf der lokalen VIC-Karte löschen zu können.

Ergänzend zu den beschriebenen Interrupts, die von mehreren Karten parallel empfangen werden können, bietet jede VIC-Karte durch die Register *Mailbox 1-3* die Möglichkeit, lokale VME-Interrupts durch externe Schreibzugriffe auszulösen. Diese haben die gleiche Priorität wie die weiter oben beschriebenen, senden jedoch wiederum einen anderen Interruptvektor. Zur weiteren Unterscheidung können unterschiedliche Datenworte in die Register geschrieben und durch die aktivierte ISR ausgelesen und ausgewertet werden. Diese Form des Interrupts ist durch Registerzugriffe auf die lokale VIC-Karte zu löschen.

## A.2 Programmierung

Abbildung A.3 zeigt einen Überblick über die Elektronikkomponenten des FTT und die sie verbindenden Datenbusse. Für die Datenauslese sind dabei in jedem FTT-Subsystem eine Prozessorkarte (CPU-Karte) und eine VIC-Karte vorgesehen, die alle durch einen gemeinsamen VMV-Bus verbunden sind. In den L1- und L2-Systemen befindet sich darüberhinaus je ein Servicemodul als Schnittstelle für STC-Signale. Die zentrale Steuerung der Datenauslese geschieht durch Karten im sogenannten STC-Crate[4], in dem als programmierbare Logik ebenfalls eine CPU-Karte integriert ist. Ihr obliegt die Verwaltung der lokalen TAXI- und STC-Karten und des Daten- und Signaltransfers zu und von den FTT-Subsystemen über die VIC-Karte. Alternativ können einige Triggersignale durch die STC-Fanoutkarte zu den Service-Modulen oder der L3-Triggerbit-Karte gesendet werden.

Bezüglich der Programmierung wird hauptsächlich zwischen der als R/O-Mutterkarte[5] dienenden CPU-Karte in dem STC-Crate und den als R/O-Tochterkarten[6] bezeichneten CPU-Karten in den Subsystemen unterschieden. Die resultierende, hierarchische Struktur von Programmbibliotheken ist in Abbildung A.4 gezeigt. Auf der untersten Stufe ist dabei die Elektronik der Systeme dargestellt, die teilweise durch spezielle, nicht für den FTT spezifische Programmbibliotheken angesprochen wird. Darauf aufbauend wurden Schnittstellen für bestimmte Abläufe in der Datenauslese des FTTs entwickelt. Auch für die Elektronik jedes Subsystems gibt es dabei eine genau definierte Schnittstelle, an der die Daten aus den Elektronikkarten ausgelesen und an die Ausleseroutinen einer R/O-Tochterkarte übergeben werden. Schließlich wurden Bibliotheken erstellt, die die Rahmenprogramme für die Datenauslese bereitstellen. Die Abbildung soll allerdings nur den strukturellen Aufbau des Programmbaums veranschaulichen, Bibliotheken höherer Ebenen rufen teilweise auch Routinen in weiter unten liegenden Bibliotheken direkt auf.

[4] Mit dem englischen Begriff Crate wird ein die Rückwandplatine aufnehmendes Elektronikgehäuse bezeichnet.

[5] In der Software als R/O-Master bezeichnet

[6] In der Software als R/O-Slave bezeichnet

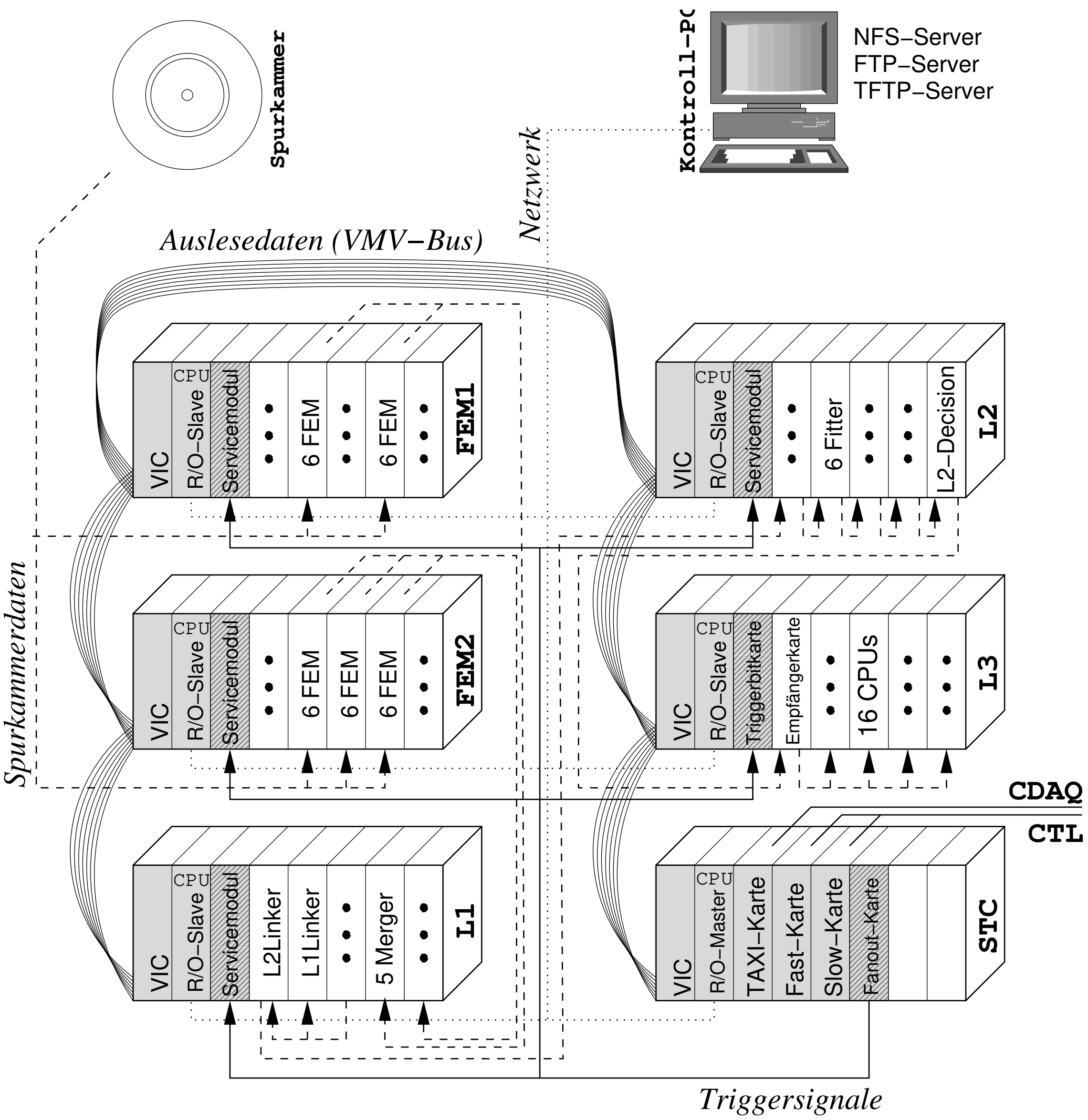



*Abbildung A.3:* Komponenten des FTTs. Für die Auslese jedes Subsystems ist eine Prozessorkarte (CPU) vorgesehen, die über eine VIC-Karte und den VMV-Bus mit dem die Auslese steuernden Prozessor im STC-Crate verbunden ist. Über die TAXI-Karte wird dort die Verbindung zur CDAQ, über Fast- und Slow-Karte zur CTL hergestellt.

Für den Datentransfer wurde ein mehrstufiges Konzept entwickelt, bei dem in der synchronen Phase die Daten zunächst nur innerhalb der einzelnen Subsystem zusammengetragen werden, um möglichst schnell ein **FER** an die CTL senden zu können. Der endgültige Transfer über die R/O-Mutterkarte zu der TAXI-Karte findet erst statt, wenn kein Ereignis in der synchronen Phase bearbeitet werden muß. Daher ist zwischen jedem einzelnen Schritt die Möglichkeit einer temporären Datenspeicherung vorzusehen.

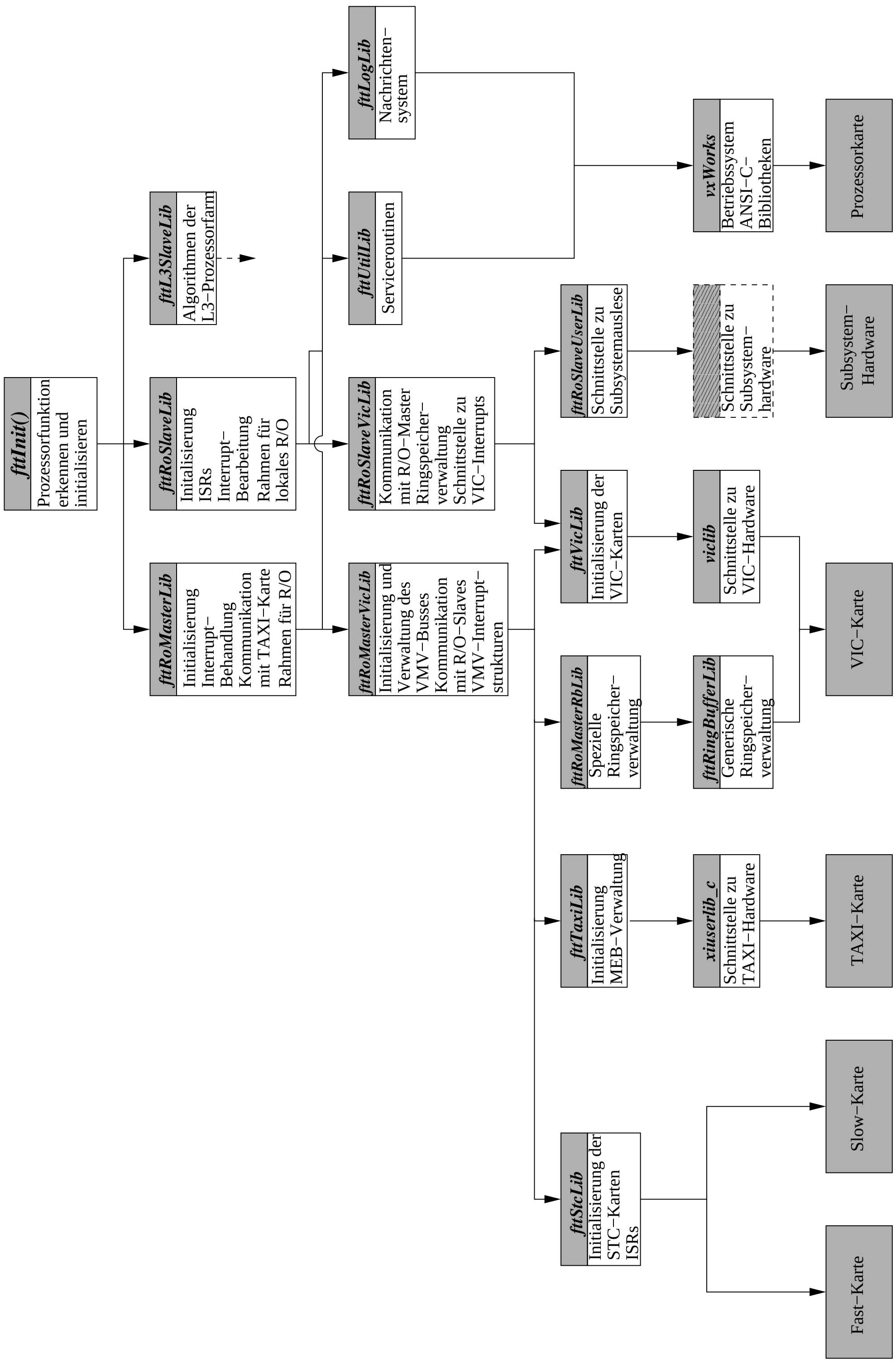



*Abbildung A.4:* Hierarchische Struktur der bei der Datenauslese des FTTs verwendeten Softwarebibliotheken.

Im folgenden sollen Programmabläufe bei der Systemkontrolle und Datenübertragung beschrieben werden.

### A.2.1 Kontrollstrukturen

In Abschnitt 4.3.1.1 wurde beschrieben, daß mit dem Betriebssystem weitere Anwendungen beim Einschalten einer CPU-Karte gestartet werden können. Wie in Abbildung A.4 angedeutet, wird auf jeder CPU-Karte zunächst die Routine *fttInit()* ausgeführt. Diese wertet wie in Abschnitt 4.3.2 beschrieben zunächst die Prozessornummer der Karte aus. Ist diese ungleich Null, wird die Karte der L3-Rechnerfarm zugeordnet. Anderenfalls kann aus einem Register der VIC-Karte deren Nummer ausgelesen und das Subsystem dadurch identifiziert werden. Die VIC-Karte im STC-Crate ist mit der Nummer 1 konfiguriert und löst auf der CPU-Karte den Aufruf der Routine *fttRoMasterInit()* aus. Liegt die gelesene Nummer zwischen 2 und 6 wird dementsprechend *fttRoSlaveInit()* gestartet, alle anderen Fälle werden unter Erzeugung einer Fehlermeldung abgefangen. Auf jeder Karte werden anschließend notwendige Tasks, ISRs und Datenstrukturen initialisiert. Durch die R/O-Mutterkarte werden außerdem die Karten im STC-Crate konfiguriert.

Nach der Initialisierungsphase prüft die R/O-Mutterkarte etwa sekündlich durch einen VMV-Broadcall-Zyklus die aktuelle Konfiguration des VMV-Busses. Eine neu eingeschaltete VIC-Karte wird sofort, oder nach dem Ende eines eventuell stattfindenden Runs, initialisiert. Durch die R/O-Mutterkarte werden dabei folgende Pages in der lokalen und entfernten VIC-Karte aktiviert:

- Eine A24-Page von der lokalen VIC-Karte in den Adreßraum der Register der entfernten VIC-Karte. Durch diese Page wird die entfernte VIC-Karte nach dem Einschalten konfiguriert.
- Eine A32-Page von der lokalen VIC-Karte in den Adreßraum des Speichers der entfernten VIC-Karte. Durch diese Page werden die Auslesedaten übertragen.
- Eine A24-Page von der entfernten VIC-Karte in den Adreßraum der Register der lokalen VIC-Karte. Diese Page wird benötigt, um von einem Subsystem auf das *Mailbox 0*-Register zum Löschen des Interrupts zugreifen zu können.

Kann zusätzlich durch die weiter unten beschriebene Funktionalität Kontakt zu der CPU-Karte aufgebaut werden, wird das entsprechende Subsystem in den FTT-R/O integriert. Die endgültige Integration wird später zusätzlich durch Konfigurationsbefehle des Kontrollrechners gesteuert werden können. Ein Subsystem kann demnach aus Sicht der R/O-Mutterkarte folgende Zustände annehmen:

- **Offline**: Die Elektronik des Subsystems ist nicht eingeschaltet.
- **Online**: Die Elektronik wurde eingeschaltet, die VIC-Karte aber noch nicht initialisiert. Diesen Zustand nimmt typischerweise ein während eines Runs eingeschaltetes Subsystem ein.
- **Initialisiert**: Die VIC-Karte im Subsystem wurde initialisiert, es konnte aber noch keine Kommunikation mit der dortigen CPU-Karte aufgenommen werden. Dieser Zustand tritt entweder während der Startphase der CPU-Karte nach dem Einschalten des Subsystems oder nach einem Fehler in der CPU-Karte auf.

- **Integriert**: Das Subsystem ist vollständig in den FTT-R/O integriert und muß während eines Runs durch die R/O-Mutterkarte ausgelesen werden können.

- **Signalbedürftig**: Dies ist ein Zustand, den ein **integriert**es Subsystem zusätzlich annehmen kann. Er besagt, daß die STC-Signale nicht durch eine gesonderte Karte in das System übertragen werden, sondern über den VMV-Bus zu senden sind.

- **Auszulesen**: Dieser Zustand wird später ein System bezeichnen, das **integriert** ist und zusätzlich durch den Kontrollrechner zur Datennahme ausgewählt wurde.

Antwortet eine aktuell in das Auslesesystem **integriert**e VIC-Karte nicht auf einen Broadcall-Zyklus, wird sie von einer eventuell stattfindenden Datennahme unter Generierung einer Fehlermeldung ausgeschlossen. Dieses Subsystem kann frühestens nach Beendigung des laufenden Runs wieder **integriert** werden. Die beschriebene, automatische Erkennung der Konfiguration des VMV-Busses durch die R/O-Mutterkarte erlaubt es, die einzelnen Subsysteme und das STC-Crate in beliebiger Reihenfolge ein- oder auszuschalten.

Eine zusätzliche Kontrolle der Verfügbarkeit eines Subsystems sei an dieser Stelle ebenfalls erwähnt. In der VIC-Karte jedes Subsystems ist eine Speicherstelle reserviert, auf die R/O-Mutter- und Tochterkarte alternierend zugreifen. Dabei wird zunächst der Inhalt ausgewertet. Entspricht dieser einem Kennwort der jeweiligen Gegenseite, wird der Speicher mit dem eigenen Kennwort überschrieben. Bleibt für mehr als zwei Lesezugriffe ein Überschreiben durch die Gegenseite aus, wird angenommen, daß die dortige CPU-Karte nicht mehr einsatzbereit ist. Ein Subsystem unterbricht in diesem Fall eigenständig eine eventuelle Datennahme und kontrolliert weiterhin regelmäßig die Speicherstelle, um eine Reaktivierung der Verbindung detektieren zu können. Bleibt die Antwort einer Tochterkarte aus, wird das entsprechende Subsystem als **offline** markiert und von der aktuellen Datennahme ausgeschlossen. Dies wird wiederum durch einen Warnhinweis auf dem Kontrollrechner angezeigt werden.

Eine weitere Initialisierung findet mit jedem Runstart statt. Dabei werden die MEBs in der TAXI-Karte eingerichtet und der R/O-Mutterkarte deren maximale Größe mitgeteilt. Desweiteren wird jedem Subsystem durch Beschreiben einer bestimmten Speicherstelle in der VIC-Karte mitgeteilt, wieviel Speicherplatz davon für dieses reserviert ist. Durch *Mailbox 1*-Interrupts wird den einzelnen Subsystemen schließlich das **PrepareRun**-Signal übermittelt. Der Vollständigkeit halber sei hier angemerkt, daß ein **TerminateRun**-Signal auf gleichem Wege, aber mit einem anderen Eintrag in den *Mailbox 1*-Registern übertragen wird.

### A.2.2 Datentransfer

Konzeptionell orientiert sich der Datentransfer an dem Aufbau der Bussysteme einer Prozessorkarte. Busse verschiedener Geschwindigkeit sind dabei voneinander entkoppelt, um einen möglichst hohen Datendurchsatz erreichen zu können. Abbildung A.5 zeigt, wie dies für der FTT-R/O realisiert ist.

Nach Empfang eines **L2Keep**-Interrupts schreibt die R/O-Mutterkarte in den Speicher der VIC-Karte jedes Subsystems einen Zeiger auf einen Speicherbereich $\boxed{2}$ in dieser Karte, an den später die Daten abzulegen sind, sowie die aktuelle Ereignisnummer, die

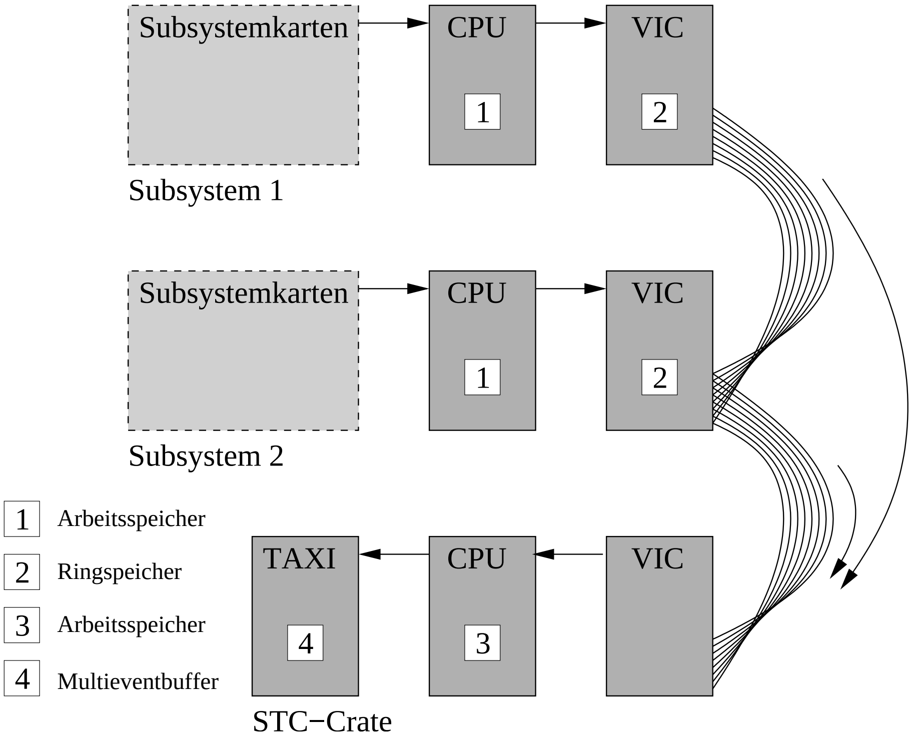



*Abbildung A.5:* Datenfluß und Datenzwischenspeicherung des Auslesesystems des FTTs. Während der synchronen Phase werden alle Daten eines Subsystems zunächst im Arbeitsspeicher der lokalen Prozessorkarten "CPU" gesammelt. Während der asynchronen Phase gelangen die Daten dann über zwei Zwischenschritte in den MEB der TAXI-Karte.

mit jedem **L2Keep**-Interrupt inkrementiert und mit dem Registerinhalt der Slow-Karte abgeglichen wird. Anschließend initiiert die R/O-Mutterkarte einen *Mailbox 0*-Interrupt[7]. Daraufhin beginnt die CPU-Karte jedes Subsystems mit der Datenübertragung aus den Elektronikkarten über den VME-Bus in den lokalen Arbeitsspeicher $\boxed{1}$. Als vereinheitlichte Schnittstelle dient hier die Routine *fttRoSlaveUserTask()*, die den Zeiger auf den Speicherbereich $\boxed{1}$ und die maximale Datengröße übergibt. Bei Terminierung der Routine nimmt letztere Variable den Wert der wahren Datengröße an. Die Daten sollten an dieser Stelle schon in einer BOS-Struktur vorliegen. Ist dieser Transfer abgeschlossen und ein **L3Keep**-Interrupt durch einen *Mailbox 1*-Interrupt übertragen worden, werden der Zeiger auf $\boxed{1}$ und $\boxed{2}$ zusammen mit der Angabe des gelesenen Datenvolumens und der aktuellen Ereignisnummer in eine Message-Queue geschrieben und der belegte Speicher für weitere Daten blockiert. Die R/O-Mutterkarte reserviert nun ebenfalls ausreichend Platz in ihrem Arbeitsspeicher $\boxed{3}$, um die Daten aller Subsysteme aufnehmen zu können, und fordert von der TAXI-Karte einen MEB $\boxed{4}$ an. Der Zeiger darauf wird zusammen mit den Zeigern auf die Speicherbereiche $\boxed{2}$ und $\boxed{3}$ und der aktuellen Ereignisnummer ebenfalls in eine Messagequeue geschrieben. Zusätzlich werden Register der STC-Karten

[7] Im folgenden wird der ausschließliche Signaltransfer über den VMV-Bus angenommen.

ausgelesen und deren Inhalte in eine Datenstruktur gefüllt, die ebenfalls der Messagequeue übergeben wird. Ein alternativer **L3Reject**-Interrupt wird als *VME-VMV*-Interrupt weitergeleitet und bewirkt den sofortigen Abbruch der laufenden Auslese und die Freigabe der Speicherbereiche $\boxed{1}$ und $\boxed{2}$ für Daten des folgenden Ereignisses. Der bisher beschriebene Datentransfer sollte im Falle einer **L3Keep**-Entscheidung nicht länger als eine Mikrosekunde dauern. Das Ende wird der R/O-Mutterkarte von den R/O-Tochterkarten durch Überschreiben der Ereignisnummer in der VIC-Karte mit Nullen mitgeteilt. Hat die R/O-Mutterkarte aus allen VIC-Karten dieses Signal ausgelesen, sendet sie das **FER**-Signal an die CTL und beendet somit die synchrone Phase der Datenauslese.

Der nachfolgend beschriebene Teil der Datenauslese findet asynchron zu anderen H1-Systemen statt und muß jederzeit durch einen **L2Keep**-Interrupt unterbrochen werden können. Um kurzfristig hohe **L2Keep**-Raten handhaben zu können, ist sicherzustellen, daß für jeden Schritt der Datenübertragung ausreichend paralleler Speicherplatz für die Daten mehrere Ereignisse bereitgehalten wird. Limitierender Faktor ist hier die VIC-Karte mit den oben genannten 512 kBytes. Dies muß allerdings in Relation zu einem maximalen Datenvolumen von 4 kBytes pro Subsystem während der normalen Datennahme gesehen werden. Lediglich für Testzwecke wird sich dies erhöhen.

Befindet sich ein Eintrag in der Messagequeue einer R/O-Tochterkarte, geht ein Task für die asynchrone Auslesephase in den Zustand **ready** über. Dieser transferiert die Daten in einem DMA-Block-Transfer aus dem lokalen Arbeitsspeicher an die angegebene Adresse in der VIC-Karte und aktualisiert eine unten beschrieben Datenstruktur am Anfang dieses Speichers. Speicher $\boxed{1}$ kann daraufhin von dem Betriebssystem für die weitere Nutzung wieder freigegeben werden.

Der Speicher der VIC-Karten wird zentral durch die R/O-Mutterkarte verwaltet. Dafür wurden aus der CTL-Programmbibliothek generische Routinen übernommen [Sch-2001], die diesen als Ringspeicher organisieren, in dem reihum Speicherblöcke für die Daten bereitgestellt werden. Die Speicherblöcke werden in der Reihenfolge des Beschreibens wieder ausgelesen. Ist kein Speicherblock im Ring mehr frei, muß die Datennahme unterbrochen werden. Der Aufbau eines Speicherblocks ist in Abbildung A.6 gezeigt. Nachdem Daten in den Speicherblock geschrieben wurden, wird an den zweiten und dritten Adreßplatz die Größe des Datenvolumens für dieses Subsystem und die den Daten zugehörige Ereignisnummer geschrieben. Schlußendlich wird die erste Speicherstelle mit einem Wert überschrieben, der der R/O-Mutterkarte anzeigt, daß der betreffende Ringspeicherblock vollständig gefüllt ist.

Ein Task auf der R/O-Mutterkarte arbeitet sequentiell die Einträge in der Messagequeue ab und kontrolliert reihum in den VIC-Karten, ob die Ringspeicherblöcke für das jeweilige Ereignis gefüllt sind. Zusätzlich wird überprüft, ob die Ereignisnummer in den Ringspeicherblöcken mit der in der Messagequeue übereinstimmt. Vollständige Blöcke werden über den VMV-Bus und die lokale VIC-Karte in den Speicher $\boxed{3}$ der R/O-Mutterkarte übertragen. Danach wird der Ringspeicherblock in der VIC-Karte wieder freigegeben. Zusätzlich wird aus der Datenstruktur mit den Einträgen der STC-Kartenregister eine BOS-Bank erzeugt und ebenfalls nach $\boxed{3}$ geschrieben. Weitere globale FTT-Bänke können an dieser Stelle ebenfalls erzeugt werden, die zum Beispiel Reaktionszeiten bei der Auslese der FTT-Subsysteme enthalten. Sind alle Daten in $\boxed{3}$ gespeichert, wird dieser Speicherbereich komplett in den für dieses Ereignis reservierten MEB über-

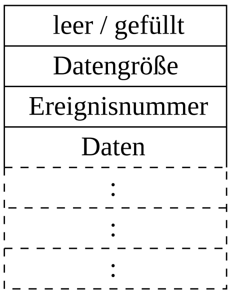



*Abbildung A.6:* Aufbau eines Ringspeicherblocks. Durch Überschreiben der ersten Speicherstelle wird der R/O-Mutterkarte die Vollständigkeit der Daten mitgeteilt. Darüberhinaus benötigt diese die Größe der zu übertragenden Daten und die assoziierte Ereignisnummer. Daran schließen sich die eigentlichen Daten an.

tragen und wieder freigegeben. Schließlich wird der TAXI-Karte mitgeteilt, daß dieser MEB nun der H1-CDAQ übergeben werden kann. Damit ist die Datenauslese eines Ereignisses aus Sicht des FTT beendet.

## A.3 Systemverhalten

Das FTT-Auslesesystem kann für verschiedene Testszenarien konfiguriert werden. Zum einen ist es wie oben erwähnt möglich, die Fast-Karte von der CTL entkoppelt zu betreiben und lokale Triggersequenzen zu generieren, oder die Triggersignale von der CTL direkt zu beziehen. Darüberhinaus kann der Status der TAXI-Karte ignoriert und die Daten in eine Datei geschrieben werden. So ist es beispielsweise möglich, Tests parallel zur laufenden H1-Datennahme durchzuführen, ohne diese zu stören. Hierzu können Runs lokal im FTT gestartet und gestoppt werden.

Zunächst wurde nur das STC-Crate in das H1-Datennahmesystem integriert. Alle STC-Signale werden dabei durch die R/O-Mutterkarte zuverlässig erkannt und bearbeitet. Die Synchronisation mit den aus der TAXI-Karte gelesenen Runstatus funktioniert ebenfalls, so daß dieses minimale FTT-Auslesesystem ohne Probleme bei der H1-Datennahme mitlaufen konnte.

Da sich der FTT zur Zeit noch in der Inbetriebnahmephase befindet, konnte noch nicht das Verhalten des vollständigen Systems getestet werden. Maximal wurden neben dem STC-Crate bisher zwei Subsysteme betrieben, deren Auslesedaten meistens von der CPU-Karte generiert wurden. Damit war es möglich zu zeigen, daß auch bei der Übertragung von jeweils 4 kBytes Daten aus beiden Subsystemen bei einer zur Zeit typischen, maximalen **L2Keep**-Rate von 100 Hz das FTT-Auslesesystem funktioniert. Diese Limitierung war durch die aktive Triggerstufe L4 gegeben und konnte nicht umgangen werden, da die Tests während der normalen H1-Datennahme stattfanden. Darüberhinaus konnte verifiziert werden, daß die Daten vollständig und fehlerfrei durch das gesamte System übertragen werden.

Schließlich stand eine FEM-Karte für weitere Tests zu Verfügung, bei der die von FADCs digitalisierten Spurkammersignale aus den anschließenden FPGAs ausgelesen

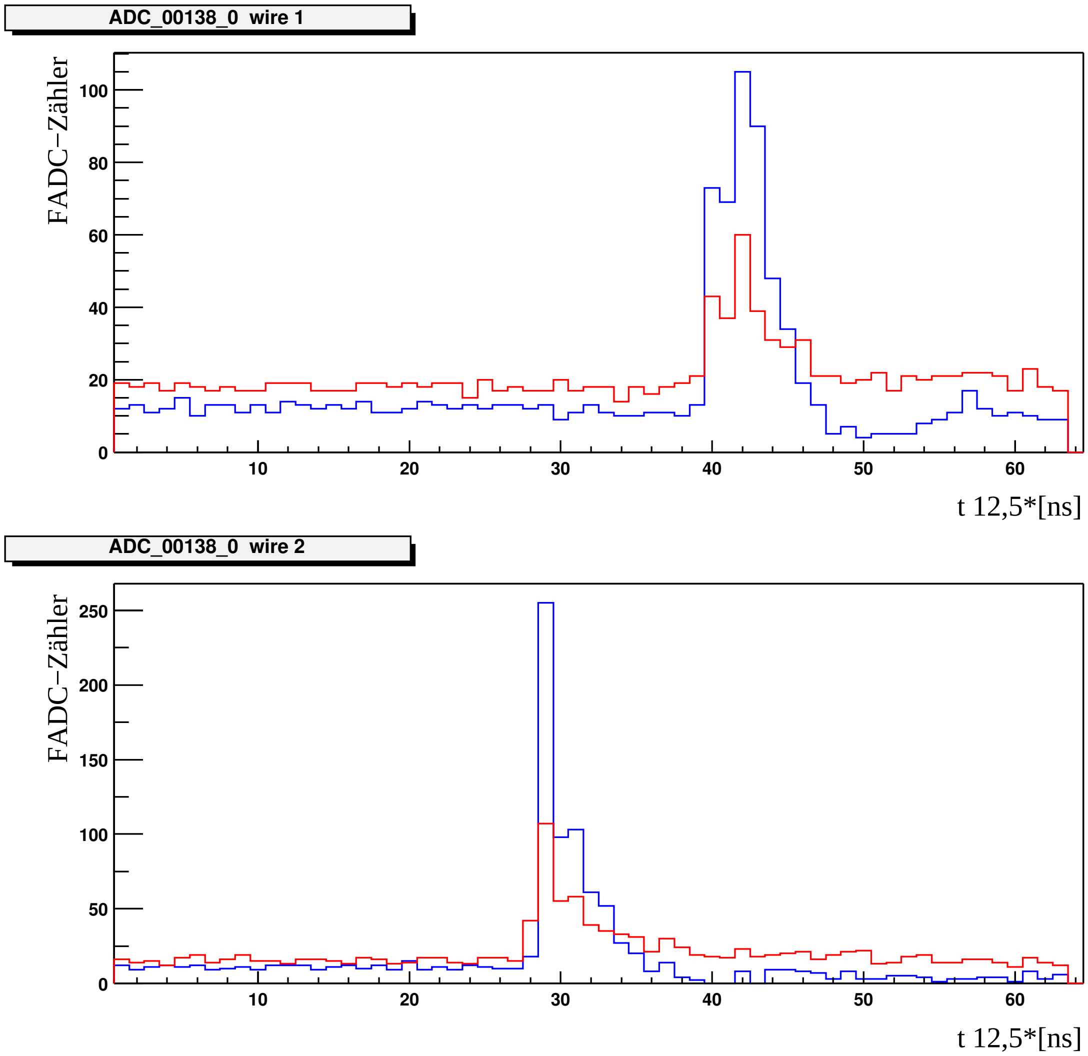



*Abbildung A.7:* Durch Testpulse erzeugte Spurkammersignale, die durch FADCs einer FTT-FEM-Karte digitalisiert und durch das FTT-Auslesesystem der H1-CDAQ zugeführt wurden. Die beiden Kurven einer Abbildung zeigen die Signale an beiden Enden des Drahtes.

werden konnten. Hierbei wurde ebenfalls ein Datenvolumen von 4 kBytes aus dem Subsystem übertragen. Abbildung A.7 zeigt Testpulssignale zweier Spurkammerdrähte für jeweils beide Drahtenden, die aus Daten, die das gesamte H1-Auslesesystem durchlaufen haben, offline rekonstruiert wurden.

# Abkürzungsverzeichnis

**BOS** ***B**ank **O**bject **S**ystem*
Bei dem H1-Experiment verwendetes Datenformat. BOS erlaubt es, auch bei einer Programmierung mit *FORTRAN77* dynamische Datenstrukturen zu verwenden. Die Daten sind dabei in sogenannten Banken angeordnet.

**BPC** ***B**ackward **P**roportional **C**hamber*
Proportionalkammer im rückwärtigen Bereich des H1-Detektors.

**BPU** ***B**ranch **P**rocessing **U**nit*
Register in einer CPU zur Vorhersage der wahrscheinlichsten Programmverzweigungen bei konditionalen Programmausdrücken.

**BSP** ***B**oard **S**upport **P**ackage*
Programmbibliothek, die die Schnittstelle zwischen der Hardware einer Elektronikkarte und generischen Routinen eines Betriebssystems bereitstellt.

**BST** ***B**ackward **S**ilicon **T**racker*
Silizium-Spurdetektor im rückwärtigen Bereich des H1-Detektors.

**CAM** ***C**ontent **A**ddressable **M**emory*
Speicher, der die zu einem RAM inverse Funktionalität zeigt. Der Eingabewert wird als Datum interpretiert und die Adresse, an der dieser Wert gespeichert ist, ausgegeben.

**CDAQ** ***C**entral **D**ata **A**c**q**uisition*
Am H1-Experiment verwendete Bezeichnung des zentralen Datenaufzeichnungssystem.

**CIP** ***C**entral **I**nner **P**roportional Chamber*
Proportionalkammer im Zentralbereich des H1-Detektors nahe am Strahlrohr. Für die HERA-II-Datennahmeperiode ist hier die fünflagig CIP2000 eingebaut.

**CJC** ***C**entral **J**et **C**hamber*
Zwei Drahtkammern (CJC1, CJC2) im Zentralbereich des H1-Detektors.

**COP** ***C**entral **O**uter **P**roportional Chamber*
Proportionalkammer im Zentralbereich des H1-Detektors zwischen den beiden CJCs.

**COZ** ***C**entral **O**uter **Z** Chamber*
Für die Auflösung der $z$-Koordinate optimierte Drahtkammer im Zentralbereich des H1-Detektors.

**cPCI** *Compact **PCI***
Aus dem in Arbeitsplatzrechnern gebräuchlichen PCI-Bus abgeleiteter Rückwandplatinenbus.

**CPU** ***C**entral **P**rozessing **U**nit*
Kernstück eines Rechners, an dem Programme abgearbeitet werden.

**CST** ***C**entral **S**ilicon **T**racker*
Silizium-Spurdetektor im Zentralbereich des H1-Detektors direkt am Strahlrohr.

**CTL** ***C**entral **T**rigger **L**ogic*
Zentrale Steuerung und Taktgeber des H1-Datennahmesystems.

**DCA** ***D**istance of **C**losest **A**pproach*
Minimaler Abstand zwischen einer Teilchenspur und dem Ereignisursprung.

**DCR$\phi$** ***D**rift **C**hamber **R**$\phi$*
Auf CJC-Informationen basierendes Triggersystem, das Spuren in der $r$-$\varphi$-Ebene identifiziert.

**DESY** ***D**eutsches **E**lektronen-**Sy**nchrotron*
Forschungsinstitut der Helmholtz-Gesellschaft in Hamburg. Keimzelle war der dort angesiedelte Elektronensynchrotronring.

**DGLAP** ***D**okshitzer, **G**ribov, **L**ipatov, **A**ltarelli, **P**arisi*
Theoretische Physiker, die gekoppelte Evolutionsgleichungen zur Beschreibung der Protonstruktur im Rahmen der QCD entwickelt haben, die sogenannten DGLAP-Gleichungen.

**DIS** ***D**eep **I**nelastic **S**cattering*
Bezeichnet bei HERA tief-inelastische Lepton-Proton-Streuung mit einem Viererimpulsübertrag von $Q^2 \geq 1$ GeV$^2$.

**DMA** ***D**irect **M**emory **A**ccess*
Datentransfer zwischen einem Speichermedium und einer weiteren elektronischen Komponente ohne direkte Einflußnahme durch eine CPU. Dazu muß der sendenden oder empfangenen Komponente ein Satz von Adressen in diesem Speicherbereich bereitstehen, der angesprochen werden soll.

**DPIO** ***D**igital **P**arallel **In**put **O**utput*
Bezeichnung für eine Elektronikkartenfamilie der Firma VMETRO, die die Datenein- oder ausgabe paralleler Daten über digitale Datenbusse unterschiedlicher Signalstandards unterstützt.

**DSP** ***D**igital **S**ignal **P**rocessor*
Spezieller Prozessortyp, der für die schnelle Bearbeitung algebraischer Algorithmen optimiert ist.

**EOB** ***E**nd **o**f **B**lock*
Signal der DPIO-Karte, wenn ein Block von DMA-Adressen abgearbeitet ist.

**EOC** *End of Chain*
Signal der DPIO-Karte, wenn bei der automatischen Verknüpfung mehrerer DMA-Adressblöcke der letzte abgearbeitet ist oder der DMA-Transfer andersweitig abgebrochen wurde.

**EOT** *End of Transfer*
Generelle Bezeichung für ein Signal, das einzeigt, daß ein bestimmter Datentransfer beendet ist.

**EPROM** *Erasable Programmable Read-Only Memory*
Programmierbarer ROM-Baustein, dessen Inhalt zur Neuprogrammierung durch Einstrahlung von UV-Licht oder ein elektrisches Signal gelöscht werden kann.

**ET** *Electron Tagger*
Detektorsystem entlang des auslaufenden Elektronstrahlt bei H1 zum Nachweis von unter kleinen Winkeln gestreuten Elektronen. Teile davon werden dabei zur Bestimmung der Luminosität verwendet.

**FADC** *Flash Analog Digital Converter*
Schneller elektronischer Baustein, der Amplituden analoger Signale mißt und in digitaler Form ausgibt. Die Zahl der Bits auf der Ausgangsseite ist dabei ein Maß für das Auflösungsvermögen des FADCs.

**FEM** *Front End Modul*
Bezeichnung für eine Elektronikkarte, die bei der ersten Stufe des FTT zum Einsatz kommt. Durch ein FEM werden Spurkammersignale digitalisiert und Spursegmente in den CJCs gesucht.

**FER** *Front End Ready*
Signal an die CTL das anzeigt, daß ein H1-Subsystem vollständig ausgelesen und bereit für die Aufzeichung des nächsten Ereignisses ist.

**FIFO** *First In First Out*
Elektronischer Speicherbaustein, aus dem Daten nur in der Reihenfolge des Beschreibens wieder ausgelesen werden können.

**FlashROM** *Flash Read Only Memory*
Speicherbaustein, der nur als Ganzes beschrieben werden kann und seinen Inhalt auch bei Stromabschaltung konserviert.

**FNE** *FIFO not empty*
Signal der DPIO-Karte, das anzeigt, daß mindestens ein Datenwort im FIFO gespeichert ist.

**FPDP** *Front Panel Data Port*
Bussystem zur parallelen Datenübertragung zwischen Karten in einem gemeinsamen Rückwandsystem. Ein fester Sender kann dabei mehrere Empfänger gleichzeitig mit den identischen Daten speisen.

**FPGA** *Field Programmable Gate Array*
Elektronikbaustein mit reprogrammierbaren Logikzellen, der in speziellen Programmiersprachen geschriebene Algorithmen mit hoher Geschwindigkeit ausführen kann.

**FPU** *Floating Point Unit*
Recheneinheit in einer CPU, die Operationen mit Fließkommazahlen ausführt.

**FST** *Forward Silicon Tracker*
Silizium-Spurdetektor im vorwärtigen Bereich des H1-Detektors.

**FTD** *Forward Tracking Detector*
Zusammenfassende Bezeichnung für Spurkammern in vorwärtigen Bereich des H1-Detektors.

**FTP** *File Transfer Protocol*
Netzwerkprotokoll zur Übertragung ganzer Dateien.

**FTT** *Fast Track Trigger*
Schneller Spurtrigger des H1-Experiments.

**FTT-R/O** *FTT Read-Out*
Bezeichnung für das Auslesesystem des FTT.

**HERA** *Hadron-Elektron-Ringanlage*
Elektron-Proton-Speicherring am DESY. Durch Umbau des ursprünglichen Rings kann die Luminosität bei HERA-II etwa verfünffacht werden.

**IET** *Inclusiv Electron Trigger*
Triggersystem, das auf Energiedepositionen im rückwärtigen H1-Kalorimeter SpaCal sensitiv ist.

**IU** *Integer Unit*
Recheneinheit in einer CPU, die Operationen mit ganzen Zahlen ausführt.

**ISR** *Interrupt Service Routine*
Spezielle Routine in einem Programmpaket, die mit einem Interrupt verknüpft wird und diesen bearbeitet.

**JTAG** *Joint Test Action Group*
Durch IEEE standartisiertes, bitweises Übertragungsprotokoll zum Laden eines elektronischen Bausteins von einem Arbeitsplatzrechner.

**LIET** *Local Inclusiv Electron Trigger*
Feinsegmentierte IET-Informationen, die durch den PQZP-Bus übertragen werden.

**LSU** *Load/Store Unit*
Datenverwaltungseinheit in einer CPU, die die Daten- und Befehlsübertragung zwischen Recheneinheiten und Speicher kontrolliert.

**L3KEEP** hspace1cm
Durch die CTL generierte Entscheidung, ein Ereignis über die dritte Triggerstufe hinaus zu bearbeiten.

**L3REJECT** hspace1cm
Durch die CTL generierte Entscheidung, ein Ereignis auf der dritten Triggerstufe zu verwerfen.

**L2KEEP** hspace1cm
Durch die CTL generierte Entscheidung, ein Ereignis über die zweite Triggerstufe hinaus zu bearbeiten.

**LVDS** ***L****ow* ***V****oltage* ***D****ifferential* ***S****ignal*
Differentieller Signalstandard.

**LVTTL** ***L****ow* ***V****oltage* ***TTL***
Signalstandard der logisch dem TTL-Standard entspricht, wobei eine logische 1 allerdings durch nur ein Potential von 3,3 V realisiert wird.

**MEB** ***M****ulti* ***E****vent* ***B****uffer*
Speicherblock in der TAXI-Karte, der die Daten eines H1-Subsystems für ein Ereignis aufnimmt.

**MPB** ***M****ulti* ***P****urpose* ***B****oard*
Von SCS entwickelte Elektronikkarte, die in verschiedenen Bestückungs- und Programmiervarianten in der zweiten Stufe des FTTs zum Einsatz kommt. Sie bietet Steckplätze für maximal vier SCS-Aufsteckkarten, drei FPGAs und vier DSPs.

**MUX** ***Mu****ltiple****x****er*
Elektronisches Bauteil, das Daten aus mehreren Quellen bündelt und über eine gemeinsame Leitung seriell weiterverschickt.

**MVME** ***M****otorola* ***VME***
Prozessorkartenserie von Motorola, die eine Schnittstelle zu einer VME-Rückwandplatine haben.

**NFS** ***N****etwork* ***F****ile* ***S****ystem*
Protokoll zum Bearbeiten von Dateien auf nicht-lokalen Festplatten über das Netzwerk. Hierbei muß nicht zwangsläufig die Datei vollständig übertragen werden.

**PCI** ***P****eripheral* ***C****omponent* ***I****nterconnect*
Für Arbeitsplatzrechner entwickeltes Bus-System zur Anbindung von Peripheriegeräten an den Prozessor.

**PIO** ***P****arallel* ***I****nput* ***O****utput*
Zwei Signalleitungen des FPDP-Busses, deren Bedeutung und Übertragungsrichtung frei konfigurierbar sind. Die DPIO-Karten sind so aufgebaut, daß diese Signale PCI-Interrupts auslösen können.

**PLL** ***P**hase **L**ocked **L**oop*
Elektronische Schaltung, die in einer festen Phasenbeziehung den Takt eines Oszillators unter- oder übersetzen kann.

**PMC** ***P**CI **M**ezzanine **C**ard*
Standatisierter Formfaktor für PCI-Aufsteckkarten.

**PowerPC** ***P**erformance **O**ptimized **W**ith **E**nhanced **R**ISC microprocessor architecture for **P**ersonal **C**omputers*
Prozessorachitektur von IBM/Motorola/Apple.

**PPC** *Power**PC***
Prozessorachitektur von IBM/Motorola/Apple.

**PQZP** ***P**arallel **Q**uickbus **Z**ero-Suppression **P**rocessor*
Bussystem zur Übertragung von 16 Bits breiten Datenworten, der im H1-Experiment für den L2L3-Bus verwendet wird.

**QCD** ***Q**uanten**c**hromo**d**ynamik*
Eichtheorie zur Beschreibung der durch Gluonen zwischen Quarks übertragenen starken Wechselwirkung.

**RAM** ***R**andom **A**ccess **M**emory*
Speicher, dessen Inhalt byteweise addressiert, beschrieben und ausgelesen werden kann.

**RES** ***Re**erved **S**ignal*
Drei Signale des FPDP-Busses, die für spezielle Aufgaben reserviert sind. Typische Anwendung ist die Übertragung eines EOT-Signals.

**RISC** ***R**educed **I**nstruction **S**et **C**omputer*
Bezeichnung für Prozessoren, die alle Befehle durch einen reduzierten Satz kurzer Instruktionen realisieren können.

**R/O** ***R**ead **O**ut*
Bei H1 gebräuchliche Bezeichung für das Datenauslesesystem.

**ROM** ***R**ead **O**nly **M**emory*
Generell Speicher, der nur ausgelesen, nicht aber beschrieben werden kann. Spezielle ROM-Bauarten sehen allerdings ein komplettes oder blockweises Beschreiben vor.

**RTOS** ***R**eal**t**ime **O**perating **S**ystem*
Betriebssystem, das kurze Antwortzeiten und innere Reaktionszeiten garantiert und somit auch in zeitkritischen Anwendungen deterministische Resultate liefert.

**RTOS-UH** ***R**eal**t**ime **O**perating **S**ystem **U**ni **H**annover*
An der Universität Hannover entwickeltes RTOS mit sehr kleinem Betriebssystemkern.

**SCS** *Super Computing Systems*
Schweizer Firma mit Sitz in Zürich, die an der Entwicklung des FTTs beteiligt war.

**SLAC** *Stanford Linear Accelerator*
Linearbeschleuniger für Elektronen und Positronen am gleichnahmigen Forschungsinstitut in Kalifornien.

**SMP** *Symmetric Multiprocessing*
Gleichberechtigter und gleichzeitiger Einsatz mehrerer Prozessoren zur Lösung eines Rechenproblems unter der Verwaltung einer Instanz eines Betriebssystems.

**SPEC** *Standard Performance Evaluation Corporation*
Arbeitsgruppe, die durch standartisierte Tests die Leistungsfähigkeit von Prozessoren bestimmt.

**SRU** *System Register Unit*
Einheit in einem Prozessor, die den Datentransfer innerhalb des Prozessors steuert und systemnahe Befehle ausführt.

**ST** *Subtrigger*
Logische Verknüpfung von TEs, die für jede Triggerstufe bestimmt werden und zur finalen Triggerentscheidung der jeweiligen Stufe herangezogen werden.

**STC** *Subsystem Trigger Controller*
Schnittstelle zwischen H1-Subsystemen und der CTL.

**SYSMODE** *SYStem Mode*
Register in der TAXI-Karte, das den aktuellen Zustand der H1-DAQ anzeigt.

**TAXI-Karte** *VME-TAXI-Karte*
VME-Karte der H1-DAQ mit TAXI-Baustein zur optischen Datenübertragung.

**TE** *Trigger Element*
Bitweise Informationen, in denen von den Triggersystemen bestimmte Ereignischarakteristika kodiert an die CTL gesendet werden.

**TTL** *Transistor Transistor Logic*
Digitale Logikstandard, bei dem eine logische 0 durch ein Potential von 0 V, eine logische 1 durch ein Potential von 5 V dargestellt wird.

**TOF** *Time of Flight*
Aus Szintillatoren aufgebautes Detektorsystem, das Flugzeiten von Teilchen entlang des Strahlrohrs oder vom Wechselwirkungspunkt aus mißt. Dies stellt ein effektives System zur Untergrundunterdrückung dar.

**VIC** *Vertical Interconnection*
Elektronik-Karte der Firma CES, die eine Verbindung zwischen VME und VMV-Bus herstellt.

**VME** *VERSAmodule Eurocard*
Standartisiertes Bussystem für Rückwandplatinen.

**VMV** *Bedeutung unbekannt*
Durch Flachbandkabel realisiertes Bussystem zur Verbindung mehrerer VME-Systeme.

# Literaturverzeichnis

[Abt-1997] I. Abt *et al.* [H1 Collaboration],
"The H1 Detector at HERA",
Nucl. Instrum. Meth. A **386** (1997) 310 und 348.

[Ada-1996] M. R. Adams *et al.* [E665 Collaboration],
"Proton and deuteron structure functions in muon scattering at 470-GeV",
Phys. Rev. D **54** (1996) 3006.

[Adl-1999] C. Adloff *et al.* [H1 Collaboration],
"Measurement of D* meson cross sections at HERA and determination of the gluon density in the proton using NLO QCD",
Nucl. Phys. B **545** (1999) 21
[arXiv:hep-ex/9812023].

[Adl-2000] C. Adloff *et al.* [H1 Collaboration],
"Measurement of neutral and charged current cross-sections in positron proton collisions at large momentum transfer",
Eur. Phys. J. C **13** (2000) 609
[arXiv:hep-ex/9908059].

[Adl-2001a] C. Adloff *et al.* [H1 Collaboration],
"Deep-inelastic inclusive e p scattering at low x and a determination of alpha(s)",
Eur. Phys. J. C **21** (2001) 33
[arXiv:hep-ex/0012053].

[Adl-2001b] C. Adloff *et al.* [H1 Collaboration],
"Measurement of neutral and charged current cross sections in electron proton collisions at high Q**2",
Eur. Phys. J. C **19** (2001) 269
[arXiv:hep-ex/0012052].

[Alt-1977] G. Altarelli and G. Parisi,
"Asymptotic Freedom In Parton Language",
Nucl. Phys. B **126** (1977) 298.

[Alt-2002a] Altera Corporation,
"APEX 20K - Programmable Logic Device Family",
verfügbar unter http://www.altera.com.

[Alt-2002b] Altera Corporation,
"Implementing High-Speed Search Applications with Altera CAM",

Application Note 119,
verfügbar unter http://www.altera.com.

[And-1988] V. Andreev *et al.*,
"Electron Tagger Trigger and DAQ",
Interne H1-Note, H1-05/88-086.

[And-1996] V. Andreev,
"Acceptance Determination of Electron Tagger (ET44) in 1995",
Interne H1-Note, H1-10/96-493.

[And-1998] V. Andreev *et al.*,
"Proposal for an Upgrade of the H1 Luminosity System and its Associated Electronics for HERA2000",
Interne H1-Note, H1-06/98-544, DESY PRC - 98/05.

[And-2000] V. Andreev *et al.*,
"ET-8 Tagger Performance and User Guide",
Interne H1-Note, H1-06/00-586.

[And-1993] B. Andrieu *et al.* [H1 Calorimeter Group Collaboration],
"The H1 liquid argon calorimeter system",
Nucl. Instrum. Meth. A **336** (1993) 460.

[ANS-1987] American National Standards Institute, Institute of Electrical and Electronics Engineers, Inc.,
"Versatile Backplane Bus: VMEbus",
ANSI/IEEE Standard 1014-1987.

[ANS-1994a] American National Standards Institute, VMEbus International Trade Association,
"VME64 Specification",
ANSI/VITA 1-1994

[ANS-1994b] American National Standards Institute, VMEbus International Trade Association,
"RACEway Specifications",
ANSI/VITA 5b-1994.

[ANS-1995] American National Standards Institute, VMEbus International Trade Association,
"Industry Pack Modules",
ANSI/VITA 4-1995.

[ANS-1997] American National Standards Institute, VMEbus International Trade Association,
"VME64 Extension",
ANSI/VITA 1.1-1997

[ANS-1998] American National Standards Institute, VMEbus International Trade Association,
"Front Panel Data Port Specifications",
ANSI/VITA 17-1998.

[Arn-1997] M. Arneodo *et al.* [New Muon Collaboration],
"Measurement of the proton and deuteron structure functions, F2(p) and F2(d), and of the ratio sigma(L)/sigma(T)",
Nucl. Phys. B **483** (1997) 3
[arXiv:hep-ph/9610231].

[Bai-2001a] A. Baird *et al.*,
"A Fast Track Trigger for the H1 Collaboration",
Nucl. Instrum. Meth. A **461** (2001) 461.

[Bai-2001b] A. Baird *et al.*,
"A fast high resolution track trigger for the H1 experiment",
IEEE Trans. Nucl. Sci. **48** (2001) 1276
[arXiv:hep-ex/0104010].

[Bar-1988a] E. Barrelet *et al.*,
"The Software L3 Triggers in H1",
Interne H1-Note H1-09/88-92.

[Bar-1988a] E. Barrelet *et al.*,
"The Hardware Implementation of L3 Triggers in H1",
Interne H1-Note H1-12/88-100.

[Bau-2002] S. Baumgartner,
"The Graphical User Interface (fttgui) for the Fast Track Trigger FTT",
verfügbar unter http://www-h1.desy.de/~simonba/fttgui.html.

[Bec-2000] J. Becker,
"The Data Acquisition and Control System for a Fast Trigger at H1",
Diplomarbeit, Universität Heidelberg (2000).

[Beh-2002] O. Behrendt,
"Entwicklung von Algorithmen zur Identifikation von Vektormeson-Ereignissen mit dem neuen H1-Spurtrigger",
Diplomarbeit, Universität Dortmund (2002).

[Bei-1998] C. Beigbeider *et al.*,
"Level 2 Topological Trigger (L2TT) Hardware",
Interne H1-Note H1-07/98-547.

[Ben-1989] A. C. Benvenuti *et al.* [BCDMS Collaboration],
"A High Statistics Measurement Of The Proton Structure Functions F(2) $(x, Q^2)$ And R From Deep Inelastic Muon Scattering At High $Q^2$",
Phys. Lett. B **223** (1989) 485.

[Ber-2002] N. Berger,
"Development of a z-vertex trigger based on drift chamber signals at H1",
Diplomarbeit, ETH Zürich (2002).

[Biz-1991] J. C. Bizot *et al.*,
"Proposal for a Topological Level 2 Trigger",
Interne H1-Note H1-06/91-181.

[Biz-2000] L. Bizzarro, VSYSTEMS Electronic GmbH,
persönliche Mitteilung.

[Bjo-1969] J. D. Bjorken and E. A. Paschos,
"Inelastic Electron Proton And Gamma Proton Scattering, And The Structure Of The Nucleon ",
Phys. Rev. **185** (1969) 1975.

[Blo-1969] E. D. Bloom *et al.*,
"High-Energy Inelastic *ep* Scattering At 6-Degrees And 10-Degrees",
Phys. Rev. Lett. **23** (1969) 930.

[Blo-2001] V. Blobel, "The BOS system - Bank Object System", H1 Softwaremanual, Fourth updated printing, 2001.

[Bot-2000] M. Botje,
"A QCD analysis of HERA and fixed target structure function data",
Eur. Phys. J. C **14** (2000) 285
[arXiv:hep-ph/9912439].

[Bou-1995] V. Boudry *al.*,
"The Inclusive Electron Trigger for the SpaCal: Design and CERN-Test Results",
Interne H1-Note H1-03/95-430.

[Bre-1969] M. Breidenbach *et al.*,
"Observed Behavior Of Highly Inelastic Electron - Proton Scattering,"
Phys. Rev. Lett. **23** (1969) 935.

[Bre-2000] J. Breitweg *et al.* [ZEUS Collaboration],
"Measurement of high-$Q^2$ charged-current $e^+p$ deep inelastic scattering cross sections at HERA",
Eur. Phys. J. C **12** (2000) 411
[arXiv:hep-ex/9907010].

[Bro-1991] I. N. Bronstein, K. A. Semendjajew,
"Taschenbuch der Mathematik",
Teubner Verlagsgesellschaft, Stuttgart, Leipzig, 25.Auflage (1991).

[Cal-1995] B. Callaghan, B. Pawlowski, P. Staubach,
"NFS Version 3 Protocol Specification",
Network Working Group, RFC 1813 (1995).

[Cam-1992] A. J. Campbell,
"A RISC multiprocessor event trigger for the data acquisition system of the H1 experiment at HERA",
IEEE Trans. Nucl. Sci. **39** (1992) 255.

[Cam-1998] A. J. Campbell *et al.*,
"Proposal to merge Level-4 and Level-5 Systems of the H1 Experiment",
Proposal submitted to the Physics Reseach Committee,
Interne H1-Note H1-12/98-558.

[CES-1992] CES Creative Electronic Systems S.A.,
"VIC 8250 VMV to VME One Slot Interface – User's Manual",
Version 3.0, 1992.

[Cha-1932a] J. Chadwick,
"Possible Existence Of A Neutron",
Nature **129** (1932) 312.

[Cha-1932b] J. Chadwick,
"The Existence Of A Neutron",
Proc. Roy. Soc. A **136** (1932) 692.

[Che-2001] S. Chekanov *et al.* [ZEUS Collaboration],
"Measurement of the neutral current cross section and F2 structure function for deep inelastic e+ p scattering at HERA",
Eur. Phys. J. C **21** (2001) 443
[arXiv:hep-ex/0105090].

[Cuj-1998] M. Cuje *et al.*,
"H1 High Luminosity Upgrade 2000 – CIP and Level 1 Vertex Trigger",
DESY PRC 98/02.

[Dok-1977] Y. L. Dokshitzer,
"Calculation Of The Structure Functions For Deep Inelastic Scattering And $e^+e^-$ Annihilation By Perturbation Theory In Quantum Chromodynamics. (In Russisch),"
Sov. Phys. JETP **46** (1977) 641 [Zh. Eksp. Teor. Fiz. **73** (1977) 1216].

[Dow-1993] K. Dowd,
"High Performance Computing",
erschienen bei O'Reilly & Associates, Inc. (1993).

[Dre-1984] E. Dreisigacker,
"Der Hadron Elektron Speicherring HERA",
Phys. Bl. **40** (1984) 153.

[Eck-2000] G. Eckerlin, DESY Hamburg,
persönliche Mitteilung.

[Fey-1969] R. P. Feynman,
"Very High-Energy Collisions Of Hadrons",
Phys. Rev. Lett. **23** (1969) 1415.

[Ful-1998] S. Fuller,
"Motorola's AltiVec™ Technology",
White Paper, Motorola Inc. (1998).

[Gei-1909] H. Geiger, E. Marsden,
"On a Diffuse Reflection of the $\alpha$-Particles",
Proc. Roy. Soc. A **82** (1909) 495.

[Gel-1964] M. Gell-Mann,
"A Schematic Model Of Baryons And Mesons",
Phys. Lett. **8** (1964) 214.

[Ger-1999] Prof. Dr.-Ing. W. Gerth,
"RTOS-UH",
Universität Hannover (1999),
verfügbar unter http://www.irt.uni-hannover.de.

[Giw-1998] GI-Working Group 4.4.2,
"PEARL90 – Language Report",
Version 2.2, (1998), verfügbar unter http://www.irt.uni-hannover.de.

[Glü-1991] M. Glück, E. Reya and A. Vogt,
"Parton distributions for high-energy collisions",
Z. Phys. C **53** (1992) 127.

[Glü-1998] M. Glück, E. Reya and A. Vogt,
"Dynamical parton distributions revisited",
Eur. Phys. J. C **5** (1998) 461,
[arXiv:hep-ph/9806404].

[Gog-1994] N. Gogitidze.
"H1 Lumi System Status Report",
H1 Kollaborationstreffen, Hamburg, 1994.

[Gor-1999] A. Gore,
"Fantastic Four – With the Power MAC G4, APPLE brings super power to your desktop",
verfügbar unter http://www.macworld.com.

[Gri-1972] V. N. Gribov and L. N. Lipatov,
"Deep Inelastic $ep$ Scattering In Perturbation Theory",
Yad. Fiz. **15** (1972) 781, [Sov. J. Nucl. Phys. **15** (1972) 438].

[Gri-1972] V. N. Gribov and L. N. Lipatov,
"$e^+e^-$ Pair Annihilation And Deep Inelastic E P Scattering In Perturbation Theory",
Yad. Fiz. **15** (1972) 1218, [Sov. J. Nucl. Phys. **15** (1972) 675].

[Gro-1973] D. J. Gross and F. Wilczek,
"Asymptotically Free Gauge Theories. I",
Phys. Rev. D **8** (1973) 3633.

[Gro-1973] D. J. Gross and F. Wilczek,
"Ultraviolet Behavior Of Non-Abelian Gauge Theories",
Phys. Rev. Lett. **30** (1973) 1343.

[Gro-1974] D. J. Gross and F. Wilczek,
"Asymptotically Free Gauge Theories. 2",
Phys. Rev. D **9** (1974) 980.

[H1-1996] H1 Collaboration,
"Luminosity Measurement in the H1 Experiment at HERA",
Paper submitted to the 28th International Conference on High Energy Physics ICHEP'96, Warsaw, Poland (1996).

[H1-1997] H1 Collaboration,
"ep Physics beyond 1999",
Interne H1-Note, H1-10/97-531.

[H1-1999a] H1 Collaboration,
"A Forward Silicon Tracker for H1",
DESY PRC 99-01 (1999).

[H1-1999b] H1 Collaboration,
"Proposal to Upgrade the LAr Calorimeter Trigger: The Jet Trigger",
DESY PRC 99-02 (1999).

[H1-1999c] H1 Collaboration,
"A Fast Track Trigger with High Resolution for H1",
Interne H1-Note, H1-06/99-573, DESY PRC 99-06 (1999) and
"Addendum to the proposal *A Fast Track Trigger with High Resolution for H1*",
Interne H1-Note, H1-09/99-576.

[H1S-1997] H1 SpaCal Group,
"The H1 Lead/Scintillating-Fibre Calorimeter",
Nucl. Instrum. Meth. A **386** (1997) 397.

[Hag-2002] K. Hagiwara *et al.* [Particle Data Group Collaboration],
"Review Of Particle Physics",
Phys. Rev. D **66** (2002) 010001.

[Hay-1993] W. J. Haynes,
"VMExi2_SSP – VMEtaxi Mark-2 System Software Package",
Version 4.2, 1993.

[Hei-1927] W. Heisenberg,
"Über den anschaulichen Inhalt der quantentheoretischen Kinematik

und Mechanik",
Z. Phys. **43** (1927) 172

[Hen-2000] H. Henschel and R. Lahmann,
"The Backward Silicon Tracker Of The H1 Experiment At Hera",
Nucl. Instrum. Meth. A **453** (2000) 93.

[Hof-2000] D. Hoffmann,
"Zwei-Elektron-Ereignisse im H1-Detektor",
Dissertation, Universität Hamburg (2000).

[Hof-2002] G. Hofmann,
VSYSTEMS Electronic GmbH,
persönliche Mitteilung.

[IEE-1995b] Institute of Electrical and Electronics Engineers, Inc.,
"IEEE - Common Mezzanine Card Specification (CMC)",
P1386.1 Draft 2.0. (1995).

[Ing-1996] G. . Ingelman, A. . De Roeck and R. . Klanner,
"Future physics at HERA",
Proceedings, Workshop, Hamburg, Germany, September 25, 1995-May 31, 1996. Vol. 1, 2,
DESY-96-235.

[Jun-2002] A. Jung,
"A Jet-Searching Algorithm for the FTT-Trigger",
DESY Summer Student Report (2002).

[Kar-1991] V. Karimaki,
"Effective Circle Fitting For Particle Trajectories",
Nucl. Instrum. Meth. A **305** (1991) 187.

[Köh-1997] J. K. Köhne *et al.*,
"Realization Of A Second Level Neural Network Trigger For The H1 Experiment At Hera",
Nucl. Instrum. Meth. A **389** (1997) 128.

[Kol-2002] M. Kolander,
Elektronikentwicklung, Universität Dortmund,
persönliche Mitteilung.

[Krä-1998] T. Krämerkämper,
"Messung der Gluondichte im Photon und Entwicklung eines Neuronalen Triggers",
Dissertation, Universität Dortmund (1997).

[Kre-1989] H. Krehbiel,
"The Fast Card of the Subsystem Trigger Controller",
H1-Internes User's Manual (1989).

[Kre-1991] H. Krehbiel,
"The Extended Fan Out Card of the H1 STC",
H1-Internes User's Manual (1991).

[Lai-1996] H. L. Lai *et al.*,
"Improved parton distributions from global analysis of recent deep inelastic scattering and inclusive jet data",
Phys. Rev. D **55** (1997) 1280
[arXiv:hep-ph/9606399].

[Luk-60 v.Chr.] T.L.C. Lukrez,
"De rerum natura",
ca. 60 v.Chr.

[Lyn-2001] Lynux Works, Inc.,
"LynxOS User's Guide",
LynxOS Release 4.0, DOC-0453-01 (2001),
verfügbar unter http://www.lynuxworks.com/

[Mac-2002] Macinfo,
"SPECmark95",
verfügbar unter http://www.macinfo.de.

[Mee-2002] D. Meer, D. Müller, J. Müller, A. Schöning and C. Wissing,
"A multifunctional processing board for the fast track trigger of the H1 experiment",
IEEE Trans. Nucl. Sci. **49** (2002) 357
[arXiv:hep-ex/0107010].

[Mer-2002] J. Merkel,
"$J/\Psi$-vector meson identification with the Fast Track Trigger and the Central $\mu$-Detector",
DESY Summer Student Report (2002).

[Mot-1994] Motorola Inc. Computer Group,
"MVME 2400 Series VME Processor Modules",
Datenblatt verfügbar unter http://mcg.motorola.com.

[Mot-1997] Motorola Literature Distribution Center,
"PowerPC 750 RISC Microprocessor Technical Summary",
Datenblatt MPC750/D, 1997,
verfügbar unter http://e-www.motorola/com (1997).

[Mot-1999] Motorola Inc. Computer Group,
"MVME 5100 Series VME Processor Modules",
Datenblatt verfügbar unter http://mcg.motorola.com (1999).

[Mot-2001] Motorola Inc. Computer Group,
"PPCBug Firmware Package, User's Manual Part 1 & 2",
PPCBUGA1/UM5 und PPCBUGA2/UM5, Edition 2001.

[Mot-2002] Motorola Inc. Computer Group,
"MVME 5500 Series VME Single Board Computer",
Datenblatt verfügbar unter http://mcg.motorola.com.

[Nat-2002] National Semiconductor Corporation,
"DS90CR483/DS90CR484 48-Bit LVDS Channel Link SER/DES — 33 - 112 MHz",
Version Mai 2002, verfügbar unter http://www.national.com.

[Nic-1998] T. Nicholls *et al.*,
"Concept, Design and Performance of the Second Level Triggers of the H1 Detector",
IEEE Trans. Nucl. Sci. **45** (1998) 810.

[Nut-1913] M. J. Nuttall, E. Rutherford,
"Scattering of Alpha Particles by Gases",
Phil. Mag. 26 (1913) 702.

[Ols-1990] J. Olszowska,
"The Slow Card of the Subsystem Trigger Controller",
User's Manual, 1990.

[Rut-1911] E. Rutherford,
"The scattering of $\alpha$ and $\beta$ Particles by Matter and the Structure of the Atom",
Phil. Mag. 21 (1911) 669.

[PCI-1995] PCI Special Interest Group,
"Peripheral Component Interconnect (PCI) Local Bus Specification",
Revision 2.1 (1995).

[PIC-1997] PICMG - PCI Industrial Computers Manufacturers Group,
"Compact PCI Specification",
Revision 2.1 (1997).

[Pie-1991] E. Pietarinen,
"VMEbus Cross Interface Processor Module with High Speed Fibre Optics Link: VMExi",
Report HU-SEFT-1991-14, Universität Helsinki, Finnland, 1991.

[Pit-2000] D. Pitzl *et al.*,
"The H1 silicon vertex detector",
Nucl. Instrum. Meth. A **454** (2000) 334
[arXiv:hep-ex/0002044].

[Pol-1973] H. D. Politzer,
"Reliable Perturbative Results For Strong Interactions?",
Phys. Rev. Lett. **30** (1973) 1346.

[Pos-1985] J. Postel, J. Reynolds,
"File Transfer Protocol (FTP)",
Network Working Group, RFC 959 (1985).

[Pre-1992] S. Prell,
"$z$-Kalibration und $dE/dx$-Kalibration der zentralen Spurkammer des H1-Detektors",
Diplomarbeit, Universität Hamburg (1992).

[Ryn-2002] J. Rynearson,
"From BLE to 2eSST – A Look at the Evolution of VMEbus Protocols",
verfügbar über http://www.vita.com (2002).

[Sch-1997] D. Schmidt,
"Topologischer Trigger für inelastisch produzierte $J/\Psi \to e^+e^-$ am H1-Detektor bei HERA",
Diplomarbeit, Universität Hamburg (1997).

[Sch-1998] U. Schneekloth,
"The HERA luminosity upgrade",
DESY-HERA-98-05 (1998).

[Sch-1999] H. C. Schultz-Coulon, E. Elsen, T. Nicholls, J. Coughlan and H. Rick,
"A General Scheme For Optimization Of Trigger Rates In An Experiment With Limited Bandwidth",
IEEE Trans. Nucl. Sci. **46** (1999) 915.

[Sch-2001] H. C. Schultz-Coulon, Universität Dortmund,
persönliche Mitteilung.

[Sch-2002a] H. C. Schultz-Coulon, Universität Dortmund,
persönliche Mitteilung.

[Sch-2002b] A. Schöning, ETH Zürich,
persönliche Mitteilung.

[Sef-1995] F. Sefkow, E. Elsen, H. Krehbiel, U. Straumann and J. Coughlan,
"Experience With The First Level Trigger Of H1",
IEEE Trans. Nucl. Sci. **42** (1995) 900.

[See-2002] J. Seehafer,
"Jets finden mit dem FTT",
DESY Summer Student Report (2002).

[SPE-2000] Standard Performance Evaluation Corporation,
"SPEC CPU95 Benchmarks",
verfügbar unter http://www.spec.org/osg/cpu95/.

[SPE-2002] Standard Performance Evaluation Corporation,
"SPEC CPU2000 Benchmarks",
verfügbar unter http://www.spec.org/osg/cpu2000/.

[Sta-2002] R. M. Stallman,
"Using the GNU compiler collection",
Free Software Foundation, 2002,
verfügbar unter http://gcc.gnu.org.

[Tex-2000] Texas Instruments Inc.,
"TMS320C6701 Floating Point Digital Signal Processor",
Datenblatt verfügbar unter http://www.ti.com, Mai 2000.

[Urb-2000a] M. Urban,
"Ein schneller Trigger für H1 bei HERA",
Diplomarbeit, Universität Heidelberg (2000).

[Urb-2000b] M. Urban, U. Straumann, A. Rausch and J. Becker,
"First level trigger for H1, using the latest FPGA generation",
Prepared for 6th Workshop on Electronic for LHC Experiments,
Cracow, Poland, 11-15 Sep 2000.

[VDC-1999] Venture Development Corporation,
"Real-Time/Embedded Board Market Needs for OEMs and End Users",
Bus&Board Conference (1999),
verfügbar unter http://www.vita.com.

[VIT-1999] VITA Standards Organization,
"Prozessor PMC Standards for Processor PCI Mezzanine Cards",
VITA 32-199x, Draft 0.21 (1999).

[VME-2000] VMETRO / VSYSTEMS
"DPIO - FPDP Digital Parallel I/O PMC Module",
Produktinformation, verfügbar unter http://www.vmetro.com oder http://www.vsystems.de.

[Wag-2002] J. Wagner,
"Open Charm Triggers - Triggering Charm with $D^*$ Mesons",
H1 Workshop on Post-Upgrade Triggering, Ringberg (2002).

[Wat-1994] J. Watson,
Beitrag in Newsgroup comp.os.lynx, (1994).

[Web-1998] Ph. Weber,
"The Convergence of Processor Architectures for Embedded Computers",
Real-Time Magazin **2** (1998) 52,
verfügbar unter http://www.realtime-info.be.

[Win-1999a] Wind River Systems, Inc.,
"Tornado User's Guide (Windows Version), 2.0",
Edition 1 (1999).

[Win-1999b] Wind River Systems, Inc.,
"VxWorks Programmer's Guide, 5.4",
Edition 1 (1999).

[Win-1999c] Wind River Systems, Inc.,
"VxWorks Network Programmer's Guide, 5.4",
Edition 1 (1999).

[Win-1999d] Wind River Systems, Inc.,
"GNU Toolkit User's Guide, Tornado 2.0",
Edition 1, (1999).

[Win-2002a] M. Windisch, Universität Dortmund,
persönliche Mitteilung.

[Win-2002b] M. Windisch,
Diplomarbeit, Universität Dortmund, in Vorbereitung.

[Wis-1998] C. Wissing,
"Bestimmung der Charakteristika des Flugzeitsystems des H1-Detektors zur Messung des Strahluntergrundes",
Diplomarbeit, Universität Dortmund (1998).

[Wis-2002] C. Wissing,
"Entwicklung eines Simulationsprogramms und Implementierung schneller Spurfitalgorithmen für den neuen H1-Driftkammertrigger",
Dissertation, Universität Dortmund (2003).

[Wol-1992] T. Wolff *et al.*,
"A Drift Chamber Track Finder for the first Level Trigger of the H1 Experiment",
Nucl. Instrum. Meth. A **323** (1992) 537.

[Zwe-1964] G. Zweig,
CERN Reports No. 8182/TH. 401 und No. 8419/TH. 412, 1964 (unveröffentlicht).

# Danksagung

Eine ganze Mannschaft von Leute hat am Entstehen dieser Arbeit direkten oder indirekten Einfluß gehabt. Diesen sei an dieser Stelle ein großer Dank ausgesprochen. Und nun zur Aufstellung:

Trainer:
Zuallererst möchte ich Herrn Prof. Dr. D. Wegener danken, mir die Möglichkeit gegeben zu haben, an dem aufregenden und innovativen FTT-Projekt mit einer interessanten Aufgabe beteiligt gewesen zu sein. Sein Führungsstil, jederzeit hilfsbereite Betreuung mit gleichzeitigem Gewähren aller Freiheiten zu verbinden, hat den Grundstein für eine angenehme und fruchtbare Arbeitsatmosphäre gelegt. Hoch anzurechnen ist auch sein Bemühen, durch Lesen vorher vernachlässigter Teile der FAZ immer auf Ballhöhe zu bleiben.

Co-Trainer:
Herrn Priv.-Doz. Dr. M. Schmelling danke ich für die Bereitschaft, die Aufgabe des Zweitgutachters dieser Arbeit zu übernehmen.

Spielgestalter:
In der Winterpause 98/99 wurde Herr Dr. Hans-Christian Schultz-Coulon als neuer Mittelfeldmotor verpflichtet und fand gleich die Bindung zur Mannschaft. Von seinen Erfahrungen mit Triggern, Hardware-naher Programmierung und Datentransfer habe ich sehr profitiert. Sein jederzeit offenes Ohr für Probleme hat Frustphasen erträglich, und das für Klatsch Kaffeepausen unterhaltsam gemacht. Die kritische Durchsicht dieser Arbeit war mir eine sehr große Hilfe.

Zeugwart:
Herrn Markus Kolander danke ich, die Vorgaben für die L3-Empfängerkarte umgesetzt, mit Leben gefüllt und kenntnisreich erweitert zu haben. Seine Erfahrungen bei der Elektronikentwicklung waren unersetzlich, seine gewissenhafte Arbeitsweise beeindruckend, sein Beitrag zu FTT-L3 groß. Zusätzlich sein an dieser Stelle der Abteilung für Elektronik-Entwicklung am physikalischen Institut der Universität Dortmund und der Abteilung -ZE- am DESY gedankt.

Stürmer:
Als Leiter der H1-FTT-Gruppe zeichnet sich Herr Dr. André Schöning durch den direkten Zug nach vorne aus. Seine Einstellung “geht nicht, gibt’s nicht” ist charakteristisch für das gesamte Projekt und sehr motivierend. Da er zu den wenigen gehörte, die jederzeit den Gesamtüberblick über den Status des FTT behielten, waren Diskussionen mit ihm sehr anregend. Für seine Hilfsbereitschaft beim Lösen großer und kleiner Probleme habe ich zu danken.

Mittelfeld:
Daß der “Wohlfühlfaktor” nicht zu kurz kam, war dem gut besetzten Kader des

Lehrstuhls E5 (& Assoziierte) zu verdanken. Allen gilt mein großer Dank, besonders erwähnen möchte ich aber Herrn Christoph Wissing, der unermüdlich im Einsatz war, optimale Voraussetzungen für die Arbeit an Rechnern zu schaffen, Herrn Dr. Rainer Stamen, der stets bereit war, gefragt und ungefragt, wichtige und unwichtige Beiträge zu leisten, Herrn Olaf Behrend, der schnell durch Abgänge entstandene Lücken schließen konnte und Herrn Dr. Klaus Wacker, der Teile der Arbeit linguistisch überprüft hat.

Ergänzungsspieler:
Zum erweiterten Kreis der Mannschaft gehören auch Mitglieder der FTT Gruppe, die anderen Vereinen angehören. Um eine angenehme und produktive Arbeitsatmosphäre haben sich dabei Herr Yves Flemming, Herr Simon Baumgartner und Herr Nik Berger verdient gemacht.

Kassenwart:
Nicht vergessen möchte ich an dieser Stelle, Frau Lorenz zu danken, die jedes bürokratische Versäumnis meinerseits auszubügeln wußte und für alles das passende Formular hat.

Sponsoren:
Ohne Rückendeckung im Hintergrund wäre der erfolgreiche Abschluß der Arbeit nicht möglich gewesen. Ohne die Zuwendungen meiner Eltern in allen notwendigen Formen könnte ich diese Zeilen jetzt nicht schreiben. Ich danke ihnen für die Geduld, mit der sie meinen Ausbildungsweg begleitet haben. Zu danken habe ich auch Momie und Gerd Schwarz, die völlig selbstlos (:-) einsprangen, wenn es Lücken in der Kinderbetreuung zu schließen galt und auch sonst alles Erdenkliche taten, mir Lasten abzunehmen.

Spielerfrau:
Am wichtigsten aber war mir der Rückhalt durch Sandra, die viel Verständnis für spontane Terminänderungen, physische und psychische Abwesenheit und abendliche Notebook-Sitzungen aufzubringen hatte. Für die wohldosierten Ablenkungen von der Arbeit bin ich ihr sehr dankbar.

F-Jugend:
Ein Dank für seine Geduld geht auch an Yannick, der recht schnell wußte: “Uni” bedeutet, daß Papa abends zur Gutenachtgeschichte wiederkommt, “Hamburg”, daß er frühstens in drei Tagen wiederkommt.

Diese Arbeit wurde mit Mitteln des Bundesministeriums für Bildung, Wissenschaft, Forschung und Technologie unter der Nummer 05H11PEA/6 gefördert.