Objektorientierte Simulation der Trigger Decision Unit des HERA-B First Level Triggers

Diplomarbeit
am II. Institut für Experimentalphysik
der Universität Hamburg

vorgelegt von
Michael Nörenberg
im August 1999

Objektorientierte Simulation der Trigger Decision Unit des HERA-*B* First Level Triggers

Diplomarbeit
am II. Institut für Experimentalphysik
der Universität Hamburg

vorgelegt von
Michael Nörenberg
im August 1999

Betreuer:	Prof. Dr. Albrecht Wagner
Zweitgutachter:	Dr. Andreas Schwarz

Inhaltsverzeichnis

Vorwort

Immanuel Kant untersucht in seiner 'Kritik der reinen Vernunft' die Voraussetzungen und Rechtfertigung für die Beschäftigung mit Metaphysik. Zu Kants Zeiten und auch heute noch war und ist nicht klar, ob die Metaphysik den Rang einer sicheren Wissenschaft erreichen kann, da sie sich per definitionem mit den nicht erfahrbaren Dingen beschäftigt. Nach Kant muß hierzu in der Metaphysik eine Methode gefunden werden, welche sichere Erkenntnisse hervorbringt, die eine objektive Gültigkeit beanspruchen können. Als ein Vorbild gibt Kant die Naturwissenschaft an, die Erkenntnisse mit Hilfe des *experimentellen Prinzips* gewinnt. Entscheidend ist hierbei, daß ein Experiment nicht passive Naturbeobachtung ist, sondern daß beim Forschen durch die Präparation eines Versuchs die Natur quasi dazu gezwungen wird, auf eine von den Naturwissenschaftlern gestellte Frage zu antworten. Über die ersten Naturwissenschaftler, die nach diesem Prinzip geforscht haben, sagt Kant: *"Sie begriffen, daß die Vernunft nur das einsieht, was sie selbst nach ihrem Entwurfe hervorbringt, daß die mit Prinzipien ihrer Urteile nach beständigen Gesetzen vorangehen und die Natur nötigen müsse auf ihre Fragen zu antworten, [...]"*.[1] Kants Aussage trifft gerade im Bereich der Hochenergiephysik zu. Die Reaktionen von Elementarteilchen, die Gegenstand des Interesses sind, verbergen sich hinter einer erdrückenden Flut von Hintergrundereignissen. Die Selektion der gesuchten Signale entpuppt sich als die sprichwörtliche Suche nach der Nadel im Heuhaufen. Ideal wäre es, könnten sämtliche in einem Experiment aufgetretenen Reaktionen ausführlich untersucht werden. Doch dazu reicht die verfügbare Rechenkapazität nicht aus. Die einzige Chance, sich einen Pfad durch das Dickicht der inelastischen Ereignisse zu bahnen, liegt in der vorläufigen Selektion der gemessenen Daten mit Hilfe von speziell für diesen Zweck entwickelten Kriterien, welche eine erfolgreiche Suche nach neuen Erkenntnissen versprechen. Diese vorläufige Selektion hat den Nachteil, daß auch interessante Reaktionen aussortiert werden können, da die Selektion in möglichst kurzer Zeit erfolgen muß und damit nicht den Grad an Zuverlässigkeit aufweisen kann, wie eine vollständige Untersuchung und Rekonstruktion aller Ereignisse. Die Instrumente, die der Hochenergiephysiker zu diesem Zweck heranzieht, sind die sogenannten Trigger[2]. Gemeint sind die Teile des Auslesesystems, die unmittelbar nach der Datennahme entscheiden, ob ein registriertes Ereignis gespeichert oder verworfen werden soll, da es aufgrund der Triggerkriterien als 'interessant' oder 'unerheblich' kategorisiert wurde. Die Trigger sind in der Lage, aus der unübersehbaren Vielzahl von Information diejenigen Daten herauszufiltern, die uns zu einem tieferen Verständnis der Physik verhelfen. Hierzu ist der Hochenergiephysiker gezwungen, eine Vorentscheidung zu treffen, die seine Erwartung an die zu untersuchende Physik widerspiegelt. Die Trigger sind in dieser

[1] Immanuel Kant 1787, Zweite Auflage der 'Kritik der reinen Vernunft', Seite XIII.
[2] *trigger*, engl. für 'Auslöser'.

Hinsicht eine Manifestation des von Kant formulierten Prinzips, daß wir nur das einsehen können, was unser Verstand zuvor als Frage formuliert hat. Durch die Wahl von Triggerparameter sehen wir die Natur wie durch eine eingefärbte Glasscheibe. Interessantes wird hervorgehoben, doch müssen wir stets darauf achten, dabei nicht aufschlußreichere Details zu übersehen.

Danksagung

An dieser Stelle ist es mir ein natürliches Bedürfnis, allen danken, die mir mit Rat und Tat sowie moralischer Unterstützung bei dieser Arbeit geholfen haben. Zunächst möchte ich meinen Eltern Marjut und Klaus-Dieter Nörenberg sowie meiner Schwester Kristina danken, die mir den Weg durch mein Studium geebnet haben. Besonderer Dank gilt auch Prof. Dr. Albrecht Wagner und Dr. Andreas Schwarz, die freundlicherweise neben all ihren Pflichten meine Betreuung übernommen haben. Weiterhin möchte ich die Bemühungen von Heiko Itterbeck und Ekkehard Gerndt erwähnen, die meine direkten Ansprechpartner in allen Fragen des First Level Triggers gewesen sind. Den kritischen und konstruktiven Anmerkungen von Joachim Flammer, Holger Fleckenstein, Thomas Fuljahn und Christian Rothe ist zu verdanken, daß die Arbeit in der vorliegenden Form vollendet werden konnte. Es folgen die unzähligen Kolleginnen und Kollegen, die mir Tag für Tag zur Seite standen: Georg Baki, Christoph Borgmeier, Verena Buchholz, Alexander Gröpl, Christian Haehnel, Andrea Hamel, Olya Igonkina, Dirk Kahnert, Peter Kreuzer, Birgit Lewendel, Mike Medinnis, Andrea Michetti, Isabelle Negri, Thorsten Oest, Robert Pernack, Fedor Ratnikov, Natalia Ratnikova, Dominik Ressing, Imma Riu, Dorothea Samtleben, Stefan Steinbeck, Michael Symalla, Norbert Tesch, Gregor Wagner, Mauro Villa, Raoul Zimmermann. Allen ganz herzlichen Dank :-).

Einleitung

Das primäre Ziel des Experiments HERA-B liegt im Studium der CP-Verletzung im B-Mesonen-System. Hierzu soll der Zerfall von B^0- und $\bar{B}^0$-Mesonen in den CP-Eigenzustand $J/\psi\, K_S^0$ beobachtet werden. Dieser Kanal wird durch eine um sechs Größenordnungen höhere Hintergrundrate überdeckt. Um diesen Hintergrund wirksam zu unterdrücken, ist ein vierstufiges Triggersystem vorgesehen. Bereits innerhalb der ersten Triggerstufe, die auch First Level Trigger[3], oder kurz FLT, heißt, wird eine vorläufige Spurrekonstruktion vorgenommen, die durch die Berechnung von invarianten Massen von Spurpaaren die Selektion von J/ψ-Zerfällen erlaubt. Verwirklicht wird dies durch ein Netzwerk von parallel arbeitenden Prozessoren, die speziell für diese Aufgabe entwickelt worden sind.

Die vorliegende Diplomarbeit ist einer Phase angefertigt worden, in welcher sich der HERA-B Detektor noch im Aufbau befindet. Das grundlegende Design der Detektorkomponenten ist abgeschlossen, doch treten in diesem Stadium eines Experimentes fast täglich neue Probleme auf, die stets für neue Herausforderungen sorgen. In diesem Zusammenhang ist die vorliegende Arbeit zu sehen. Die erste Triggerstufe ist komplett entworfen und die Inbetriebnahme hat ihre volle Aktivität erreicht. Dennoch ist es stets notwendig, einen Schritt zurückzugehen und das gesamte System in seiner Komplexität immer wieder neu verstehen zu können. Um dies zu ermöglichen, soll neben den bereits bestehenden Simulationen des First Level Trigger eine weitere entwickelt werden, die durch eine Modellierung der Elektronik des FLT auf Taktebene das Studium von Effekten erlaubt, die durch die konkrete Realisierung des FLT in der Hardware hervorgerufen werden. Insbesondere ist die Latenzzeit der ersten Triggerstufe von Interesse, die stark von dem einlaufenden Datenfluß in Form von Teilchenspuren pro Ereignis abhängt. Die Latenzzeit ist das Zeitintervall, welches vom Auftreten eines Ereignisses bis zu einer positiven Triggerentscheidung vergeht. Treten Ereignisse mit vielen Wechselwirkungen und somit vielen Teilchenspuren auf, so wächst die Latenzzeit an. Die gesamten Detektordaten müssen bis zu einer positiven Triggerentscheidung zwischengespeichert werden, wobei der hierfür vorgesehene Speicherplatz aus technischen und finanziellen Gründen begrenzt ist. Die gesamte Latenzzeit des First Level Trigger darf einen Wert von $12{,}3\,\mu$s nicht überschreiten, um eine totzeitfreie Datennahme zu gewährleisten.

Die Simulation des FLT auf Taktebene soll unter Verwendung objektorientierter Methoden in der Programmiersprache C++ implementiert werden. Durch einen modularen Aufbau sollen Übersichtlichkeit des Codes und Flexibilität in der Anwendung gewährleistet sein.

Im Mittelpunkt der Diplomarbeit steht die Entwicklung eines programmiertechnischen Rahmens, oder Frameworks[4], welches die Implementierung einer hardwarenahen Simula-

[3] engl. für 'Erste Auslösestufe'

[4] engl. für 'Rahmen', 'System'

tion unterstützt. Mit Hilfe dieser Basis ist eine Simulation der Trigger Decision Unit des FLT bereitgestellt worden.
Mit Hilfe dieser Simulation wurden Untersuchungen zur Präzision der Berechnung der invarianten Masse von Spurpaaren in der Trigger Decision Unit durchgeführt. Darüber hinaus sind die zu erwartenden Triggereffizienzen betrachtet worden, die bei der Ausbaustufe der ersten Triggerstufe zum Zeitpunkt der Fertigstellung dieser Arbeit (ab August '99) zu erwarten sind.

Kapitel 1

Physikalischer Hintergrund

1.1 Die CKM-Matrix

Das primäre Ziel des Experiments HERA-B ist die Untersuchung der CP-Verletzung im $B^0/\bar{B}^0$-System. Es wird hier von der Theorie ein deutlich größerer Effekt als im $K^0/\bar{K}^0$-System vorausgesagt, in dem die Verletzung der CP-Symmetrie entdeckt worden ist.
Die Koppelung der Quarks ($|u\rangle, |c\rangle, |t\rangle$) über den geladenen Strom mittels der $W^{\pm}$-Bosonen erfolgt nicht exakt an die Massen-Eigenzustände ($|d\rangle, |s\rangle, |b\rangle$), sondern an die Eigenzustände der elektroschwachen Wechselwirkung ($|d'\rangle, |s'\rangle, |b'\rangle$). Diese ergeben sich aus den Massen-Eigenzuständen mittels der Cabibbo-Kobayashi-Maskawa-Matrix V_{CKM} oder kurz CKM-Matrix [kob73]:

$$\begin{pmatrix} |d'\rangle \\ |s'\rangle \\ |b'\rangle \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \cdot \begin{pmatrix} |d\rangle \\ |s\rangle \\ |b\rangle \end{pmatrix}$$

Es gibt drei Quark-Generationen, weshalb die CKM-Matrix unitär sein muß. Die Mischung zwischen den Quark-Generation ist klein, es gilt also $V_{ii} \approx 1$ und $V_{ij} \ll 1 \; \forall \; i \neq j$. In diesem Zusammenhang hat Wolfenstein folgende nach ihm benannte Parametrisierung vorgeschlagen, in der die CKM-Matrix nach dem Cabibbo-Winkel $\lambda = \sin\theta_{Cabbibo}$ entwickelt wird [wol83]:

$$V_{CKM} = \begin{pmatrix} 1-\frac{\lambda^2}{2} & \lambda & \lambda^3 A(\rho - i\eta) \\ -\lambda & 1-\frac{\lambda^2}{2} & \lambda^2 A \\ \lambda^3 A(1-\rho-i\eta) & -\lambda^2 A & 1 \end{pmatrix} + \mathcal{O}(\lambda^4)$$

Dabei sind λ, A, ρ und η reelle Parameter. Der imaginäre Anteil der Komponenten V_{ub} und V_{td}, dessen Größe durch den Parameter η angegeben wird, ist ein Maß für die CP-Verletzung. Aus der Unitarität der CKM-Matrix ergibt sich unter anderem folgende Bedingung:

$$V_{ud}V_{ub}^* + V_{cd}V_{cb}^* + V_{td}V_{tb}^* = 0$$

Die Summanden dieser Unitaritätsbedingung bilden in der komplexen Zahlenebene ein Dreieck, welches Unitaritätsdreieck genannt wird (s. Abb. 1.1). Sollten sich die drei Seiten des Dreiecks, die sich aus der Unitaritätsbedingung ergeben, nicht schließen lassen, so wäre dies ein Hinweis auf Nichtunitarität der CKM-Matrix und damit auf über das Standardmodell hinausgehende Phänomene.

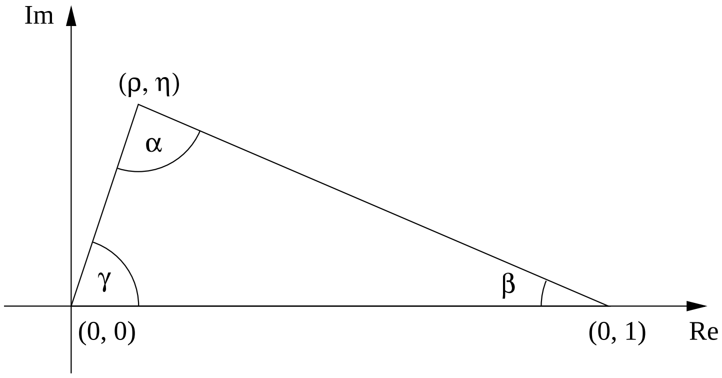

Abbildung 1.1: Das Unitaritätsdreieck in der komplexen Ebene.

1.2 CP-Verletzung im $B^0\bar{B}^0$-System durch Mischung

Zur Beschreibung der CP-Verletzung durch Mischung folgen wir der Argumentation von [zim99]. Die Verletzung der CP-Symmetrie äußert sich im $B^0\bar{B}^0$-System durch unterschiedliche Zerfallsraten von CP-konjugierten Kanälen der Form

$$B^0 \to X \quad \stackrel{CP}{\longleftrightarrow} \quad \bar{B}^0 \to \bar{X}$$

Dies führt zu einer von Null verschiedenen Asymmetrie A der Zerfallsraten Γ der betrachteten Kanäle:

$$A = \frac{\Gamma(B^0 \to X) - \Gamma(\bar{B}^0 \to \bar{X})}{\Gamma(B^0 \to X) + \Gamma(\bar{B}^0 \to \bar{X})}$$

Betrachtet man einen Zerfall von B^0 und $\bar{B}^0$ in einen CP-Eigenzustand X_{CP}, so wird eine CP-Verletzung durch Mischung der Zustände B^0 und $\bar{B}^0$ hervorgerufen. Die CP-Verletzung in den Zerfallskanälen $B^0 \to X_{CP}$ und $\bar{B}^0 \to X_{CP}$ entsteht durch Interferenz des direkten Zerfalls mit dem Zerfall mit vorhergehendem Übergang $B^0 \to \bar{B}^0 \to X_{CP}$ sowie $\bar{B}^0 \to B^0 \to X_{CP}$. Die entsprechenden Wahrscheinlichkeitsamplituden lassen sich in der in Tabelle 1.2 angegebenen Form schreiben. Hierbei sei δ die aus der komplexen CKM-Matrix folgende CP-verletzende Phase (Vorzeichenwechsel bei Übergang zum CP-konjugierten System) und α die CP-erhaltende Phase (Vorzeichen bleibt beim Übergang zum CP-konjugierten System erhalten). Die Amplituden von den CP-konjugierten Zuständen unterscheiden sich also gerade um den Faktor $e^{2\delta_{1,2}}$. Die Zerfallsraten Γ sind durch das Absolutquadrat der Wahrscheinlichkeitsamplituden gegeben:

$$\begin{aligned} \Gamma(B^0 \to X_{CP}) &= |a_1 + a_2|^2 \\ &= |a_1|^2 + |a_2|^2 + a_1^* a_2 + a_1 a_2^* \\ &= |a_1|^2 + |a_2|^2 + |a_1||a_2| e^{i(-\delta_1+\delta_2)} e^{i(-\alpha_1+\alpha_2)} e^{i(\delta_1-\delta_2)} e^{i(\alpha_1-\alpha_2)} \end{aligned} \tag{1.1}$$

	Kanal	Wahrscheinlichkeitsamplitude
Direkter	$B^0 \rightarrow X_{CP}$	$a_1 = \lvert a_1\rvert e^{i\delta_1} e^{i\alpha_1}$
Kanal	$\bar{B}^0 \rightarrow X_{CP}$	$\bar{a}_1 = \lvert a_1\rvert e^{-i\delta_1} e^{i\alpha_1}$
Interferenz-	$B^0 \rightarrow \bar{B}^0 \rightarrow X_{CP}$	$a_2 = \lvert a_2\rvert e^{i\delta_2} e^{i\alpha_2}$
Kanal	$\bar{B}^0 \rightarrow B^0 \rightarrow X_{CP}$	$a_2 = \lvert a_2\rvert e^{-i\delta_2} e^{i\alpha_2}$

Tabelle 1.1: Zerfallskanäle mit zugehörigen Wahrscheinlichkeitsamplituden, die zur CP-Verletzung im Zerfall von $B^0/\bar{B}^0$ in den CP-Eigenzustand X_{CP} führen.

und entsprechend

$$\begin{aligned} \Gamma(\bar{B}^0 \rightarrow X_{CP}) &= |\bar{a}_1 + \bar{a}_2|^2 \\ &= |\bar{a}_1|^2 + |\bar{a}_2|^2 + \bar{a}_1^*\bar{a}_2 + \bar{a}_1\bar{a}_2^* \\ &= |a_1|^2 + |a_2|^2 + |a_1||a_2| e^{i(\delta_1-\delta_2)} e^{i(-\alpha_1+\alpha_2)} e^{i(-\delta_1+\delta_2)} e^{i(\alpha_1-\alpha_2)} \end{aligned} \tag{1.2}$$

Mit diesen Beziehungen erhält man für die Asymmetrie A:

$$A = \frac{2|a_1||a_2|\sin(\delta_1-\delta_2)\sin(\alpha_1-\alpha_2)}{|a_1|^2 + |a_2|^2 + 2|a_1||a_2|\cos(\delta_1-\delta_2)\cos(\alpha_1-\alpha_2)} \tag{1.3}$$

Um eine CP-Verletzung aus der Asymmetrie A ableiten zu können, müssen sich sowohl der CP-erhaltende Anteil der Phase δ als auch der CP-erhaltende Phase α der beiden CP-konjugierten Zerfallskanäle unterscheiden.

1.3 Mischung im $B^0\bar{B}^0$-System

Die Flavour-Eigenzustände B^0 und $\bar{B}^0$ sind keine Eigenzustände des Hamiltonoperators der schwachen Wechselwirkung, da die schwache Wechselwirkung Übergänge zwischen B^0 und $\bar{B}^0$ gemäß Abb. 1.2 erlaubt.
Zur Beschreibung der $B^0\bar{B}^0$-Mischung werden die Flavour-Eigenzustände durch zweikomponentige Vektoren beschrieben:

$$\psi = \begin{pmatrix} |B^0\rangle \\ |\bar{B}^0\rangle \end{pmatrix}$$

Die zeitliche Entwicklung dieser Zustände ergibt sich aus folgender Schrödingergleichung:

$$i\frac{d}{dt}\psi = H\psi = \left(M - \frac{i}{2}\Gamma\right)\psi = \begin{pmatrix} M - \frac{i}{2}\Gamma & M_{12} - \frac{i}{2}\Gamma_{12} \\ M_{21} - \frac{i}{2}\Gamma_{21} & M - \frac{i}{2}\Gamma \end{pmatrix}\psi \tag{1.4}$$

Die Diagonalelemente des Hamiltonoperators H beschreiben die zeitliche Entwicklung der Zustände B^0 und $\bar{B}^0$ mit der Masse M und der Zerfallsbreite Γ. Aufgrund der CPT-Invarianz muß $H_{11} = H_{22}$ gelten. Die Übergänge zwischen den neutralen B-Mesonen gemäß Abb. 1.2 werden durch die Nichtdiagonalelemente des Hamiltonoperators ausgedrückt. Durch Diagonalisierung kann man aus den Flavour-Eigenzuständen $|B^0\rangle$ und $|\bar{B}^0\rangle$

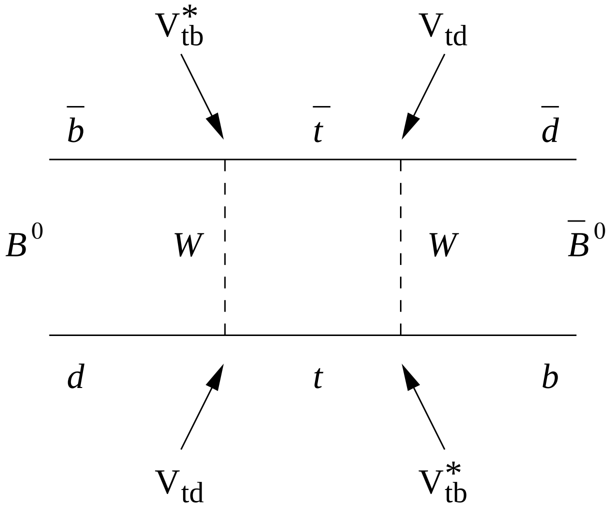

Abbildung 1.2: Feynman-Graph, der den Übergang $B^0 \to \bar{B}^0$ dominiert. Es sind die Elemente der CKM-Matrix vermerkt, die die Amplitude für den jeweiligen Vertex bestimmen.

Eigenzustände der schwachen Wechselwirkung $|B_H\rangle$ und $|B_L\rangle$ gewinnen[1]:

$$|B_H\rangle = \frac{1}{N}\Big((1+\epsilon)|B^0\rangle + (1-\epsilon)|\bar{B}^0\rangle\Big) \tag{1.5}$$

$$|B_L\rangle = \frac{1}{N}\Big((1+\epsilon)|B^0\rangle - (1-\epsilon)|\bar{B}^0\rangle\Big) \tag{1.6}$$

mit dem Normierungsfaktor N und dem Parameter ϵ:

$$N = \sqrt{|(1+\epsilon)^2 + (1-\epsilon)^2|}$$

$$\frac{1+\epsilon}{1-\epsilon} = \sqrt{\frac{M_{12} - \frac{i}{2}\Gamma_{12}}{M_{12}^* - \frac{i}{2}\Gamma_{12}^*}}$$

Die zeitliche Entwicklung von $|B_H\rangle$ und $|B_L\rangle$ ist somit gegeben durch

$$i\frac{d}{dt}|B_{H,L}\rangle = H|B_{H,L}\rangle = \lambda_{H,L}|B_{H,L}\rangle$$

mit den Eigenwerten $\lambda_{H,L} = M_{M,L} - \frac{i}{2}\Gamma_{H,L}$ des durch die Schrödingergleichung definierten Gleichungssystems. Die Massen- und Zerfallsbreitendifferenz sind gegeben durch:

$$\Delta M = M_H - M_L = 2\,\mathrm{Re}\left(\sqrt{\left(M_{12} - \frac{\Gamma_{12}}{2}\right)\left(M_{12}^* - \frac{\Gamma_{12}^*}{2}\right)}\right) \tag{1.7}$$

$$\Delta\Gamma = \Gamma_H - \Gamma_L = -4\,\mathrm{Im}\left(\sqrt{\left(M_{12} - \frac{\Gamma_{12}}{2}\right)\left(M_{12}^* - \frac{\Gamma_{12}^*}{2}\right)}\right) \tag{1.8}$$

[1]Die Indizes von $|B_H\rangle$ und $|B_L\rangle$ stehen für *heavy* (engl. für 'schwer') und *light* (engl. für 'leicht') und sind durch den geringen Massenunterschied von $|B_H\rangle$ und $|B_L\rangle$ motiviert.

Entsprechend kann die zeitliche Entwicklung der B^0-Mesonen durch die Eigenzustände der schwachen Wechselwirkung $|B_{H,L}\rangle$ beschrieben werden. Zum Zeitpunkt $t = 0$ gilt für einen reinen B^0-Zustand

$$\psi(0) = |B^0\rangle = \frac{N}{2(1+\epsilon)}\left(|B_H\rangle + |B_L\rangle\right)$$

und zur Zeit t

$$\psi(t) = \frac{N}{2(1+\epsilon)}\left(e^{-i\lambda_H t}|B_H\rangle + e^{-i\lambda_L t}|B_L\rangle\right)$$

Die an den Vertizes des Feynman-Graphen angegebenen Elemente der CKM-Matrix entsprechen dem Element $M_{12} - \frac{i}{2}\Gamma_{12}$ des Hamiltonoperators in Gleichung 1.4. Mögliche gemeinsame Zerfallskanäle von B^0 und $\bar{B}^0$ wie B^0, $\bar{B}^0 \to \pi^+\pi^-$ oder B^0, $\bar{B}^0 \to D^0\bar{D}^0$ sind Cabbibo-unterdrückt. Der Anteil Γ_{12} im Hamiltonoperator ist daher gegenüber M_{12} vernachlässigbar klein:

$$\Delta M \approx 2|M_{12}|, \;\; \Delta\Gamma \approx 0$$

Verwendet man noch die Gleichungen 1.5 und 1.6 sowie 1.7, so kann die zeitliche Entwicklung eines für den Zeitpunkt $t = 0$ reinen B^0-Zustandes durch reine B^0- und $\bar{B}^0$-Zustände formuliert werden (s. auch [kay91]):

$$|B^0(t)\rangle = e^{-i\left(M-\frac{i}{2}\Gamma\right)}(\cos\left(\frac{\Delta M}{2}t\right)|B^0\rangle + i\omega\sin\left(\frac{\Delta M}{2}t\right)|\bar{B}^0\rangle)$$

Hierbei ist der Mischungsfaktor $\omega = e^{-2i\delta^M_{\text{CKM}}}$ eingeführt worden:

$$\omega = \sqrt{\frac{M^*_{12}}{M_{12}}} = \frac{V_{td}V^*_{tb}}{V^*_{td}V_{tb}} = e^{-2i\delta^M_{\text{CKM}}}$$

Die verwendeten Proportionalitäten $M_{12} = (V^*_{td})^2(V_{tb})^2$ bzw. $M^*_{12} = (V_{td})^2(V^*_{tb})^2$ ergeben sich aus dem Feynman-Graphen, der in Abb. 1.2 angegeben ist, welcher den Übergang $B^0 \to \bar{B}^0$ darstellt.

1.4 Der 'goldene Zerfall'

Die durch das Unitaritätsdreieck definierte Größe $\sin 2\beta$ kann durch Beobachtung des sogenannten 'goldenen Zerfalls'

$$B_0/\bar{B}^0 \to J/\psi K^0_S \to l^+l^-\pi^+\pi^-$$

bestimmen werden. Hier soll kurz die durch die Mischung von B^0- und $\bar{B}^0$ hervorgerufene CP-Verletzung in diesem Zerfall betrachtet werden, welche dazu führt, daß der Kanal des direkten Zerfalls $B_0 \to J/\psi K^0_S$ mit dem Kanal $B_0 \to \bar{B}_0 \to J/\psi K^0_S$ interferiert. Sei a_1 die Amplitude für den direkten Zerfall und a_2 diejenige für den Zerfall nach dem Übergang $B^0 \to \bar{B}^0$ (s. Abb. 1.3). Für diese Amplituden gilt:

$$\begin{aligned} a_1 &\propto V^*_{cb}V_{cs} \\ a_2 &\propto \frac{V_{tb}V^*_{td}}{V^*_{tb}V_{td}}V_{cb}V^*_{cs} \propto \frac{e^{i\beta}}{e^{-i\beta}} = e^{i2\beta} \end{aligned}$$

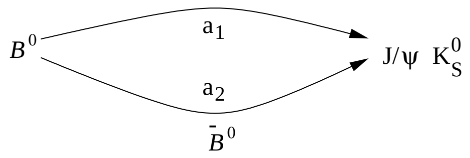

Abbildung 1.3: Die beiden Zerfallskanäle mit den zugehörigen Amplituden a_1 und a_2, die durch Interferenz zur CP-Verletzung durch Mischung beim 'goldenen Zerfall' führen.

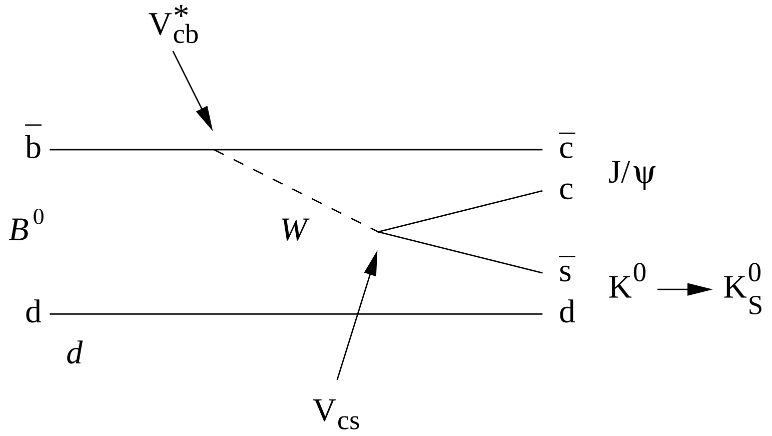

Abbildung 1.4: Feynman-Graph, der den Zerfall des B^0 in J/ψ und K_S^0 beschreibt. An den beiden Vertizes treten die Elemente V_{cb}^* und V_{cs} der CKM-Matrix auf, die die Amplitude des Zerfalls bestimmen.

Die Beiträge von $V_{cb}^* V_{cs}$ bzw. $V_{cb} V_{cs}^*$ können in diesem Zusammenhang vernachlässigt werden, da sie reell sind und daher keinen Einfluß auf die betrachtete CP-Verletzung haben. Die Proportionalitäten lassen sich aus den Feynman-Graphen gemäß Abb. 1.4 und Abb. 1.2 ablesen. Die Amplitude a_1 ergibt sich aus dem Graphen des Zerfalls $B_0 \to J/\psi K_S^0$, während für die Amplitude a_2 die Amplituden des im vorigen Abschnitt betrachteten Übergangs $B^0 \to \bar{B}^0$ mit der des Zerfalls $\bar{B}_0 \to J/\psi K_S^0$ multipliziert werden muß. Für den CP-konjugierten Zerfall gilt für die entsprechenden Amplituden $\bar{a}_1$ und $\bar{a}_2$

$$\begin{aligned}
\bar{a}_1 &\propto V_{cb} V_{cs}^* \\
\bar{a}_2 &\propto \frac{V_{tb}^* V_{td}}{V_{tb} V_{td}^*} V_{cb}^* V_{cs} \propto \frac{e^{-i\beta}}{e^{i\beta}} = e^{-i2\beta}
\end{aligned}$$

Unter Verwendung der Gleichungen 1.1 und 1.2 erhalten wir somit

$$\Gamma(B_0 \to J/\psi K_S^0) \propto e^{-\Gamma t}(1 + \sin 2\beta \sin \Delta m t)$$

$$\Gamma(\bar{B}_0 \to J/\psi K_S^0) \propto e^{-\Gamma t}(1 - \sin 2\beta \sin \Delta m t)$$

Für die Asymmetrie gemäß Gleichung 1.3 erhält man hiermit

$$\begin{aligned}
A(t) &= \frac{e^{-\Gamma t}(1 + \sin 2\beta \sin \Delta m t) - e^{-\Gamma t}(1 - \sin 2\beta \sin \Delta m t)}{e^{-\Gamma t}(1 + \sin 2\beta \sin \Delta m t) + e^{-\Gamma t}(1 - \sin 2\beta \sin \Delta m t)} \\
&= \sin 2\beta \sin \Delta m t
\end{aligned}$$

Letztendlich ergibt die Integration über die Zeit der Zusammenhang zwischen Asymmetrie A, der Größe $\sin 2\beta$ sowie der Massendifferenz ΔM und der B^0-Lebensdauer τ_{B^0}:

$$A = \sin 2\beta \frac{x}{1+x^2}$$

mit dem Mischungsparameter $x = \Delta M \tau_{B^0}$.

1.5 Signatur des 'goldenen Zerfalls'

Der 'goldene Zerfall' kann aufgrund spezieller Charakteristika klar vom Hintergrund getrennt werden. Ein Beispiel einer den 'goldenen Zerfall' enthaltenden Zerfallskette ist in Abb. 1.5 gegeben. Die klare Signatur des 'goldenen Zerfalls' ergibt sich aus folgenden

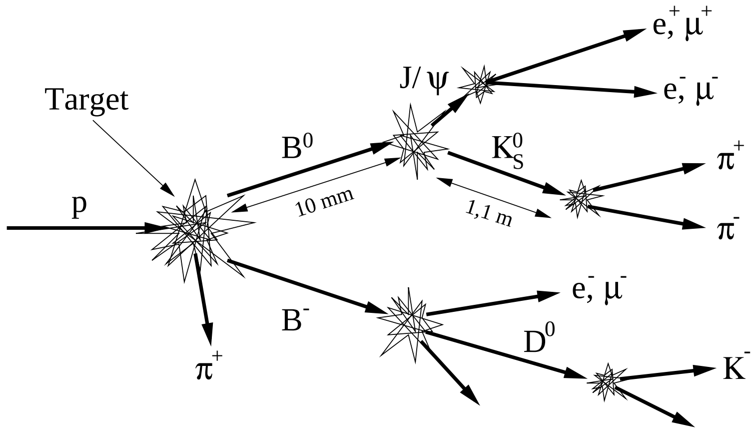

Abbildung 1.5: Zerfallskette mit dem 'goldenen Zerfall' (oberer Ast) und dem sogenannten Charge Tag (unterer Ast). Durch die Wechselwirkung von Protonenstrahl und Targetdraht entstehen unter anderem B-Mesonen. Im goldenen Zerfall geht ein B^0 nach einer typischen Weglänge von 10 mm in ein J/ψ und ein K_S^0 über. Das J/ψ zerfällt sofort in ein Leptonenpaar e^+e^-, $\mu^+\mu^-$ und das K_S^0 zerfällt nach einer typischen Wegstrecke von 1,1 m in $\pi^+\pi^-$. Im unterem Ast ist das zweite B-Meson zu sehen, welches zur Identifizierung des B^0- bzw. $\bar{B}^0$-Mesons herangezogen wird.

Umständen:

- Das Signal der von dem J/ψ-Zerfall herrührenden Leptonen.
- Die aus dem J/ψ-Zerfall stammenden Leptonen entstehen aufgrund der typischen Zerfallslänge des B-Mesons von 10 mm nicht am Hauptwechselwirkungspunkt, sondern an einem sekundären Vertex[2].
- Der Zerfall des K_S^0 in ein geladenes Pionenpaar nach einer typischerweise zurückgelegten Strecke vom sekundären Vertex von 1,1 m.
- Aufgrund ihrer Masse können die Pionen aus dem K_S^0-Zerfall klar von dem leptonischen Hintergrund getrennt werden.

[2] Als *sekundärer Vertex* wird ein Raumpunkt bezeichnet, an dem ein am primären Vertex, also am Hauptwechselwirkungspunkt, entstandenes Teilchen zerfällt.

Die Untersuchung der CP-Verletzung erfordert zusätzlich die Bestimmung der Flavourquantenzahl des im 'goldenen Zerfall' vorkommenden B-Mesons. Dieser Vorgang wird als *Tagging*[3] bezeichnet. Hierzu muß das zweite b-Quark identifiziert werden. Eine Möglichkeit hierzu, die als *Charge Tagging*[4] bezeichnet wird, nutzt den Umstand, daß das zweite b-Quark in ungefähr 40 % aller Ereignisse in einem geladenen B-Meson enthalten ist [sam98]. Die Ladung eines zweiten B-Mesons gibt daher Aufschluß darüber, ob ein B^0- oder $\bar{B}^0$-Meson im 'goldenen Zerfall' aufgetreten ist. Andere Wege zur Identifizierung der Flavourquantenzahl bieten sich in der Suche nach aus dem zweiten B-Meson hervorgegangenen Leptonen (Lepton-Tagging) oder Kaonen (Kaon-Tagging).
(s. Abb. 1.5).

1.6 Weitere physikalische Ziele des Experiments HERA-B

Neben der Untersuchung der Größe $\sin 2\beta$ können mit dem HERA-B Experiment folgende Bereiche untersucht werden:

- Die im $B_d^0/\bar{B}_d^0$-System auftretende Mischung tritt auch im $B_s^0/\bar{B}_s^0$-System auf. Das Verhältnis der Mischungsparameter x_d und x_s hängt dabei von dem Verhältnis der Elemente V_{td} und V_{ts} der CKM-Matrix ab:

$$\frac{x_s}{x_d} \propto \left|\frac{V_{ts}}{V_{td}}\right|^2$$

  Über das Verhältnis x_s/x_d kann so der Quotient V_{ts}/V_{td} bestimmt werden [loh94].

- Untersuchung der Kanäle $B^0 \to \pi^+\pi^-$ und $B_s \to K_S^0\rho^0$ zur Bestimmung der Winkel α und γ im Unitaritätsdreieck [alb95].

- Beschäftigung mit seltenen B-Zerfällen. Dies sind Zerfälle von B-Mesonen, bei denen die Flavourquantenzahl bei Übergängen des neutralen Stroms verändert wird. Da solche Zerfälle im Standardmodell verboten sind, bietet die Suche nach seltenen B-Zerfällen einen Test für den Gültigkeitsbereich es Standardmodells an [loh94].

[3] engl. für 'identifizieren'
[4] engl. für 'Identifikation durch die Ladung'

Kapitel 2

Das Experiment HERA-B

2.1 Der Aufbau des Detektors

Der HERA-B-Detektor besteht aus folgenden Komponenten (s. Abb. 2.1):

- Vertex-Tank mit den Targetdrähten und dem Silizium-Vertexdetektor zur Auflösung von sekundären Vertizes
- Ein Spektrometermagnet, der die Impulsbestimmung geladener Teilchen in den Spurkammern ermöglicht
- Neun Spurkammerlagen im Magnetfeld zur Impulsbestimmung sowie drei Spurkammerlagen mit sogenannten *Pad*-Kammern, die als Pretrigger für Hadronen mit hohem Transversalimpuls dienen.
- Vier Spurkammerlagen hinter dem Magneten
- Ein ringabbildender Čerenkov Detektor
- Zwei weitere Spurkammerlagen im feldfreien Raum
- Ein Übergangsstrahlungs-Detektor
- Ein elektromagnetisches Kalorimeter, das auch als Pretriggerquelle für Elektronkandidaten vorgesehen ist.
- Ein Myonsystem bestehend aus drei Absorbern und vier Spurkammerlagen, von denen zwei Lagen zusätzlich *Pad*-Kammern enthalten, welche Myonkandidaten als Pretriggerquelle zur Verfügung stellen.

Im folgenden soll ein kurzer Überblick der im HERA-B-Detektor (Abb. 2.1) vorgesehenen Subdetektoren gegeben werden. Die für den First Level Trigger relevanten Subdetektoren werden im nächsten Abschnitt behandelt.

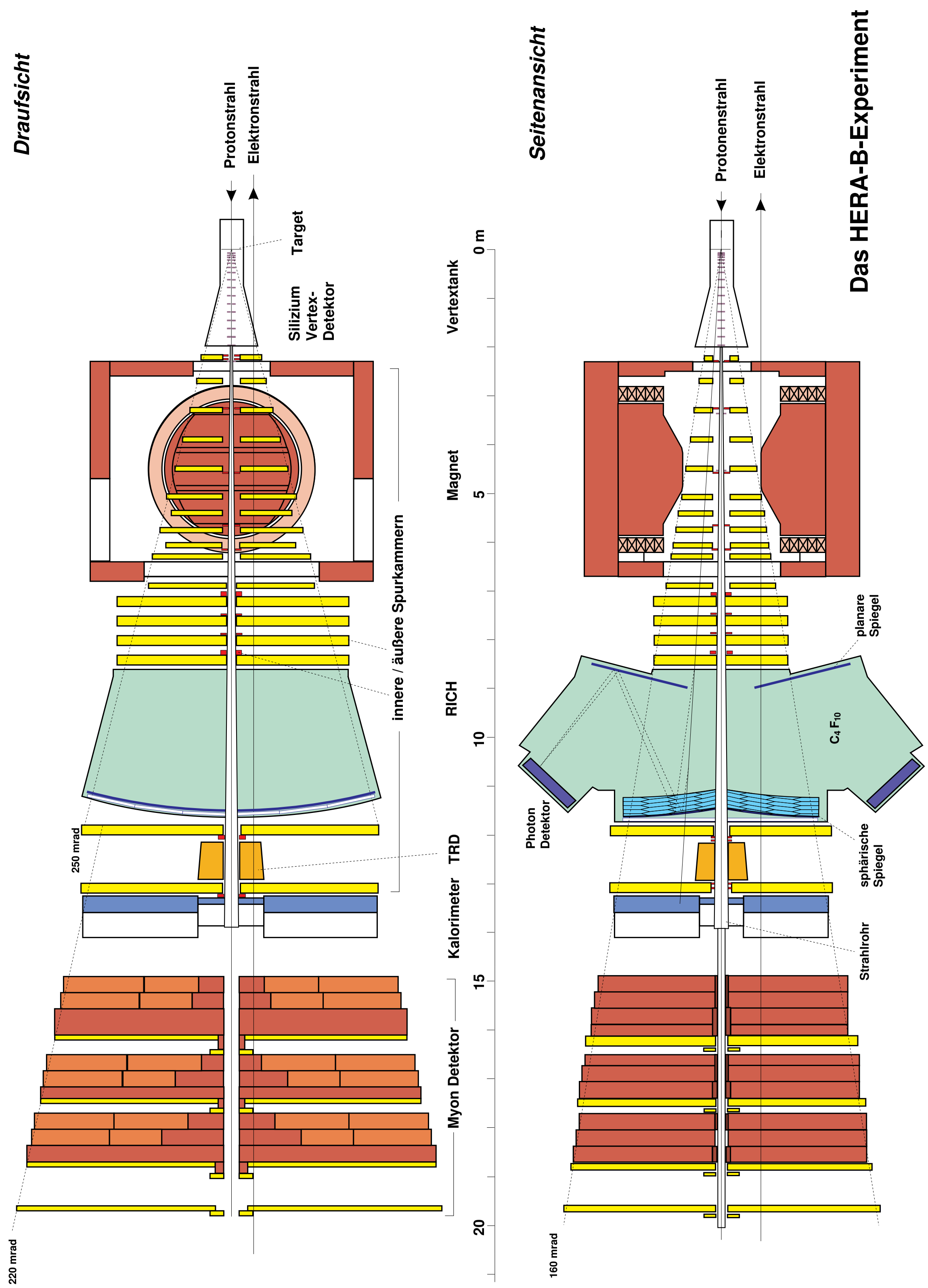

Abbildung 2.1: Schematische Ansicht des HERA-B Detektors

2.1.1 Das Target

HERA-B ist ein *Fixed Target*[1] Experiment, d.h. ein Teilchenstrahl — hier der 920 GeV Protonenstrahl von HERA — wechselwirkt mit einem unbewegten Zielmaterial, dem *Target*[2]. Als Target werden mehrere von insgesamt acht Drähten aus Kupfer, Aluminium, Titan und Kohlefasern mit einer Ausdehnung von 50 μm quer und 500 μm parallel zur Strahlrichtung in den Randbereich des Protonenstrahls eingefahren. Auf diese Weise soll nur der Halo des Protonenstrahls die Targetdrähte erreichen, der Kernbereich des Strahls bleibt unbeeinflußt (Abb. 2.2). Diese Technik gewährleistet eine nur geringe Beeinflussung des Betriebs der Experimente H1 und ZEUS, die ebenfalls den HERA-Protonenstrahl verwenden.

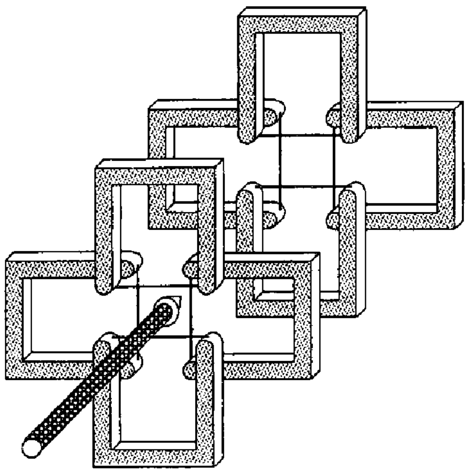

Abbildung 2.2: Aufbau des Halo-Targets [loh94]

Die Drähte können separat von oben, unten, links und rechts an den Protonenstrahl herangefahren werden. Die Protonen treffen auf die Nukleonen der Drähte und erzeugen dabei gemäß $pN \rightarrow B + X$ die gewünschten B-Mesonen. Die Protonenpakete treffen mit einem zeitlichen Abstand von 96 ns auf das Target. Um hierbei die angestrebte Wechselwirkungsrate von 40 MHz zu erreichen, ist vorgesehen, vier Drähte gleichzeitig zu verwenden. Die gleichzeitige Verwendung von mehreren Drähten bietet auch den Vorteil, bei der Analyse die Vertizes der auftretenden Wechselwirkungen klar unterscheiden zu können, da sich die Vertizes auf die vier Drähte verteilen. Um während der Datennahme die Wechselwirkungsrate bestimmen zu können und damit die Position der Target-Drähte zu justieren, werden Hodoskope verwendet. Dies sind Szintillationsdetektoren mit einer geringen geometrischen Akzeptanz, die im Bereich des RICH-Detektors installiert sind. In der Zukunft ist vorgesehen, die Targetrate auch mit der im inneren Bereich des elektromagnetischen Kalorimeters deponierten Energie zu bestimmen. Um die Beiträge der einzelnen Targetdrähte getrennt berücksichtigen zu können, sind weiterhin sogenannte *Charge-Integratoren*[3] vorgesehen. Sie messen die durch den Protonenstrahl an die Targetdrähte abgegebene Ladung und bestimmen so separat die Wechselwirkungsraten der Drähte [bak99].

[1] engl. für 'unbewegtes Ziel'
[2] engl. für 'Ziel', 'Zielmaterial'
[3] *charge*, engl. für 'Ladung', also 'Ladungs-Integratoren'

2.1.2 Der Vertexdetektor

Zur Auflösung des sekundären Vertex wird ein Detektor verwendet, der in der unmittelbaren Nähe des *Targets* aufgebaut ist. Dieser soll bei der vorgesehenen Analyse zur CP-Verletzung dazu verwendet werden, direkt am Hauptvertex erzeugte J/ψ von solchen zu unterscheiden, die aus einem Zerfall eines $B_0/\overline{B_0}$ hervorgegangen sind. Wegen der in diesem Bereich hohen Spurdichte wird ein Detektor benötigt, der eine besonders präzise Ortsauflösung gestattet. Diese Anforderungen werden von Silizium-Mikrostreifendetektoren erfüllt, die in acht Modulen zu je vier gleichartigen Superlagen installiert werden. Der Streifenabstand der doppelseitig ausgelesenen Siliziumdetektoren beträgt $54,6\,\mu$m auf der n-dotierten Seite und $51,7\,\mu$m auf der p-dotierten Seite. Die Streifen auf beiden Seiten sind gegeneinander um 90° gedreht um eine gleichmäßige Auflösung in der Ebene senkrecht zum Protonenstrahl zu erhalten. Zur Verbesserung der Auflösung wird auf der p-dotierten Seite zwischen je zwei Auslesestreifen ein weiter Streifen implantiert, der nicht direkt ausgelesen werden kann. Die Verbesserung der Auflösung resultiert aus einer kapazitiven Ladungsteilung, was sich in einer Verteilung der Ladung eines durchlaufenden Teilchens auf mehrere benachbarte Streifen auswirkt [rie98].

2.1.3 Der Magnet

Zur Bestimmung der Impulse geladener Teilchen ist ein Dipolmagnet vorgesehen, in dem sich neun Spurkammerlagen befinden. Es handelt sich hierbei um zehn normalleitende Spulen, die ein zylindersymmetrisches Magnetfeld vertikal zum Protonenstrahlrohr erzeugen. Die magnetische Flußdichte beträgt $0,8\,$T, das Wegintegral eines in der Nähe der Strahlachse laufenden Teilchens $2,2\,$Tm [alb95]. Da das Dipolfeld vertikal ausgerichtet ist, werden die geladenen Teilchen horizontal abgelenkt. Daher ist zum einen der Detektor in horizontaler Richtung weiter ausgedehnt als in vertikaler Richtung. Zum anderen ist bei den Spurkammern eine höhere örtliche Auflösung in horizontaler Richtung als in vertikaler vorgesehen.
Zur präzisen Bestimmung der Impulse werden die Teilchenspuren mittels der in dem Magneten enthaltenen Spurkammern verfolgt. Für den First Level Trigger wird dagegen nur die Teilchen in Spurkammern verfolgt, die hinter dem Magneten liegen und daher einen geraden Verlauf haben. Eine vorläufige Impulsinformation läßt sich dann aus dem horizontalen Abstand dieser in die Targetebene projizierten geraden Teilchenspur zum Target gewinnen [ful96].

2.1.4 Der Ring Imaging Čerenkov Detector

Der Ring Imaging Čerenkov Detector[4], im folgenden auch kurz *RICH* genannt, dient zur Trennung der beim Zerfall der B-Mesonen entstehenden Kaonen von den leichteren Pionen. Hierbei wird der Čerenkov-Effekt genutzt, bei dem geladene Teilchen Photonen emittieren, wenn sie mit einer Geschwindigkeit durch ein dichtes Medium laufen, die höher als die Phasengeschwindingkeit des elektromagnetischen Feldes in diesem Medium ist. Charakteristisch ist die Winkelverteilung der emittierten Photonen, die von geladenen Teilchen mit einer Geschwindigkeit β in einem Medium mit Brechungsindex n azimutal isotrop unter

[4]engl. für 'Ringabbildender Čerenkov Detektor'

einem Polarwinkel $\theta_{\check{C}}$ von

$$\cos\theta_{\check{C}} = \frac{1}{\beta n}$$

abgestrahlt werden. Über den Winkel läßt sich die Teilchengeschwindigkeit bestimmen. Die emittierten Photonen werden über sphärische Spiegel auf zwei Matrizen von Photomultiplieren[5] abgebildet. Ein geradlinig laufendes Teilchen erzeugt dabei einen Ring von Photonen auf den Photomultipliermatrizen. Aus den Radien der abgebildeten Ringe ergeben sich die Geschwindigkeiten der Teilchen, die sich durch den *RICH* bewegt haben. Als Medium wird C_4F_{10} verwendet, was eine Trennung von Pionen und Kaonen im Impulsbereich von $3 - 50$ GeV ermöglicht.

2.1.5 Der Übergangsstrahlungsdetektor

Der Übergangsstrahlungsdetektor oder kurz *TRD*[6] überlappt in der geometrischen Akzeptanz mit den inneren Teil des elektromagnetischen Kalorimeters. Die Aufgabe liegt in der Verbesserung der Separation von Elektronen und Hadronen. Übergangsstrahlung tritt auf, wenn relativistische, geladene Teilchen Grenzflächen zweier Medien mit unterschiedlicher Dieletrizität durchqueren. Ein Teilchen mit einem Dilatationsfaktor γ emittiert Übergangsstrahlung in einem Kegel mit einem Öffnungswinkel $\theta \sim 1/\gamma$. Die Intensität der Strahlung steigt mit γ an, wodurch die Separation von Teilchen mit ähnlichem Impuls aber unterschiedlicher Masse, wie z. B. Elektronen und Hadronen, ermöglicht wird. Der *TRD* besteht aus zwei Blöcken von jeweils 18 horizontal bzw. vertikal angeordneten Schichten, die alternierend aus einem Radiatormaterial und röhrenförmigen Proportionalzählern aufgebaut sind.

2.2 Die für den FLT relevanten Subdetektoren

In diesem Abschnitt sollen diejenigen Subdetektoren betrachtet werden, deren Informationen für den First Level Trigger[7], oder kurz FLT, verwendet werden. Da der First Level Trigger in der für ihn zur Verfügung stehenden Zeit nur eine vorläufige Analyse der Daten vornehmen kann, besitzt die Ausleseelektronik alle im folgenden betrachteten Subdetektoren einen speziellen Datenpfad, in dem eine schnelle Auslese mit reduziertem Datenvolumen ermöglicht wird.
Es gibt zwei Gruppen von für den First Level Trigger relevanten Subdetektoren. Zum einen sind Subdetektoren vorgesehen, die ein sogenanntes Pretriggersignal[8] generieren können, welches einem Spurkandidaten entspricht. Es sind vier verschiedene Arten von Pretriggersignalen vorgesehen:

- Kandidaten für Elektronen im elektromagnetischen Kalorimeter
- Kandidaten für Photonen im elektromagnetischen Kalorimeter. Ein Photonkandidat wird durch das Überschreiten einer programmierbaren Energieschwelle und einer fehlenden Koinzidenz in der Spurkammerlage unmittelbar vor dem Kalorimeter definiert.

[5] engl. für Sekundärelektronenvervielfacher
[6] **T**ransition **R**adiation **D**etector, engl. für 'Übergangsstrahlungsdetektor'
[7] engl. für 'Erste Auslösestufe'
[8] engl. für 'Vor-Auslösestufe'

- Das Myonsystem ermittelt die Myon-Spurkandidaten

- Die High-p_T-Kammern im Magnetfeld ermitteln Hadronkandidaten mit hohem Transversalimpuls

Die andere Gruppe von Subdetektoren, die inneren und äußeren Spurkammern, ist für die Spursuche vorgesehen.

2.2.1 Das elektromagnetische Kalorimeter

Das elektromagnetische Kalorimeter, im folgenden auch kurz *ECAL*[9] genannt, besitzt folgende Aufgaben [loh94]:

- Bereitstellung eines Pretriggersignals für den First Level Trigger in Form von Elektron-Spurkandidaten und Photonkandidaten.

- Separation von Elektronen aus dem Zerfall $J/\psi \to e^+e^-$ von geladenen Pionen. Der hierbei angestrebte Unterdrückungsfaktor beträgt 100 und wird erreicht, indem die Teilchenenergie im Kalorimeter mit dem im Spurkammersystem bestimmten Impuls verglichen wird. Gemäß dem Proposal [loh94] beträgt die geometrische Akzeptanz für den Nachweis beider Elektronen aus dem Zerfall eines J/ψ 70%, wobei der Akzeptanzverlust im wesentlichen auf den unbestückten Bereich des Kalorimeters bei der Durchführung des Protonenstrahlrohrs zurückzuführen ist.

- Nachweis von π^0 über den Zerfall $\pi^0 \to \gamma\gamma$ im Energiebereich von $5 - 200\,\text{GeV}$. Dieser Kanal wird auch zur Kalibration des Kalorimeters verwendet, da der Zerfall $\pi^0 \to \gamma\gamma$ ein sehr klares Signal liefert.

- Nachweis einzelner Elektronen aus semileptonischen Zerfällen von B-Mesonen, die beim *tagging*[10] verwendet werden. Die geometrische Akzeptanz liegt hier bei 83% [loh94].

Die einzelnen Kalorimeterzellen werden in Schaschlik-Bauweise gefertigt. Eine Schaschlik-Kalorimeterzelle besteht alternierend aus Schichten von Absorber- und Szintillatormaterial. Um die im Szintillator entstehenden Photonen zu detektieren, sind senkrecht zu den Schichten Wellenlängenschieber eingebracht (Abb. 2.3), die das Szintillationslicht in den sichtbaren Wellenlängenbereich umsetzen und zu den Photomultipieren leiten. Da die Spurdichte stark mit dem Abstand vom Protonenstrahlrohr abnimmt, sind drei verschiedene Granularitätsbereiche des Kalorimeters vorgesehen (Abb. 2.3). Im inneren Bereich werden $2{,}2 \cdot 2{,}2\,\text{cm}^2$ große Zellen mit Wolfram als Absorbermaterial verwendet. Im mittleren und äußeren Bereich hingegen befinden sich $6{,}0 \cdot 6{,}0\,\text{cm}^2$ bzw. $11{,}0 \cdot 11{,}0\,\text{cm}^2$ große Zellen mit Blei als Absorbermaterial.

[9] **E**lectromagnetic **Cal**orimeter, engl. für 'elektromagnetisches Kalorimeter'
[10] engl. für 'Identifizierung' oder 'Markierung'. Als tagging wird der Analyseschritt bezeichnet, in dem Reaktionsprodukte gesucht werden, die den gesuchten Zerfallskanal eindeutig identifizieren.

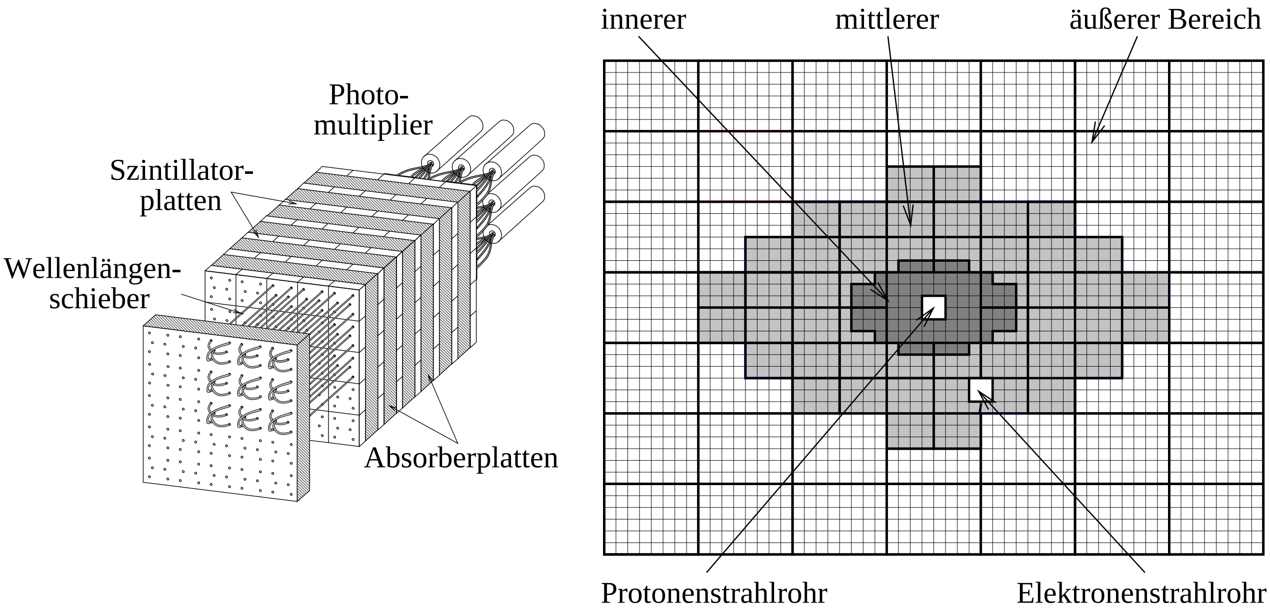

Abbildung 2.3: Die linke Abbildung zeigt die schematische Ansicht eines Schaschlik-Moduls aus dem inneren Bereich des Kalorimeters. Die rechte Abbildung verdeutlicht die Aufteilung des Kalorimeters in den inneren, mittleren und äußeren Bereich. Die Granularität nimmt von außen nach innen zu, um die zum Protonenstrahlrohr hin stark ansteigende Teilchendichte zu berücksichtigen. Die Belegungsdichte der Kalorimeterzellen steigt so nicht über 10 % [loh94].

2.2.2 Das Myonsystem

Die Aufgaben des Myonsystems sind analog zu denen des elektromagnetischen Kalorimeters:

- Bereitstellung eines Pretriggersignals für den First Level Trigger in Form von Myon-Spurkandidaten.

- Nachweis und Separation von Myonen aus dem Zerfall $J/\psi \rightarrow \mu^+\mu^-$ in einem Energiebereich von einigen GeV bis zu 200 GeV. Hadronen sollen hierbei um einen Faktor von 100 unterdrückt werden. Die geometrische Akzeptanz beträgt für beide Myonen des betrachteten Zerfalls 71,5% [loh94].

- Nachweis einzelner Myonen, die zum *tagging* von B-Mesonen dienen mit einer geometrischen Akzeptanz von 81% [loh94].

Da Myonen im Gegensatz zu den anderen im HERA-B Detektor nachgewiesenen Teilchensorten nur sehr wenig von Materie absorbiert werden, kann man ein Myonnachweissystem erhalten, indem man Spurkammern hinter einer Folge von massiven Absorberschichten aufstellt. Das HERA-B Myonsystem besteht aus drei Absorbern sowie vier Superlagen von Spurkammern, wobei sich zwei Spurkammern hinter dem letzten Absorber befinden (s. Abb. 2.4). Die ersten beiden Superlagen bestehen aus Einzellagen in drei Stereowinkeln (0°, ±20° um die y-Achse). Die Spurkammern der dritten und vierten Superlage bestehen hingegen nur aus einer 0°-Lage mit Platten (*Pads*) als Kathoden und dazu gegenüberliegenden Drähten als Anoden (Abb. 2.5).

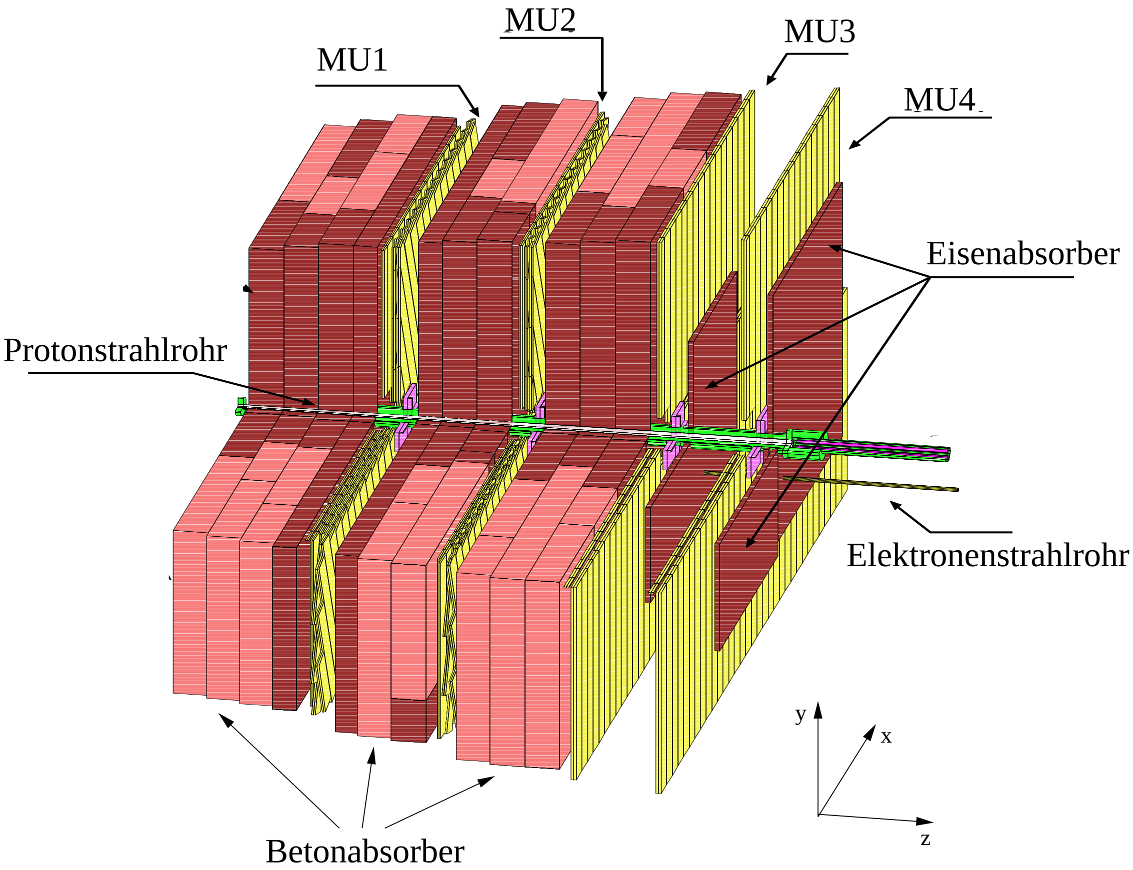

Abbildung 2.4: Perspektivische Ansicht des Myonsystems. Es sind die vier Spurkammerlagen MU1 – 4 und die drei Betonabsorber zu sehen.

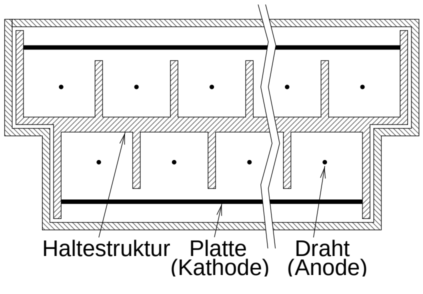

Abbildung 2.5: Schnitt durch eine Myon-*Pad*-Kammer [loh94]

Die Signale sowohl von den *Pads* als auch von den Drähten werden separat ausgelesen. Koinzidenzen in hintereinander liegenden *Pads* der dritten und vierten Superlagen dienen als Myon-Spurkandidaten und werden in Form von Pretriggersignale an den First Level Trigger gesandt. Die Signale der Anodendrähte geben eine genauere Information über die Lage der Spuren und werden bei der Spurensuche des First Level Trigger und in der Analyse verwendet.

2.2.3 Die High-p_T-Kammern

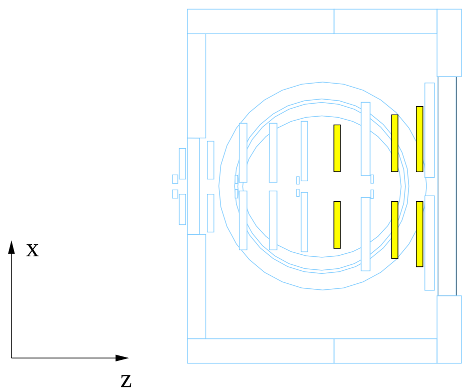

Abbildung 2.6: Position der drei Lagen von High-p_T-Kammern im Bereich des Spektrometermagneten. Sie dienen als Pretrigger von Hadronen mit hohem Transversalimpuls.

Um im First Level Trigger auch auf hadronische B-Zerfälle reagieren zu können, sind innerhalb des Spektrometermagneten drei Lagen von Spurkammern vorgesehen, die ein Pretriggersignal an den FLT senden können (s. Abb. 2.6). So kann z. B. auf den Zerfallskanal $B^0 \to \pi^+\pi^-$ getriggert werden, welcher die Bestimmung des Winkels α im Unitaritätsdreieck erlaubt [alb95]. Da es sich bei den zerfallenden B-Mesonen um relativ schwere Teilchen handelt, besitzen die Zerfallsprodukte einen großen Transversalimpuls, der sich wiederum in einem großen Streuwinkel äußert. Daher sollen zwei Spuren mit großem Transversalimpuls ($p_T > 1{,}5\,\mathrm{GeV}$) und einer invarianten Masse von mehr als $4{,}5\,\mathrm{GeV}$ als Triggerbedingung dienen. Die Triggereffizienz für den Zerfall $B^0 \to \pi^+\pi^-$ wurde zu 35% ermittelt [alb95].

2.2.4 Das Äußere Spurkammersystem

Die Spurdichte von geladenen Teilchen nimmt mit dem Abstand vom Protonenstrahlrohr stark ab. Um diesem Umstand Rechnung zu tragen sind zwei verschiedene Spurkammersysteme vorgesehen. Für das innere Spurkammersystem, welches den geometrischen Bereich mit der höchsten Spurdichte in unmittelbarer Nähe des Strahlrohrs abdeckt, ist eine spezielle Form von Mikrostreifen-Gasdetektoren vorgesehen (siehe nächster Abschnitt). Daneben sind 15 Superlagen von äußeren Spurkammern vorgesehen, die aus jeweils sechs Einzellagen von hexagonalen Driftröhren bestehen. Die sechs Einzellagen setzen sich aus drei Stereolagen mit den Winkeln 0° und ±5° um vertikale Achse zusammen, die zur Verbesserung der Auflösung jeweils doppelt um einen halben Drahtabstand verschoben vorhanden sind.

Man kann drei Bereiche des Spurkammersystems mit folgenden Funktionalitäten unterscheiden (s. Abb. 2.8):

- Neun Superlagen (MC[11] 1-9) innerhalb des Magneten zur Rekonstruktion der Spuren in dem Magnetfeld, die zur Impulsbestimmung geladener Teilchen dienen.
- Vier Superlagen (PC[12] 1-4) im feldfreien Raum zwischen dem Magneten und dem RICH. Die erste und vierte dieser Lagen wird auch für den First Level Trigger verwendet.
- Zwei Superlagen (TC[13] 1 u. 2) zwischen dem RICH und dem Kalorimeter zur Extrapolation der im Myonsystem und im Kalorimeter gefundenen Spurkandidaten. Diese beiden Lagen werden ebenfalls für den First Level Trigger verwendet.

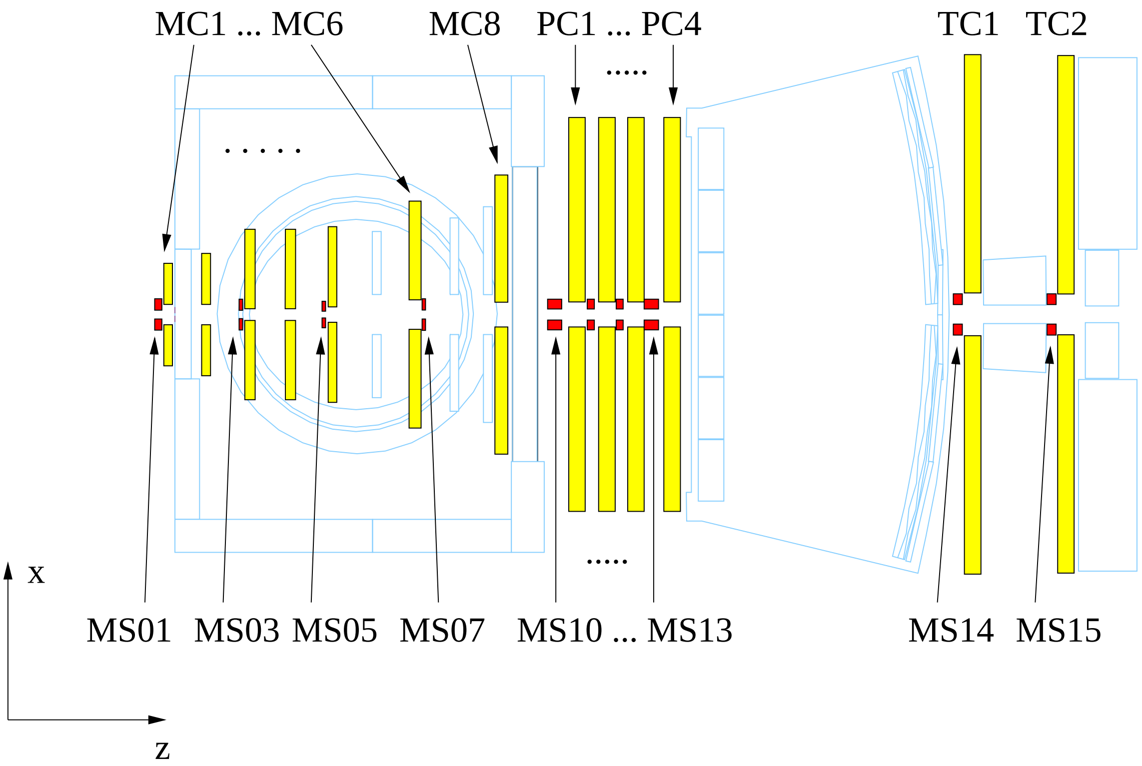

Abbildung 2.7: Superlagen der äußeren und inneren Spurkammern. Es sind die drei Gruppen der Spurkammern zu erkennen: MC1-8 und MS01-MC07 im Bereich des Spektrometermagneten, PC1-4 und MS10-13 zwischen Magnet und RICH sowie TC1 und TC2 sowie MS14 und MS15 zwischen RICH und ECAL.

Die Superlagen PC 1, PC 4, TC 1 und TC 2 (siehe Abbildung 2.7) werden für die Spursuche des First Level Trigger verwendet. Dazu ist zum einen ein spezieller Datenpfad vorgesehen, über den die Treffer-Information der Spurkammern ohne Berücksichtigung der Driftzeiten

[11] **M**agnet **C**hamber, engl. für 'Magnetkammern'
[12] **P**article **C**hamber, engl. für 'Teilchenkammern'
[13] **T**rack **C**hamber, engl. für 'Spurkammern'

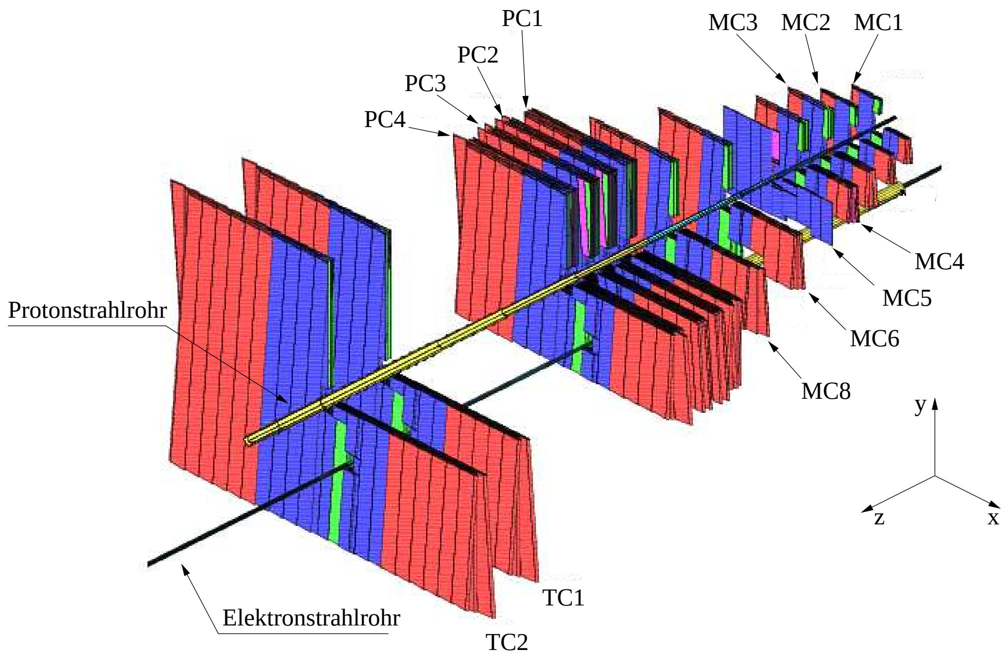

Abbildung 2.8: Perspektivische Ansicht der Superlagen der äußeren Spurkammern [lan99]. Diese Abbildung zeigt den Aufbau der einzelnen Superlagen, wobei der Übersicht wegen ein Quadrant jeder Superlage weggelassen ist. Jede Superlage besteht aus drei bzw. sechs Lagen in drei Stereowinkeln, die eine Messung der Spuren in vertikaler Richtung ermöglichen.

direkt zu den *Track Finding Units*[14] (s. nächstes Kapitel) übermittelt wird. Um die Treffer-Effizienz in diesen Superlagen und damit die Effizienz des First Level Triggers zu erhöhen, sind in diesen Lagen Stereolagen mit doppelten Spurkammern vorgesehen, deren Treffer-Information mit einer logischen Oder-Verknüpfung kombiniert werden.

2.2.5 Das Innere Spurkammersystem

Die inneren Spurkammern ergänzen die geometrische Akzeptanz der äußeren Spurkammern von einem Winkelbereich von minimal 10 mrad bis zu einem Abstand von $20-25$ cm vom Protonenstrahlrohr. Dieser geometrische Bereich ist von gleicher Bedeutung wie der von den äußeren Spurkammern erfaßte Bereich. Es liegen ungefähr 50% der gesuchten Teilchenspuren in diesem Winkelbereich [buc99]. Für zehn der 15 Superlagen der äußeren Spurkammern ist eine entsprechende Superlage von inneren Spurkammern vorgesehen, siehe Abbildung 2.7. Da in der Nähe des Strahlrohrs die Teilchendichte besonders hoch ist, wer-

[14] engl. für 'Spursuch-Einheit'

den die ersten drei Superlagen in der Nähe des Vertex aus Silizium-Mikrostreifendetektoren gefertigt, wie sie auch im Vertexdetektor verwendet werden. Für die übrigen Superlagen werden eine spezialisierte Form von $25 \times 25\,\mathrm{cm}^2$ großen MSGCs[15] eingesetzt. Die MSGC-Superlagen sind in drei Stereolagen ($0°$ und $\pm 5°$) um die y-Achse angeordnet. Eine Einzellage besteht aus vier L-förmigen voneinander unabhängigen Kammern, wobei zwölf Kammern zu einer der zehn Superlagen zusammengefaßt werden.

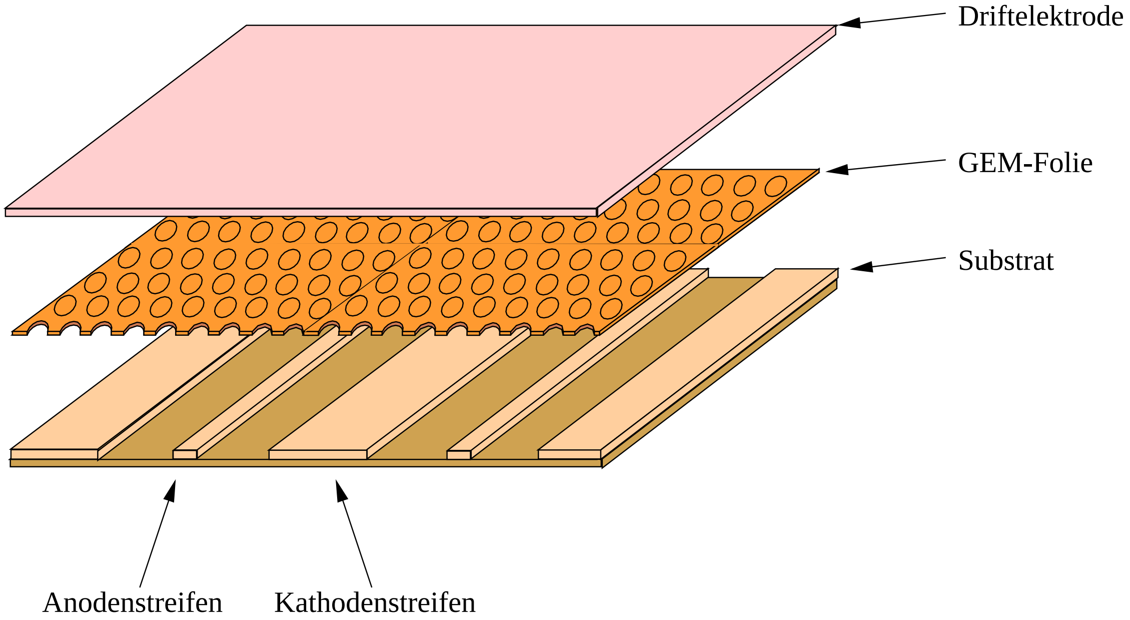

Abbildung 2.9: Prinzipieller Aufbau einer GEM-MSGC. Die Gasverstärkung erfolgt in zwei Stufen. Einlaufende Teilchen führen zu einer Ionisation von Gasmolekülen zwischen Driftelektrode und GEM-Folie. Eine Gasverstärkung der Primärionen erfolgt an den Löchern der GEM-Folie und an den Anodenstreifen am Substrat.

Herkömmliche MSGCs sind in der hadronischen Umgebung von HERA-B nicht geeignet, da bei der erforderlichen hohen Gasverstärkung Überschläge zwischen Anoden- und Kathodenstreifen zu einem Ausfall der Kammern führen. Daher wird bei HERA-B eine Weiterentwicklung der herkömmlichen MSGC, die GEM-MSGC, verwendet, bei der eine zweistufige Gasverstärkung vorgesehen ist. Dazu wird eine sogenannte GEM-Folie[16] zwischen der Driftelektrode und dem Substrat, welches die Anoden- und Kathodenstreifen fixiert, eingebracht. Die GEM-Folie ist beidseitig mit Kupfer bedampft und mit Löchern von einem Durchmesser von $50\,\mu$m bis $150\,\mu$m versehen, siehe Abbildung 2.9. Durch das Anlegen einer Hochspannung an beiden Seiten der Folie wird ein hoher Feldgradient an den Löchern der Folie hervorgerufen, der dort zu einer ersten Gasverstärkung führt. Die zweite Gasverstärkung findet an den Anodenstreifen auf dem Substrat statt, wo durch eine geringe Breite der Streifen ein hoher Feldgradient erzeugt wird. Durch die zweistufige Gasverstärkung kann die Feldstärke an den Anodenstreifen soweit reduziert werden, daß ein sicherer Betrieb ohne Funkenüberschläge ermöglicht wird [hil99].
Wie bei den äußeren Spurkammern liefert die Front End Elektronik der inneren Spur-

[15] **M**icro **S**trip **G**as **C**hamber, engl. für 'Mikrostreifen-Gasdetektor'
[16] **G**as **E**lectron **M**ultiplier, engl. für 'Gas Elektron Vervielfacher'.

kammern Trefferinformationen direkt an den First Level Trigger. Weil hier nicht die volle Präzision notwendig ist, werden jeweils vier nebeneinander liegende Streifen der MSGCs verodert, d.h., sie werden logisch als ein breiter Streifen betrachtet. Um die Effizienz des First Level Triggers zu erhöhen, sind zusätzlich in den Superlagen, die für den First Level Trigger verwendet werden, jeder der drei Stereowinkel doppelt besetzt.

Kapitel 3

Die Erste Triggerstufe

3.1 Die vier Triggerstufen

Das Ziel des HERA-B Experiments ist die Untersuchung der CP-Verletzung im $B_0\bar{B}_0$-System. Dazu sollen pro Jahr ungefähr 1000 Ereignisse des Zerfallskanals $B_0/\bar{B}^0 \rightarrow J/\psi K_S^0 \rightarrow l^+l^-\pi^+\pi^-$ gefunden werden, um eine Meßgenauigkeit von $\sigma(\sin(2\beta)) \approx 0{,}1$ zu erreichen [loh94]. Auf Grund der klaren Signatur dieses Kanals wird er auch als 'Golden Decay'[1] bezeichnet (siehe hierzu Abschnitt 1.5). Diese Zahl ergibt sich aus einer vorgesehenen Jahresbetriebszeit von 10^7 s bei einer mittleren Wechselwirkungsrate von 40 MHz und den in Tabelle 3.1 angegebenen Ereigniswahrscheinlichkeiten und Verzweigungsverhältnisse sowie Effizienzen.

$\sigma_{b\bar{b}}/\sigma_{inel.}$	$\approx 10^{-6}$
$P(b \rightarrow B^0) + P(b \rightarrow \overline{B^0})$	$\approx 0,8$
$\mathrm{BR}(B_0/\overline{B_0} \rightarrow J/\psi K_s^0)$	$\approx 5 \cdot 10^{-4}$
$\mathrm{BR}(J/\psi K_s^0 \rightarrow l^+l^-)$	$\approx 0,06$
$\mathrm{BR}(K_s^0 \rightarrow \pi^+\pi^-)$	$\approx 0,69$
Trigger- und Rekonstruktionseffizienz	0,15
Kaon-*Tag*-Effizienz	0,45
Lepton-*Tag*-Effizienz	0,15

Tabelle 3.1: Übersicht der Wirkungsquerschnitte, Verzweigungsverhältnisse und Effizienzen, die zur Abschätzung der erwarteten Produktionsrate des 'goldenen Zerfalls' verwendet werden [loh94]. Hier wird die Abkürzung 'P' und 'BR' des englischen 'probability' und '**b**ranching **r**atio' zur Bezeichnung der Ereigniswahrscheinlichkeit und Verzweigungsverhältnisse verwendet. Die *Tag*-Effizienzen geben die Wahrscheinlichkeit dafür an, daß die Flavour- Quantenzahl des im 'goldenen Zerfall' aufgetretenen B-Mesons korrekt bestimmt worden ist.

Zur Produktion der $B_0/\overline{B_0}$-Mesonen, wird der HERA-Protonenstrahl verwendet, welcher

[1] engl. für 'Goldener Zerfall'

eine Energie von 920 GeV besitzt. Der zeitliche Abstand zweier Protonenpakete im HERA-Protonenstrahl beträgt 96 ns. Da aus technischen Gründen nur 180 der möglichen 220 Pakete im HERA-Protonenstrahl besetzt sind, ergibt sich eine effektive Ereignisfrequenz von

$$\frac{1}{96\,\mathrm{ns}} \cdot \frac{180}{220} = 8{,}5\,\mathrm{MHz}$$

Um die gewünschte Wechselwirkungsrate von 40 MHz zu erreichen, sind im Mittel

$$\frac{40\,\mathrm{MHz}}{8{,}5\,\mathrm{MHz}} = 4{,}7$$

Wechselwirkungen pro Protonenpaket erforderlich. Aus Tabelle 3.1 geht hervor, daß der gesuchte Zerfall $B_0/\bar{B}^0 \rightarrow J/\psi K_S^0 \rightarrow l^+l^-\pi^+\pi^-$ durch einen starken Untergrund überlagert ist. Um die gesuchten Ereignisse von dem Untergrund trennen zu können, ist ein vierstufiges Triggersystem vorgesehen. Die (im Datenpfad) ersten Triggerstufen bearbeiten die eintreffenden Ereignisse schneller aber können wegen der kurzen Verarbeitungszeit dabei nicht die vollständige Detektorinformation berücksichtigen. Die im Datenpfad späteren Triggerstufen gestatten eine genauere Analyse bei einer längeren Verarbeitungszeit bei Verwendung der gesamten Ereignisdaten. Die Verarbeitungszeit der Triggerstufen wird auch als *Latenzzeit* bezeichnet, welche als das Zeitintervall vom Eintreffen der ersten zu einem Ereignis gehörenden Information bis zur Ausgabe der Triggerentscheidung definiert ist. Tabelle 3.2 gibt eine Übersicht der Eingangsraten und Latenzzeiten der vier Triggerstufen. Die in der Tabelle verwendeten Abkürzungen FLT, SLT, TLT und 4LT bedeuten First Level Trigger, Second Level Trigger, Third Level Trigger und Fourth Level Trigger.[2]

Triggerstufe	Eingangsrate	Unterdrückungsfaktor	mittlere Latenzzeit
FLT	≈ 10 MHz	200	$< 12{,}3\,\mu$s
SLT	≈ 50 kHz	200	≈ 3 ms
TLT	≈ 250 Hz	5	≈ 200 ms
4LT	≈ 50 Hz	5	≈ 4 s

Tabelle 3.2: Eingangsraten, Unterdrückungsfaktoren und Latenzzeiten der vier Triggerstufen.

Es folgt ein kurzer Überblick der vier Triggerstufen.

3.1.1 Die Erste Triggerstufe

Bereits die erste Triggerstufe nimmt eine vorläufige Spurrekonstruktion vor. Das elektromagnetische Kalorimeter und das Myonensystem sowie die High-p_T-Kammern stellen Spurkandidaten zur Verfügung, die dann in ausgewählten Lagen der äußeren und inneren Spurkammern sowie den Spurkammern des Myonsystems im Bereich hinter dem Spektrometermagneten verfolgt werden. Aus den Spurparametern werden dann die Impulse der Spurkandidaten berechnet und die Triggerentscheidung aufgrund der Berechnung invarianter Massen oder bei dem Überschreiten einer festgelegten Anzahl Spurkandidaten gefällt.

[2] engl. für 'Erste, Zweite, Dritte und Vierte Auslösestufe.

Ziel dieser Strategie ist es, bereits in der ersten Triggerstufe Kandidaten für den Zerfall $J/\psi \to l^+l^-$ selektieren zu können. Da die zweite Triggerstufe im Mittel nicht mehr als 50kHz an Eingangsdaten verarbeiten kann und die Front End Driver der Subdetektoren nicht schneller ausgelesen werden können, darf die Ausgangsrate des First Level Trigger im Mittel 50kHz nicht überschreiten.

Der Aufbau des First Level Trigger wird später in diesem Kapitel ausführlich dargestellt.

3.1.2 Die Zweite Triggerstufe

In der zweiten Triggerstufe werden die Spurparameter der in der ersten Triggerstufe gefundenen Spurkandidaten präzisiert. Sie verwendet hierzu zusätzlich den vom FLT gelieferten Informationen auch die Driftzeiten in den äußeren Spurkammern und berücksichtigt alle Auslesestreifen im inneren Spurkammersystem. Weiterhin werden die Spurkandidaten durch den Bereich des Spektrometermagneten in den Vertexdetektor projiziert. Hierzu wird eine spezielle Tabelle verwendet, in der Spurparameter von Spurkandidaten hinter dem Feld des Spektrometermagneten auf Spurparameter innerhalb des Vertexdetektors abgebildet werden.[3] So können ohne Inkaufnahme einer vergleichsweise rechenintensiven Spurverfolgung durch den Magneten, die Spuren im Vertexdetektor bis zum Wechselwirkungspunkt verfolgt werden. Dies erlaubt eine Unterdrückung des von direkt am Vertex erzeugten J/ψ herrührenden Hintergrunds, der das Signal des 'Goldenen Zerfalls' um drei Größenordnungen übersteigt. Dazu wird verlangt, daß die in der ersten Triggerstufe gefundenen Zerfallsprodukte des J/ψ von einem sekundären Vertex stammen, der auf einen B-Zerfall hinweist. Am Ausgang der zweiten Triggerstufe soll die Ereignisrate im Mittel auf 250 Hz reduziert worden sein.

Die zweite Triggerstufe wird als Softwaretrigger implementiert, d.h. daß nicht eine speziell für diesen Zweck entwickelte Elektronik vorgesehen ist. Vielmehr werden die entsprechenden Analyseprogramme auf einer großen Anzahl von handelsüblichen PCs (einer sogenannten Farm)abgearbeitet, was eine hohe Flexibilität bei der Auswahl des Triggeralgorithmus erlaubt. Da allerdings die Bandbreite der vom Detektor gelieferten Daten auf dieser Stufe noch sehr hoch ist, können die Daten nicht über ein gebräuchliches Netzwerk übermittelt werden. Statt dessen erlaubt es eine spezielle Technik, alle Detektordaten zu den Ereignissen zu speichern, bei denen der First Level Trigger angesprochen hat. Dieser als *Second Level Buffer*[4] oder kurz SLB bezeichnete Speicher wird mit den Informationen der *Front End Driver*[5] der einzelnen Subdetektoren gefüllt. Im Second Level Buffer werden die Daten gespeichert, bis der Second Level Trigger seine Entscheidung gefällt hat. Während dieser Zeit stellt der Second Level Buffer seine Informationen den PCs des Second Level Trigger über einen *Switch*[6] zur Verfügung.

[3] Ein ähnliches Verfahren wird auch bereits auf der Stufe des First Level Trigger zu Berechnung der kinematischen Parameter der Spurkandidaten verwendet, siehe Abschnitt. 3.6.

[4] engl. für 'Speicher der zweiten Stufe'

[5] engl. für 'Vordere-Ende-Elektronik', mit Front End Elektronik oder Front End Driver werden die Komponenten der Auswertelektronik bezeichnet, die direkt mit dem zum zugehörigen Subdetektor in Verbindung stehen und auch meist örtlich in unmittelbarer Nähe untergebracht sind.

[6] engl. für 'Schalter', gemeint ist hier ein Netzwerk, über das der PC der Farm auf alle Detektorinformationen zugreifen kann.

3.1.3 Die Dritte Triggerstufe

Die Hintergrundunterdrückung der ersten beiden Triggerstufen ist bei Anwendung der eben beschriebenen Schnitte auf ein J/ψ von einem sekundären Vertex ausreichend, um die Ereignisse aufzuzeichnen, die akzeptiert werden. Sollen allerdings weitere Kanäle bei der Datennahme berücksichtigt werden, ist eine weitere Reduktion erforderlich.
In der dritten Triggerstufe können erstmalig alle zu einem Ereignis gehörigen Meßdaten berücksichtigt und so das gesamte Ereignis rekonstruiert werden. Dies gilt insbesondere für die Daten des RICH-Detektors, dessen Rekonstruktionsalgorithmen sehr rechenintensiv sind, da die durch den Čerenkov-Effekt hervorgerufenen Ringe rekonstruiert werden müssen. Es werden alle im Ereignis aufgezeichneten Spuren rekonstruiert werden, während im Second Level Trigger lediglich die Spuren, die vom First Level Trigger gefunden worden sind, präzisiert und durch weitere Spurkammerlagen verfolgt werden. Die hierzu vorgesehenen Algorithmen sind noch in der Entwicklung.

3.1.4 Die Vierte Triggerstufe

In der vierten Stufe werden nach einer vollständigen Rekonstruktion die Ereignisse klassifiziert und noch einmal selektiert. Die vierte Triggerstufe dient auch zur kontinuierlichen Überwachung der Datenqualität. Die Berechnungen finden hier auf einer abstrakteren Ebene statt als in den vorgehenden Triggerstufen. Grundlage der Rekonstruktion sind nicht mehr Trefferinformationen in den Subdetektoren, sondern die in den vorgehenden Triggerstufen gefundenen Spuren. Mit einer Ausgangsrate von etwa 20 Hz werden dann die Daten einem zentralen Rechner zur Archivierung zugeführt.

3.2 Das Schnelle Kontrollsystem

Das Schnelle Kontrollsystem oder kurz FCS[7], dient der Synchronisierung aller Ereignisse bei der Datennahme mit dem HERA-B Detektor [ful99]. Das FCS verwendet den zentralen HERA-Takt und leitet daraus ein Taktsignal ab, welches den Zeitpunkten der physikalischen Ereignisse im HERA-B Detektor entspricht. Dieses mit den Ereignissen synchrone Taktsignal steuert die gesamte Datennahme. Dazu wird das Taktsignal und eine fortlaufende Nummer zur Identifizierung der Ereignisse an alle Front End Driver übermittelt, die die Auslese der Subdetektoren vornehmen. Diese benötigen das Signal zur Bestimmung der Zeitfenster, in denen die Daten zu einem bestimmten Ereignis registriert werden können. Die Detektordaten von 128 Ereignissen werden in den Ringspeichern der Front End Driver gespeichert. Wenn der First Level Trigger ein Ereignis auswählt, wird die Ereignisnummer des ausgewählten Ereignisses an das Fast Control System übermittelt. Dies veranlaßt darauf die Übermittlung der zu dem ausgewählten Ereignis gespeicherten Information an den Second Level Buffer, der im Abschnitt 3.1.2 vorgestellt wird.

3.3 Die Pretrigger

Der First Level Trigger sucht mit dem Netzwerk der Track Finding Units Teilchenspuren in speziellen Lagen der inneren und äußeren Spurkammersysteme. Der dabei verwendete

[7] **F**ast **C**ontrol **S**ystem, engl. für 'Schnelles Kontrollsystem'

Algorithmus ist auf sogenannte Spurkeime angewiesen, die neben den Startpunkt der Spur auch die Sorte des zur Spur gehörigen Teilchens angeben. Die drei Pretriggersysteme stellen dem First Level Trigger die Spurkeime für diese Spurensuche zur Verfügung. Sie sollen im folgenden kurz vorgestellt werden.

3.3.1 Der ECAL Pretrigger

Der Elektronen-Pretrigger oder auch ECAL Pretrigger verwendet Informationen des elektromagnetischen Kalorimeters, um Spurkeime von Elektronen und Photonen für den First Level Trigger bereitzustellen. Die ECAL Pretrigger Elektronik ist hierzu direkt mit der Front End Elektronik des Kalorimeters verbunden. Der vorgesehene Pretriggeralgorithmus läuft in zwei Schritten ab [loh94].
Zunächst wird von der Front End Elektronik ein Signal an die Pretrigger Elektronik gesandt, wenn die deponierte Energie in einer Kalorimeterzelle einen Schwellenwert von 1 GeV überschreitet.
In den Pretrigger-Boards wird er Tatsache Rechnung getragen, daß sich die Energiedeposition eines Teilchens über mehrere Kalorimeter-Zellen verteilen kann. Da Elektronen mit hohem Transversalimpuls gesucht werden, wird hier ein ortsabhängiger Schwellenwert verwendet. Um ein Pretrigger-Signal zu generieren, muß zum einen die Energie einer Kalorimeterzelle die Hälfte der ortsabhängigen Schwelle überschreiten. Zum anderen muß die Summe der Energien der betrachteten und der umliegenden acht Kalorimeter-Zellen die Triggerschwelle erreichen. Eine solche auf neun Kalorimeter-Zellen verteilte Energie-Deposition wird auch *Cluster*[8] genannt.
Der ortsabhängige Schwellenwert für den Pretrigger soll in zwei verschiedenen Weisen bestimmt werden. Für den Fall, daß der Magnet während der Inbetriebsetzung des HERA-*B* Detektors abgeschaltet ist, laufen die im Vertex entstandenen Teilchen auf geraden Trajektorien durch den Detektor. Dann läßt sich direkt aus dem Betrag und dem Ort der im Kalorimeter gemessenen Energiedeposition der Impuls des beteiligten Teilchens bestimmen:

$$\vec{p} = E \cdot \frac{\vec{r}}{|\vec{r}|}$$

wobei E die im *Cluster* deponierte Energie und $\vec{r}$ der Ort des *Cluster*s ist (der Ursprung falle mit dem Wechselwirkungspunkt zusammen). Auf diese Weise kann der Transversalimpuls[9] p_T eines zu einem *Cluster* gehörigen Teilchens gefunden und als Triggerkriterium verwendet werden. Für den Fall, daß der Magnet eingeschaltet ist, ist eine exakte Berechnung des Teilchenimpulses allein aus den Daten des elektromagnetischen Kalorimeters nicht möglich. Um trotzdem den Transversalimpuls als Triggerkriterium verwenden zu können, wird eine Abschätzung vorgenommen. Die Energieschwelle, die ein *Cluster* für ein Pretriggersignal überschreiten muß, wird durch die Formel

$$E_{Schwelle} = K_{trig} \cdot \left(\frac{1}{\sqrt{x^2+y^2}} + \frac{1}{\sqrt{x^2+|y|^3}} \right), K_{trig} = 450 - 1000\,\mathrm{GeV} \cdot \mathrm{cm}$$

[8] engl. für 'Anhäufung'

[9] Als Transversalimpuls wird die Impulskomponente der Teilchen senkrecht zur Protonstrahlrichtung bezeichnet.

bestimmt [loh94]. Der erste Summand in der Klammer entspricht der einfachen Transversalimpulsbestimmung, wie sie im Fall des abgeschalteten Magneten vorgenommen wird. Der zweite Summand soll näherungsweise den Einfluß des Magnetfeldes berücksichtigen.

3.3.2 Der Myon Pretrigger

Der Myon Pretrigger sucht nach Koinzidenzen in der Myonenkammern und sendet diese als Pretriggerinformation an den FLT weiter. Ein Myonkandidat wird durch eine Koinzidenz von Treffern in den *Pad*-Kammern der letzten beiden Myon-Spurkammerlagen definiert. Im inneren Bereich des Detektors wird eine Koinzidenz zwischen einem *Pad* der dritten Lage und einer von vier *Pad*s der vierten Lage akzeptiert, während im äußeren Bereich eine Koinzidenz zwischen einem *Pad* der dritten Lage und einer von sechs *Pad*s der vierten Lage gültig ist (s. Abb. 3.1).

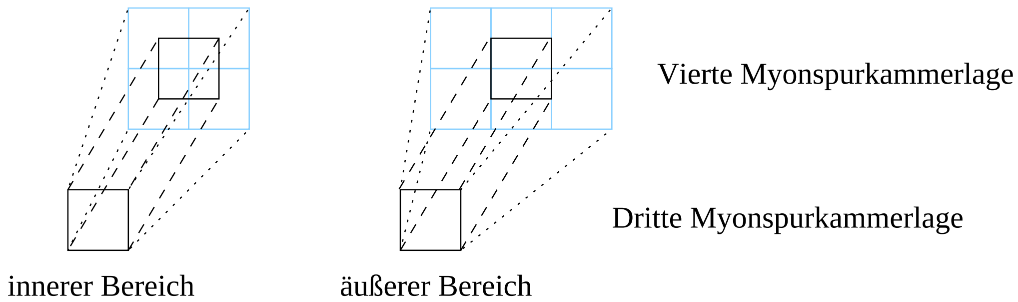

Abbildung 3.1: Koinzidenz-Schema für den Myon Pretrigger

3.3.3 Der High-p_T Pretrigger

Neben dem 'goldenen Zerfall' mit dem charakteristischen Kanal $J/\psi \rightarrow l^+l^-$ soll der First Level Trigger auch auf hadronische B-Zerfälle reagieren können. Aufgrund der hohen Masse der zerfallenden B-Mesonen können die hadronischen Zerfallsprodukte einen hohen Transversalimpuls besitzen und somit in großer transversaler Distanz vom Protonenstrahlrohr auftreten. Daher bietet es sich an, Spurkammern als Pretrigger zu verwenden, die in einem entsprechenden Bereich positioniert sind. Ein Treffer in einer solchen Spurkammer kann dann als Spurkandidat für ein Hadron mit hohem Transversalimpuls dienen.
Zur Realisierung werden drei spezielle Spurkammerlagen im Magnetfeld installiert, die mit der gleichen Front-End-Elektronik ausgelesen werden können, wie sie auch im Myonsystem verwendet wird.

3.4 Das FLT-Netzwerk

Der First Level Trigger besteht aus einem Netzwerk von speziell für die Erfordernisse des Experiments entwickelten Prozessoren. Es gibt drei Arten von Prozessoren:

- Track Finding Unit (TFU)
  zur Spursuche in den Spurkammerlagen.

- Track Parameter Unit (TPU)
  zur Berechnung der kinematischen Parameter der rekonstruierten Spuren
- Trigger Decision Unit (TDU)
  zur Auffindung der Triggerentscheidung.

Ein Überblick des FLT-Netzwerks ist in Abbildung 3.2 gezeigt.

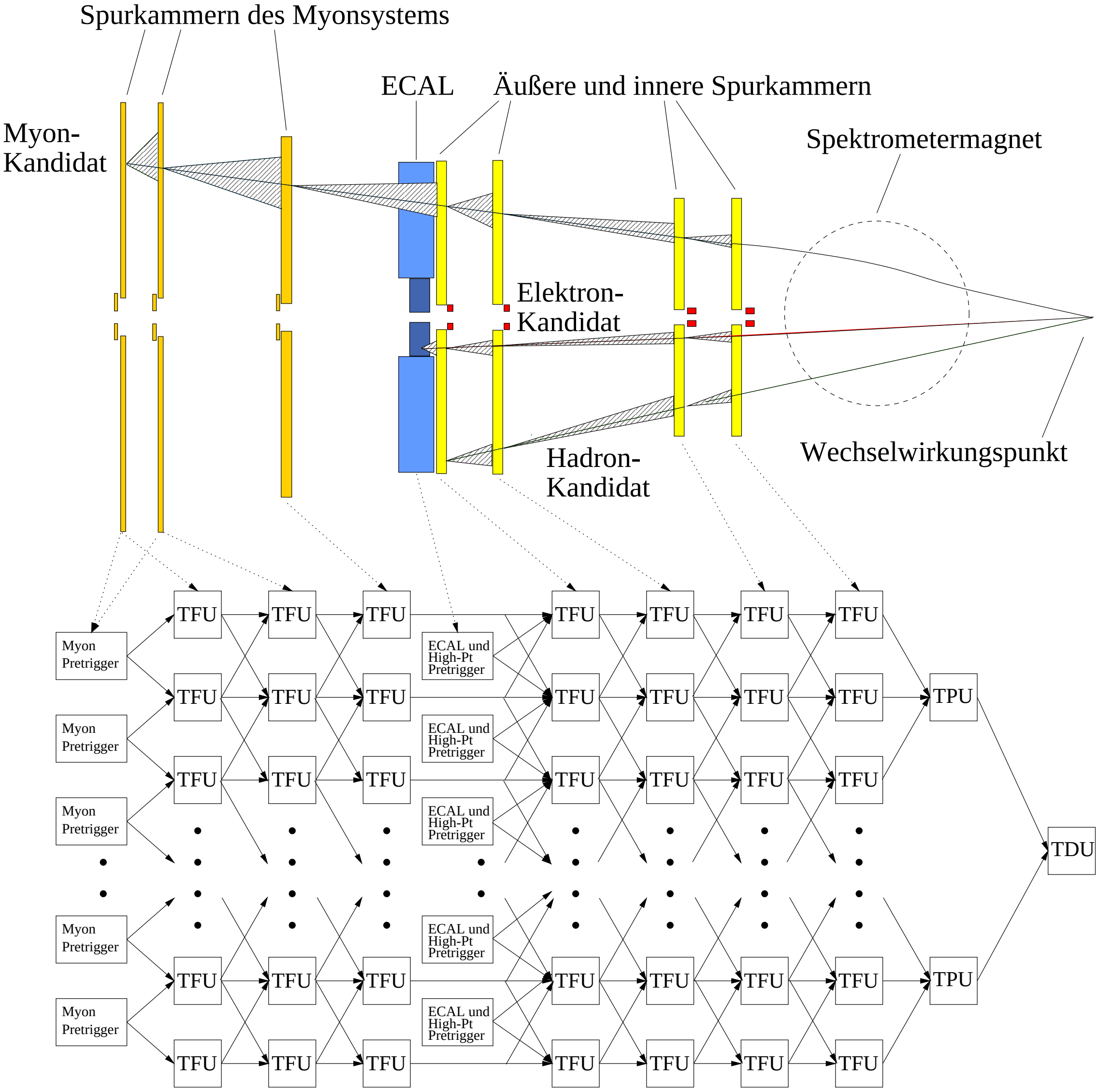

Abbildung 3.2: Das FLT-Netzwerk, nach [itt99]. Es sind nur die für den FLT verwendeten Subdetektoren eingezeichnet. Es ist angegeben, wie die Spurkammer-Superlagen den verschiedenen Bereichen des FLT-Netzwerks zugeordnet sind. Außerdem ist die Spursuche für einen Myon-Kandidaten, einen Elektron- und Hadron-Kandidaten angedeutet. Der Algorithmus der Spursuche ist im Text beschrieben.

Jeder Ebene der Spurkammern, bestehend aus den Superlagen der äußeren und inneren Spurkammern, die vom FLT verwendet werden, sind zehn bis sechzehn Track Finding

Units zugeordnet. Die Trefferinformation werden ihnen über Glasfaserkabel von der Front End Elektronik der äußeren und inneren Spurkammersysteme übermittelt. Parallel dazu erhalten die Track Finding Units Informationen über bereits gefundene Spurkandidaten. Die Spurkandidaten stammen aus den Pretriggern und werden dann von einer TFU-Lage zur nächsten weitergereicht. Wie im folgenden Abschnitt erläutert wird, besteht die wesentliche Aufgabe der TFUs darin, nach einem Treffer in dem ihr zugeordneten Bereich der Spurkammern zu suchen, der mit einem der Spurkandidaten in Deckung gebracht werden kann. Ist diese Suche erfolgreich, so werden die Parameter des Spurkandidaten unter Berücksichtigung des gefundenen Treffers aktualisiert und an die nächste Lage von TFUs weitergereicht. Wird kein Treffer gefunden, so wird der Spurkandidat verworfen. Die letzte Lage von Track Finding Units gibt die Informationen über die gefundenen Spuren an die Track Parameter Units weiter. Es sind ein bis drei Track Parameter Units vorgesehen, die die kinematischen Parameter der gefundenen Spuren berechnen. Diese werden in der Trigger Decision Unit zur Triggerentscheidung herangezogen. Dort werden zum einen die Anzahl der gefundenen Spuren gezählt, um bei einer programmierbaren Schwelle einen Trigger auszulösen. Zum anderen werden zu allen kombinatorisch möglichen Spurpaarungen eines Ereignisses die invarianten Massen berechnet, die ebenfalls zur Triggerentscheidung herangezogen werden. Die Track Finding Unit, die Track Parameter Unit und die Trigger Decision Unit werden in den folgenden drei Abschnitten genauer betrachtet.

3.5 Die Track Finding Unit

Das TFU-Netzwerk erhält zum einen Informationen von den Pretriggern, zum anderen von ausgewählten Spurkammern. Die von den Pretriggern gelieferten Spurkandidaten werden in den Spurkammern weiterverfolgt und die Spuren so rekonstruiert. Als Photonen werden Spurkeime registriert, die eine programmierbare Energieschwelle im Kalorimeter überschritten haben und zu denen keine Koinzidenz in den Spurkammerlagen gefunden wird. Diese Photonkandidaten werden ohne Einbeziehung der Daten aus den Spurkammern zur TPU propagiert, da ungeladene Teilchen keine Signale in den Spurkammern hervorrufen. Die Track Finding Unit besteht im wesentlichen aus drei Abschnitten[10]:

- Dem *Wirememory*[11], in dem die Informationen über die getroffenen Drähte in den Spurkammern gespeichert werden.
- Der Koinzidenzmatrix, die nach Treffern in den sogenannten *Regions of Interest*[12] sucht.
- Einem Abschnitt, in dem bei einer Koinzidenz von Region of Interest und Treffer die Spurparameter aktualisiert und an die nächste TFU-Lage übermittelt werden.

Die Front End Driver der inneren und äußeren Spurkammern besitzen neben dem Ausgang für den Second Level Buffer einen weiteren, der die Trefferinformation mit verminderter Präzision zur Verfügung stellt. Durch Verzicht auf die Bewertung der Driftzeiten in den äußeren Spurkammern wird die Auslese an dieser Stelle beschleunigt. Bei den inneren

[10] Eine detailliertere Beschreibung ist in [ger98] zu finden.
[11] engl. für 'Drahtspeicher'
[12] engl. für 'Region von Interesse'

Spurkammern werden jeweils die Ausgänge von vier Auslesestreifen verodert. Bei den äußeren Spurkammern wird auf die Auswertung der Driftzeit verzichtet. Diese Informationen werden über Glasfaserkabel direkt an die TFUs geleitet, wobei jede TFU fest einem geometrischen Bereich einer Superlage zugeordnet ist. Einer Superlage sind dabei zehn bis sechzehn TFUs zugeordnet von denen stets vier die Daten der inneren Spurkammern verarbeiten. Es werden die Daten der letzten 128 Ereignisse gespeichert.
Die Trefferinformationen der drei Stereolagen werden im Wirememory separat gespeichert. Damit ein Treffer akzeptiert wird, muß eine Koinzidenz in allen drei Stereolagen vorliegen. Der dazu verwendete Algorithmus ist mittels der Koinzidenzmatrix implementiert. Sie hat zum einen Zugriff auf das Wirememory, aus dem sie Trefferinformationen der letzten 128 Ereignisse anfordern kann. Zum anderen erhält sie die Parameter der Spurkandidaten, die von der vorgehenden TFU-Lage oder den Pretriggern geliefert werden.

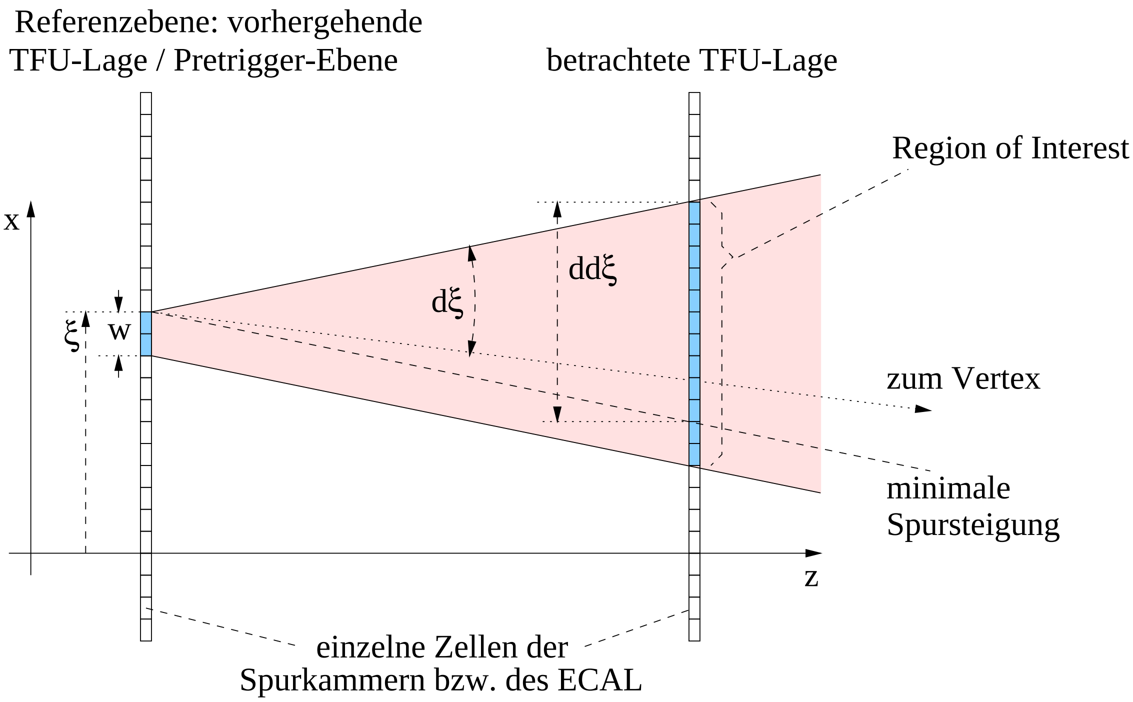

Abbildung 3.3: Definition der Spurparameter ξ, $d\xi$, $dd\xi$ und w in der x-z-Ebene (horizontal und in Strahlrichtung), welche eine Region of Interest repräsentieren. Die Parameter werden stets in Bezug auf die vorhergehende TFU-Lage bzw. die Pretriggerebene berechnet. Abbildung nach [rat97].

Ein Spurkandidat wird durch einen Raumbereich wiedergegeben, in dem die Trajektorie des Spurkandidaten liegt (s. Abb. 3.3 und 3.4). Dieser Raumbereich wird begrenzt durch die Schnittfläche mit der Referenzebene, welche die Ebene der vorhergehenden TFU-Lage bzw. der Pretrigger ist. Die Schnittfläche wird durch die Breite in x-Richtung w und die maximale Entfernung vom Strahlrohr in positiver x-Richtung ξ sowie den Mittelpunkt in y-Richtung η parametrisiert. Der die Trajektorie enthaltende Raumbereich wird von der Schnittfläche in der Referenzebene in y-Richtung durch Projektion in Vertex-Richtung fortgesetzt, da der Spektrometermagnet die Teilchen nur in x-z-Ebene ablenkt. In x-Richtung wird durch die Kanten der Raumbereich vor der Referenzebene parametrisiert. $d\xi$ ist hier-

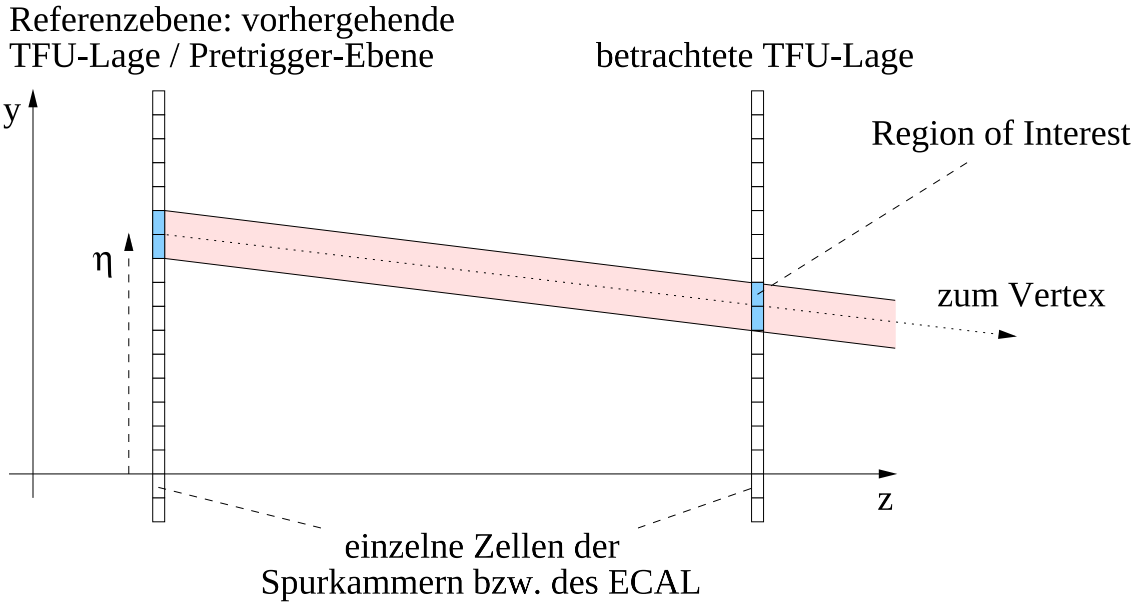

Abbildung 3.4: Definition des Spurparameters η in der y-z-Ebene (vertikal und in Strahlrichtung), welche eine Region of Interest repräsentieren. Die Parameter werden stets in Bezug auf die vorhergehende TFU-Lage bzw. die Pretriggerebene berechnet. Abbildung nach [rat97].

bei der Winkel zwischen der Kante, die in der Referenzebene die positivere x-Koordinate hat, und der Richtung zum Vertex. Damit beschreibt $d\xi$ die Ablenkung des Teilchens im Magnetfeld und ist somit direkt ein Maß für den reziproken Impuls. Die Kante, die in der Referenzebene die negativere x-Koordinate hat, wird über die Ausdehnung $dd\xi$ in x-Richtung des durch die beiden Kanten eingeschlossenen Bereichs in der betrachteten TFU-Ebene angegeben. Dieser Bereich zuzüglich w bildet die Ausdehnung der Region of Interest in x-Richtung in der betrachteten TFU-Ebene. Die Ausdehnung in y-Richtung wird als konstant angenommen.

Der in der Koinzidenzmatrix implementierte Algorithmus sucht nach Treffern in der Region of Interest. Dabei wird eine Koinzidenz in allen drei Stereowinkeln gefordert. Wird eine solche Koinzidenz gefunden, so werden die Spurparameter entsprechend neu berechnet und an die folgende TFU-Lage bzw. an die Track Parameter Units weitergeleitet. Wird keine solche Koinzidenz gefunden, so wird der Spurkandidat verworfen. Mehrere nebeneinander liegende Koinzidenzen werden als eine zusammenhängende betrachtet und zusammengefaßt. Dies ist notwendig, da ein Teilchen häufig Signale in nebeneinanderliegenden Kammern auslöst. Es kann dabei der Fall eintreten, daß in einer Reihe von ansprechenden Kammern eine innen liegende Kammer nicht anspricht. In diesem Fall können zu einem einlaufenden Spurkandidaten mehrere auslaufende Spurkandidaten erzeugt werden. Durchlaufen solche dublizierten Spurkandidaten das gesamte TFU-Netzwerk, so führt dies zu 'Spurzwillingen', die die Datenlast unnötig erhöhen und im Count Trigger der Trigger Decision Unit, der die Anzahl von Spurkandidaten als Triggerkriterium heranzieht, zu falschen Ergebnissen führt. Daher sind sowohl in der Track Parameter Unit als auch in der Trigger Decision Unit Mechanismen vorgesehen, die mehrfache Spurkandidaten mit fast gleichen Parametern bis auf den ersten Spurkandidaten verwerfen.

3.6 Die Track Parameter Unit

Die TPU ermittelt aus den geometrischen Daten der rekonstruierten Spuren die kinematischen Parameter, welche durch die Impulskomponenten angeben sind. Im Fall der Elektronspuren werden hierzu nicht nur die Spurparameter herangezogen, sondern auch zusätzliche Informationen aus dem Kalorimeter verwendet. Sie dienen zum Ausgleich von Bremsstrahlungsverlusten der Elektronen vor dem Magneten. Die Bestimmung der kinematischen Parameter wird mit Hilfe von Tabellen vorgenommen, den sogenannten Lookup-tables (siehe Abschnitt 4.2.4). Für jede Kombination der Spurparameter, die auftreten können, sind die zugehörigen Impulskomponenten in den Tabellen abgelegt. Die kinematischen Parameter müssen also nicht stets neu berechnet werden, sondern sie werden aus den Tabellen herausgelesen werden, was den zeitlichen Aufwand minimiert. Ein weiterer Vorteil dieses Verfahren ist, daß die real gemessene Feldverteilung des Spektrometermagneten in den verwendeten Tabellen berücksichtigt werden kann.
Weiterhin ist die sogenannte *Double Track Rejection*[13] vorgesehen. Hierzu werden die 21 letzten Spurkandidaten gespeichert und mit dem aktuellen Spurkandidaten verglichen. Ist der aktuelle Spurkandidat fast identisch mit einem der gespeicherten Spuren, so wird er verworfen.

3.7 Die Trigger Decision Unit

Die *Trigger Decision Unit* [14] oder kurz TDU erhält die kinematischen Parameter der rekonstruierten Spuren und wertet sie für die Triggerentscheidung aus. Eine grobe Übersicht der Komponenten bietet Abb. 3.5.
Es soll im folgenden die Funktionalität der einzelnen Komponenten der Trigger Decision Unit vorgestellt werden [groe98].

3.7.1 Messagecounter und -sequencer, Track- und Messagememory

Die Trigger Decision Unit erhält die Nachrichten von den Track Parameter Units in Form von 80 bit-Worten. Ein solches 80 bit-Wort soll im folgenden Nachricht oder auch *FLT-Message*[15] genannt werden. Die einzelnen Parameter dieser Nachricht sind in Tabelle 3.3 aufgeführt.
Insbesondere liegt für jede Nachricht der Parameter `BX`[16] vor, der die Nummer des Ereignisses bezeichnet. Die Trigger Decision Unit wird mit 50 MHz getaktet und kann bei jedem Taktzyklus, also alle 20 ns eine Nachricht entgegennehmen. Dies entspricht durchschnittlich maximal

$$\frac{96\,\mathrm{ns}}{20\,\mathrm{ns}} = 4{,}8$$

[13] engl. für 'Zurückweisung doppelter Spuren'
[14] engl. für 'Auslöse-Entscheidungs-Einheit'
[15] engl. für 'Nachricht'
[16] diese Bezeichnung leitet sich vom engl. *Bunch-Crossing* ab, was den 'Paket-Durchgang', d. h. das Durchqueren eines Protonenpakets durch den Targetbereich bedeutet.

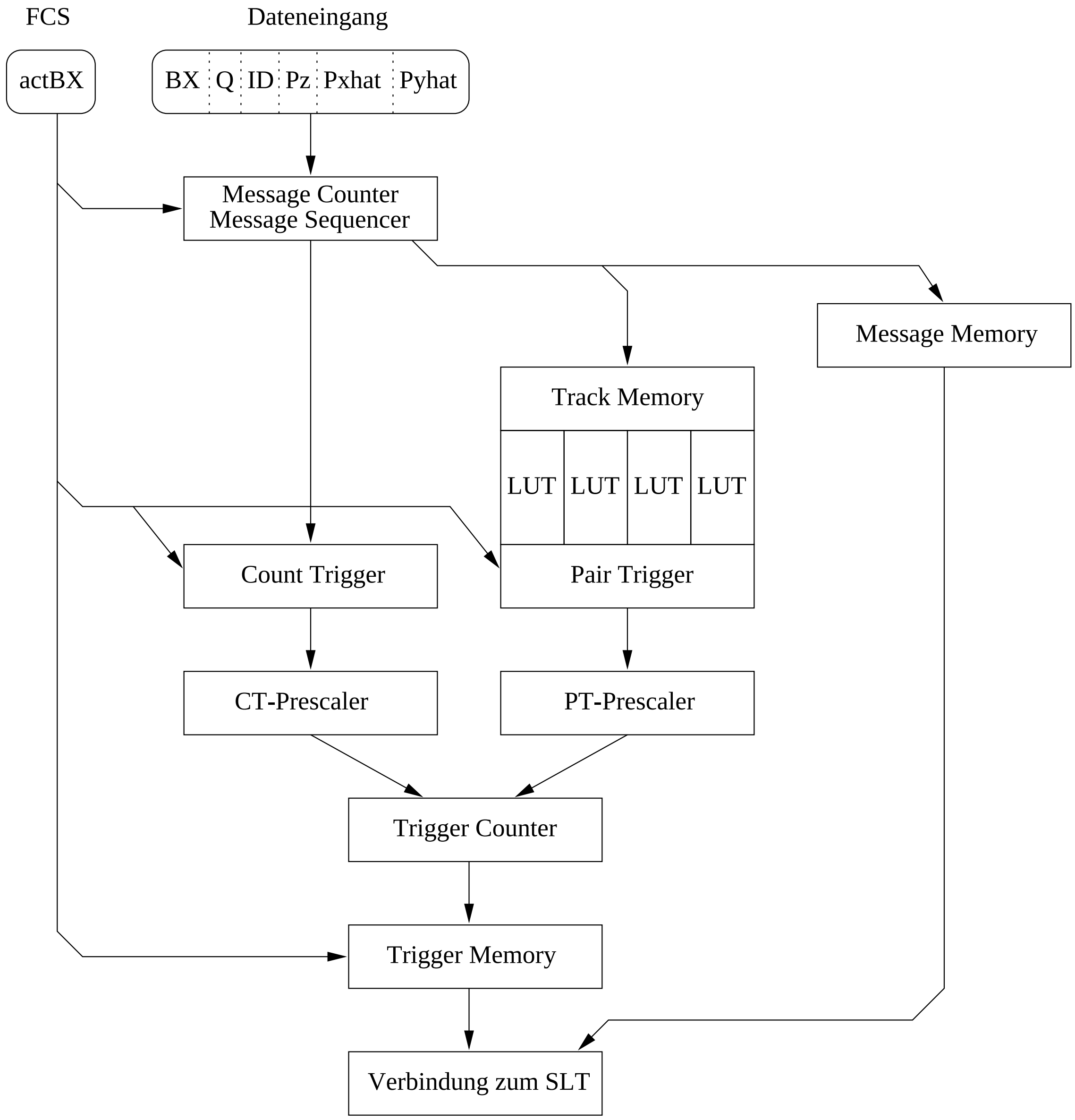

Abbildung 3.5: Datenfluß der Trigger Decision Unit

Nachrichten pro Ereignis, die von einer Trigger Decision Unit verarbeitet werden können. In der Praxis liegt dieser Wert allerdings deutlich niedriger, da bei Ereignissen mit hoher Spuranzahl zusätzliche Wartezyklen (s. u.) eingelegt werden müssen.
Der *Messagecounter*[17] hat die Aufgabe, die zu einem Ereignis eintreffenden Nachrichten zu numerieren und im *Track Memory*[18] und *Message Memory*[19] abzulegen. Eine grobe Übersicht des Datenflusses bietet Abb. 3.6.
Zu diesem Zweck ist im Messagecounter ein *Dual Ported Memory*[20] vorgesehen, in dem die

[17] engl. für 'Nachrichten-Zähler'
[18] engl. für 'Spur-Speicher'
[19] engl. für 'Nachrichten-Speicher'
[20] engl. für 'Zwei-Kanal-Speicher'

Parameter	Bedeutung	Bezeichner	Bits
$n\xi$	zwei Nachrichten	`nxi`	1
ξ	x-Koordinate in TC2-Lage	`xi`	10
$d\xi$	minimale Spursteigung	`dxi`	8
$dd\xi$	Größe des Spurfensters	`ddxi`	8
$\hat{P}_y$	P_y/P_z	`Pyhat`	9
η	y-Koordinate in TC2-Lage	`eta`	9
ω	Anz. benachbarter Zellen in TC2	`omega`	2
all	Treffer in jeder TFU	`all`	1
BX	Ereignis-Nummer	`BX`	8
ID	Pretrigger-Quelle	`ID`	2
P_z	z-Komponente des Impulses	`Pz`	8
$\hat{P}_x$	P_x/P_z	`Pxhat`	10
q	Ladung	`Q`	1

Tabelle 3.3: Eingabeparameter der Trigger Decision Unit [ger98]

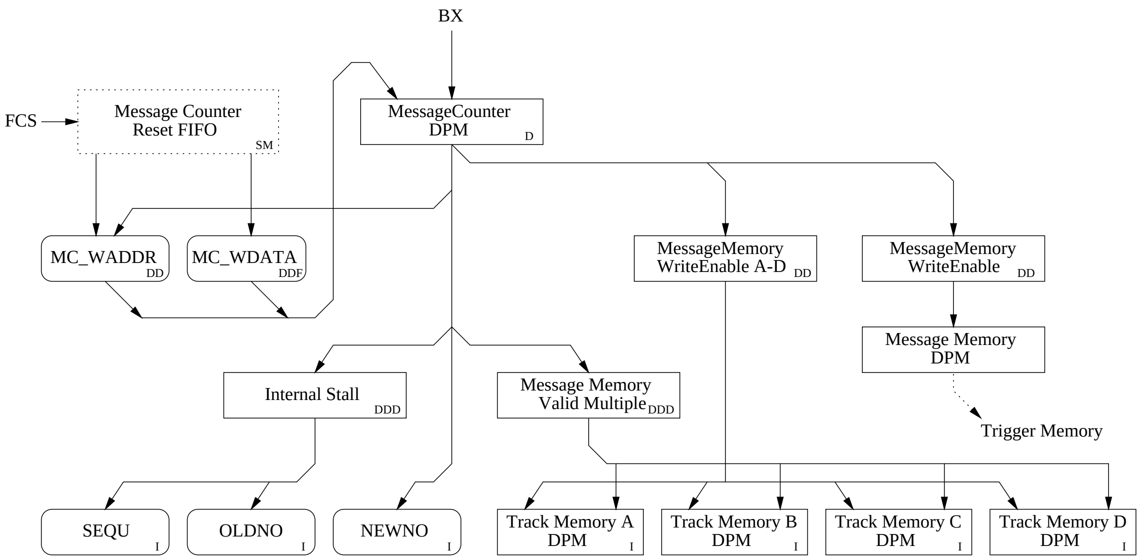

Abbildung 3.6: Vereinfachter Datenfluß von Messagecounter und -sequencer, Message- und Trackmemory

Anzahl der zu einem Ereignis eingetroffenen Nachrichten abgelegt sind. Dieser Zählerstand und die Nummer des Ereignisses dienen zur Adressierung der Nachrichten im Message Memory und im Track Memory. Die Zählstände werden periodisch durch die vom Fast

Control System gelieferte aktuelle Ereignis-Nummer gelöscht, so daß stets nur die letzten 128 Ereignisse berücksichtigt werden. In dem Track Memory werden lediglich die Parameter abgelegt, die für die Trigger-Entscheidung in der Trigger Decision Unit notwendig sind. Im Message Memory hingegen werden sämtliche Parameter der eintreffenden Nachrichten festgehalten um sie später dem Second Level Trigger zur Verfügung stellen zu können. Die teilweise Redundanz der gespeicherten Parameter in Track Memory und Message Memory sind erforderlich, um beide Speicher unabhängig voneinander auslesen zu können.

Der Message Sequencer hat darüber hinaus die Aufgabe, die Spurdaten den LUT-Prozessoren für die Berechnung der invarianten Massen bereitzustellen. Es stehen vier LUT-Prozessoren zur Verfügung, die parallel arbeiten. Damit alle zu einem Ereignis eintreffenden Spuren paarweise kombiniert werden können, ist das Track Memory viergeteilt. Die n-te Spur zu einem Ereignis wird dem (n modulo 4)-ten Bereich des Track Memorys zugeteilt. Die kombinatorische Zuteilung der Spuren zu den vier LUT-Prozessoren wird dann folgendermaßen vorgenommen: Empfängt der Message Sequencer die n- te Spur zu einem Ereignis, so wird diese Spur zunächst mit den Spuren 0 bis 3 dieses Ereignisses in die LUT-Prozessoren geleitet. Danach wird die n-te Spur mit den Spuren 4 bis 7 kombiniert und so fort bis alle Spuren dieses Ereignisses verarbeitet worden sind.

Aus diesem Verfahren ergibt sich, daß bis zu vier Spuren eines Ereignisses ohne zusätzliche Arbeitszyklen bearbeitet werden können, da bei der fünften Spur nur vier Vergleiche durchgeführt werden müssen. Erhält die Trigger Decision Unit hingegen mehr als fünf Spuren zu einem Ereignis, werden zusätzliche Taktzyklen benötigt, um alle kombinatorisch möglichen Vergleiche durchführen zu können (s. Tabelle 3.4).

Anzahl der Spuren	Wartezyklen
0 - 5	0
6 - 9	je 1 für Spuren 6 - 9
10 - 13	je 1 für Spuren 6 - 9 je 2 für Spuren 10 - 13
14 - 17	je 1 für Spuren 6 - 9 je 2 für Spuren 10 - 13 je 3 für Spuren 14 - 17
18 - 21	je 1 für Spuren 6 - 9 je 2 für Spuren 10 - 13 je 3 für Spuren 14 - 17 je 4 für Spuren 18 - 21
22 - 24	je 1 für Spuren 6 - 9 je 2 für Spuren 10 - 13 je 3 für Spuren 14 - 17 je 4 für Spuren 18 - 21 je 5 für Spuren 22 - 24

Tabelle 3.4: Wartezyklen in Abhängigkeit von der Anzahl der Spuren

Dieser Effekt bestimmt im wesentlichen die Latenzzeit der Trigger Decision Unit in Abhängigkeit von der Anzahl der zu verarbeiteten Spuren. Da die Anzahl zusätzlicher Arbeitszyklen mit der Anzahl der Spuren stark ansteigt, werden maximal 24 Spuren für ein Ereignis zugelassen. Um die Latenzzeit bei hoher Datenlast zu begrenzen, kann die Anzahl von maximal 24 Spuren pro Ereignis herabgesetzt werden. Hierzu wird die Differenz zwischen der aktuellen, vom Fast Control System bereitgestellten Ereignisnummer und der Nummer des gerade bearbeiteten Ereignisses gebildet. Ein hoher Wert für die Differenz ist ein Zeichen für eine hohe Latenzzeit. Die maximal zugelassene Anzahl von Vergleichen kann in Abhängigkeit dieser Differenz programmiert werden.

3.7.2 Count Trigger

Im *Count Trigger*[21] werden separat für jedes Ereignis die Anzahl der gefundenen Spuren gezählt. Die Spuren werden dabei in acht Kategorien eingeteilt. Zum einen wird nach der Quelle des Pretriggersignals unterschieden, was einer Sortierung nach den Teilchensorten Photon, Elektron, Myon und Hadron entspricht. Die Kandidaten für Photonen und Elektronen werden im elektromagnetischen Kalorimeter gefunden. Je nachdem, ob die TFUs Koinzidenzen von Treffern im Kalorimeter in der Spurkammerlage vor dem Kalorimeter finden oder nicht, werden die entsprechenden Spurkandidaten Elektronen oder Photonen zugeordnet. Myonen und Hadronen werden durch den Myon- bzw. High-p_T-Pretrigger identifiziert. Darüber hinaus wird in den LUT[22]-Prozessoren des *Pair Trigger*[23] (s. nächster Abschnitt) der Transversalimpuls der einlaufenden Spuren berechnet. Überschreitet der Transversalimpuls einen programmierbaren Schwellenwert, so wird die entsprechende Spur mit dem sogenannten High-p_T-Flag markiert, welcher im übrigen von dem *High-p_t*-Pretrigger unabhängig ist. So können die einlaufenden Spuren nach insgesamt acht Kategorien eingeteilt werden - vier Teilchensorten mit gelöschtem oder gesetztem High-p_T-Flag. Es lassen sich Schwellen für die Anzahl der gezählten Teilchen für jede dieser Kategorien separat und jeweils mit einer zusätzlichen Schwelle für die Anzahl der Hadronen programmieren, wie in Tabelle3.5 aufgezeichnet ist.

3.7.3 Pair Trigger mit LUT-Prozessoren

Die TDU bietet die Möglichkeit, zu den für ein Ereignis eintreffenden Spurkandidaten paarweise die invariante Masse zu berechnen und für die Triggerentscheidung zu verwenden. Hierzu werden auf der TDU-Platine vier kleinere Platinen aufgesetzt, in denen die LUT-Prozessoren implementiert sind. Der Datenfluß innerhalb dieser Prozessoren ist in Abb. 3.7 zu sehen.

Die LUT-Prozessoren dienen zur Berechnung der invarianten Masse eines Spurpaares. Dazu werden die in Tabelle 3.6 aufgeführten Parameter der aktuellen Spur und der im Trackmemory gespeicherten Spuren verwendet.

Um die invariante Masse eines Spurpaares zu berechnen, können wir in der Hochenergienäherung $E \approx P$ schreiben

$$M^2 = 2 \cdot (E_1 \cdot E_2 - \vec{P}_1 \cdot \vec{P}_2) = 2 \cdot E_1 \cdot E_2 \cdot (1 - \cos(\chi))$$

[21] engl. für 'Zähl-Auslöser'

[22] **Lookup**table, engl. für 'Nachschagtabelle'

[23] engl. für 'Paar-Auslöser'

Muster-nummer	e	μ	h	γ	high-p_T e	high-p_T μ	high-p_T h	high-p_T γ
1	n_1							
2		n_2						
3			n_3					
4				n_4				
5					n_5			
6						n_6		
7							n_7	
8								n_8
9	n_9		k_9					
10		n_{10}	k_{10}					
11			k_{11}	n_{11}				
12			k_{12}		n_{12}			
13			k_{13}			n_{13}		
14			k_{14}					n_{14}

Tabelle 3.5: Übersicht der möglichen Count Trigger-Muster. In den Spalten sind die vorhandenen Zähler aufgezeichnet, für Elektronen, Myonen, Hadronen und Photonen jeweils ohne und mit gesetztem High-p_T-Flag. Sobald alle programmierten Zählschwellen n_i und k_j bei einem Ereignis erreicht worden sind, löst ein Triggersignal mit der entsprechenden Musternummer ausgelöst.

wobei E_1 und E_2 die Energien und $\vec{P_1}$ und $\vec{P_2}$ die Impulse der beiden Teilchenspuren bezeichnen und χ der von $\vec{P_1}$ und $\vec{P_2}$ eingeschlossene Winkel ist. Um den Winkel χ zu substituieren, wird folgende Überlegung angestellt [ress95]. Der Öffnungswinkel zwischen den Spuren läßt sich aus dem Abstand d_M der beiden Spuren in der Magnetebene $z = z_M$, die mit dem Zentrum des Magneten zusammenfällt, berechnen:

$$d_M^2 = l_1^2 + l_2^2 - 2 l_1 l_2 \cos\chi$$

l_1 und l_2 bezeichnen hier die Abstände zwischen dem Vertex und den Schnittpunkten der Spuren mit der Magnetebene. Es gelten die Beziehungen

$$l_i^2 = x_i^2 + y_i^2 + z_i^2$$

$$l_1 \approx l_2 \approx z_1 \approx z_2 \approx z_M$$

Damit läßt sich der Abstand der Spuren in der Magnetebene auch durch

$$d_M^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$$

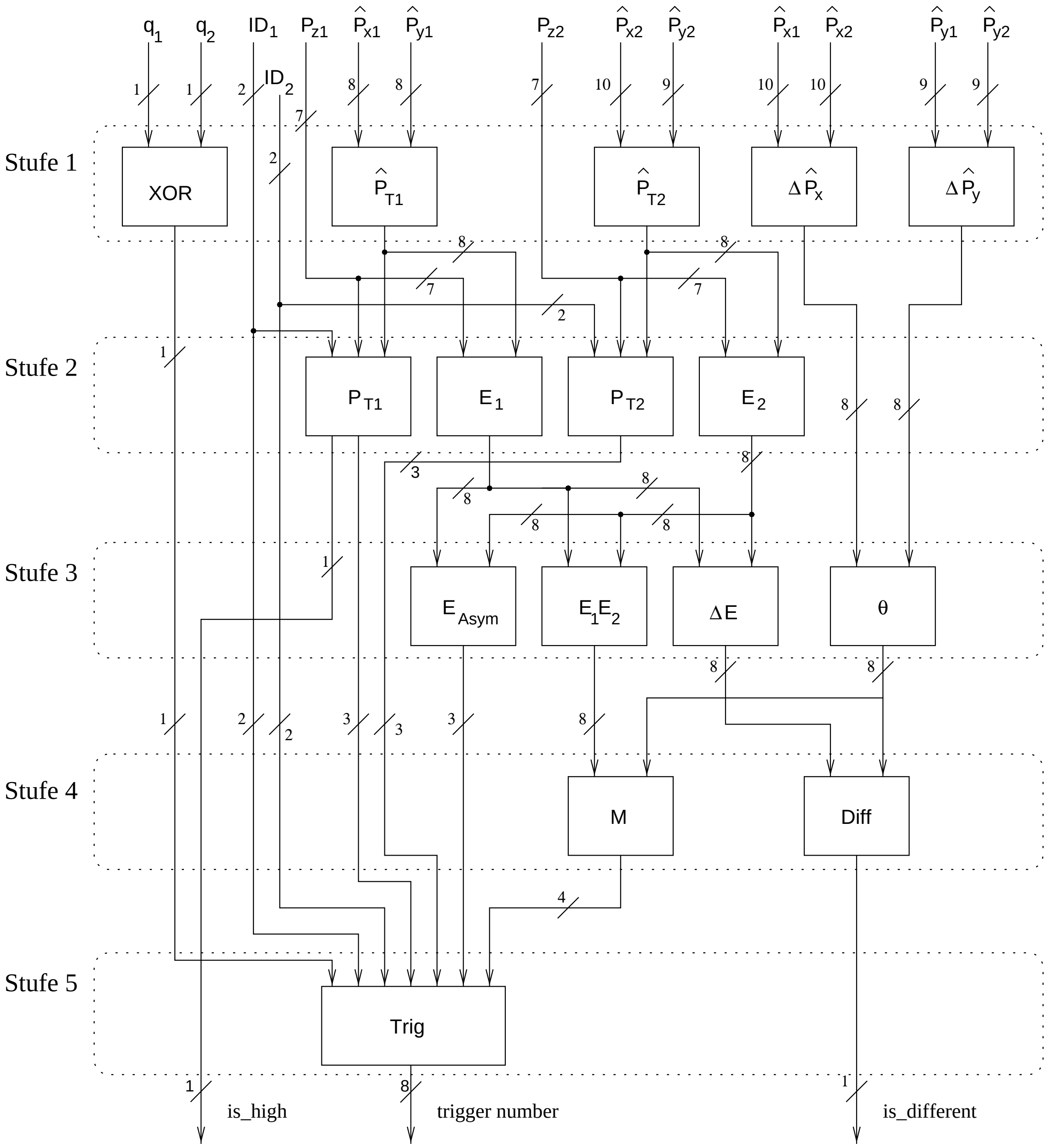

Abbildung 3.7: Datenflußdiagramm der LUT-Prozessoren [itt97]. Die Prozessoren sind in fünf Stufen aufgeteilt. Bei jedem Taktzyklus verarbeiteten alle Stufen die Ausgangswerte der jeweils vorhergehenden Stufe und erhalten so ihren eigenen neuen Ausgangswert. Jeder Kasten entspricht einer Lookuptable. Signale, die parallel zu den Lookuptables einer Stufe laufen, werden verzögert.

ausdrücken. So erhalten wir mit $\cos\chi \approx (1 - \frac{1}{2}\chi^2)$

$$
\begin{aligned}
d_M^2 &= 2z_M^2(1-\cos\chi) \approx z_M^2\chi^2 \\
&\approx (x_1 - x_2)^2 + (y_1 - y_2)^2
\end{aligned}
$$

Parameter	Bedeutung	Bits
q	Ladung	1
ID	Pretrigger-Quelle	2
P_z	z-Komponente des Impulses	8
$\hat{P}_x$	P_x/P_z	10
$\hat{P}_y$	P_y/P_z	9

Tabelle 3.6: Die im Pair Trigger der TDU verwendeten Spurparameter [ger98]. Neben der anhand der Pretriggerquelle identifizierten Teilchensorte ID und der Ladung des Teilchens q werden die drei Impulskomponenten P_z, $\hat{P}_x = P_x/P_z$ und $\hat{P}_y = P_y/P_z$ der aktuellen Spur und einer Spur des selben Ereignisses kombiniert.

Unter der Näherung, daß der Magnet keine Ausdehnung in Strahlrichtung hat, können wir annehmen, daß die Spuren vom Vertex bis zur Magnetebene gerade verlaufen. Damit erhalten wir folgenden Wert für den Öffnungswinkel χ:

$$\chi^2 \approx \frac{(x_{M1} - x_{M2})^2 + (y_{M1} - y_{M2})^2}{z_M^2} \approx (\Delta\hat{P}_x)^2 + (\Delta\hat{P}_y)^2$$

Somit ergibt sich der Ausdruck, mit dem die invariante Masse eines Spurpaares in den LUT-Prozessoren der TDU berechnet wird:

$$M^2 = 2 \cdot E_1 \cdot E_2 \cdot \left(1 - \cos\left(\sqrt{(\Delta\hat{P}_x)^2 + (\Delta\hat{P}_y)^2}\right)\right)$$

Die Berechnungen werden aus Gründen der Verarbeitungsgeschwindigkeit mit *Lookuptables* durchgeführt. Es handelt sich hierbei um Speicher, in der die Ergebnisse für die verwendeten Ausdrücke in tabellarischer Form abgelegt sind. Da nur endlich viele Einträge in diesen Tabellen Platz finden, ist die Genauigkeit der Berechnung begrenzt. Die mit der TDU erreichbare Genauigkeit wird in Kapitel 5 untersucht.
Die Berechnungen werden in fünf Stufen aufgeteilt, in der eine oder mehrere Berechnungen mit Hilfe von Lookuptables vorgenommen werden. Die in Abb. 3.7 angegebenen Lookuptables der einzelnen Stufen sollen hier kurz charakterisiert werden:

- Stufe 0, Eingangsparameter
  - $q_{1,2}$ sind die Ladungen der Spurkandidaten.
  - $ID_{1,2}$ gibt die Pretriggerquellen der Spurkandidaten an.
  - $P_{z1,2}$ bezeichnet die z-Komponente der Impulse der Spurkandidaten.
  - $\hat{P}_{x1,2}$ und $\hat{P}_{y1,2}$ sind die auf die z-Komponente normierten x- und y-Komponenten der Impulse der Spurkandidaten.
- Stufe 1
  - XOR gibt an, ob die beiden Spurkandidaten über das gleiche Ladungsvorzeichen verfügen.

  - $\hat{P}_{T1,2} = \sqrt{(\hat{P}_{x1,2})^2 + (\hat{P}_{x1,2})^2}$ bezeichnet die normierten Transversalimpulse der beiden Spurkandidaten.
  - $\Delta\hat{P}_{x,y} = |\hat{P}_{x1,y1} - \hat{P}_{x2,y2}|$ bezeichnet die Differenzen der normierten x- und y-Komponenten der Impulse.

- Stufe 2

  - $P_{T1,2} = P_{z1,2} \cdot \hat{P}_{T1,2}$ bezeichnet die Transversalimpulse; außerdem wird der Wert *is_high* bereitgestellt, der Spurkandidaten mit hohem Transversalimpuls kennzeichnet. Dieser Wert wird im Count Trigger zur Zählung von Spurkandidaten mit eben dieser Eigenschaft verwendet.
  - $E_{1,2} = P_{z1,2} \cdot \sqrt{1 + (\hat{P}_{T1,2})^2}$ bezeichnet die Energien der Spurkandidaten.

- Stufe 3

  - $E_{Asym} = \frac{E_1 - E_2}{E_1 + E_2}$ gibt die Asymmetrie der Energieverteilung auf die beiden Spurkandidaten an.
  - E_1E_2 und ΔE sind das Produkt und die Differenz der Energien E_1 und E_2.
  - $\theta = \sqrt{1 - \cos\left((\Delta\hat{P}_x)^2 + (\Delta\hat{P}_y)^2\right)}$ gibt den eingeschlossenen Winkel zwischen den Trajektorien der beiden Spurkandidaten an.

- Stufe 4

  - M ist die aus E_1E_2 und θ berechnete invariante Masse der beiden Spurkandidaten.
  - *Diff* gibt aufgrund der Werte von ΔE und θ Auskunft darüber, ob die beiden Spurkandidaten identisch sind, ob es sich folglich um einen Spurzwilling handelt, der von einem physikalischen Teilchen hervorgerufen wurde.

- Stufe 5

  - $Trig$ ist die sogenannte Triggernummer, eine Zahl von 0 bis 255. Eine 0 bezeichnet keinen Trigger, die Werte 1 bis 255 lassen sich programmierbaren Kriterien zuordnen, die sich aus den Größen XOR, $ID_{1,2}$, $P_{T1,2}$, E_{Asym} und M ableiten lassen.

Am Ausgang der Lookuptable Prozessors werden folgende Signale bereitgestellt:

- *is_high*, zu Kennzeichnung von Spurkandidaten mit hohem Transversalimpuls für den Count Trigger.
- Die Trigger Nummer *Trig* zur Charakterisierung der gefundenen Triggerentscheidung.
- *Diff* kennzeichnet Spurzwillinge, die sowohl im Count Trigger als auch im Pair Trigger verworfen werden.

3.7.4 Count Trigger- und Pair Trigger-Prescaler

Es sind zwei Prescaler[24] vorgesehen, die wie Frequenzteiler arbeiten. Der Count Trigger-Prescaler reduziert die Ausgangsrate des Count Triggers, der Pair Trigger-Prescaler die des Pair Trigger. Dazu werden die eintreffenden Triggersignale gezählt und nach einer einstellbaren Anzahl von Triggerereignissen weitergereicht. Auf diese Weise können die einzelnen Triggeroptionen in ihrer resultierenden Rate so aufeinander abgestimmt werden, daß zum einen die Ausgangsrate sich in einem Rahmen bewegt, der von dem Second Level Trigger bewältigt werden kann. Zum anderen können die relativen Raten ausbalanciert werden.
Der Prescaler des Count Triggers erlaubt die Reduktion der Ausgangsraten separat für jede Zählschwelle, wobei die Reduktionsrate in Stufen von Zweierpotenzen von 1 bis 2^{16} variiert werden kann.
Beim Pairtrigger kann die Reduktionsrate in Stufen von Zweierpotenzen von 1 bis 2^{16} eingestellt werden. Dabei ist für jede Triggernummer, die in den LUT-Prozessoren generiert werden kann (1-255), eine separate Einstellung möglich.

3.7.5 Trigger Memory und Kommunikation mit dem FCS

Im Trigger Memory[25] werden die positiven Triggerentscheidungen von Count Trigger und Pair Trigger, die die Prescaler passiert haben, nach der Ereignisnummer sortiert gespeichert. In diesem Speicher wird abgelegt, ob Count oder Pair Trigger zu einer Entscheidung geführt haben und welche Teilchensorten den Count Trigger ausgelöst oder welcher Massenbereich beim Pair Trigger aufgetreten ist. Außerdem wird festgehalten, welche der im Message Memory gespeicherten Spuren eines Ereignissen zu den Triggerentscheidungen geführt haben.
Wird zu einem Ereignis eine positive Triggerentscheidung gefunden, so wird dies sofort dem FCS mitgeteilt. Dies ist ein kritischer Zeitpunkt, da die Daten zu einem Ereignis in den Front End Driver Modulen nur für 128 Ereignisse gespeichert werden können und nach $128 \cdot 96\,\mathrm{ns} = 12{,}3\,\mu\mathrm{s}$ überschrieben werden.
Dem Second Level Buffer stellt sind die TDU wie ein Front End Driver Modul eines Subdetektors dar. Sobald das FCS die Daten von allen Subdetektoren für den Second Level Buffer anfordert, werden auch die in der TDU zu einem Ereignis gespeicherten Daten ausgelesen und dem Second Level Trigger zugesandt [groe99].
Die Triggerinformationen werden von der TDU in einem Format bereitgestellt, welches aus folgenden Elementen besteht [mic99]:

- Eine Liste von maximal 24 Einträgen des Message Memorys, in dem die Spurparameter abgelegt sind, die zu dem getriggerten Ereignis gehören.
- Eine Liste, die maximal 14 Einträge des Trigger Memorys enthält und so die gefundenen Triggerentscheidungen charakterisiert. Bei jeder verzeichneten Triggerentscheidung wird zusätzlich angegeben, welche der Spuren des Ereignisses zu der Triggerentscheidung geführt haben.

[24] engl. für 'Vorskalierer'
[25] engl. für 'Auslöse Speicher'

Der Second Level Trigger verwendet diese Informationen für seine Triggerentscheidung. Durch Kenntnis der Spurparameter, die im First Level Trigger einen Trigger ausgelöst haben, ist es möglich, gezielt nur Informationen aus den Bereichen des Detektors zu verwenden, die durch die Spurparameter des FLT definiert worden sind. So kann der Datentransfer vom Second Level Buffer zu der PC-Farm des Second Level Trigger minimiert und die Ausführungsgeschwindigkeit maximiert werden.

Kapitel 4

Simulation des First Level Trigger auf Taktebene

4.1 Motivation

Um das in Kapitel 3 beschriebene First Level Trigger Netzwerk zu verstehen, sind verschiedene Simulationen vorgesehen, die zum Teil bereits verwendet werden und zum Teil noch in der Entwicklungsphase sind. Die wesentlichen Aufgaben der Simulationen sind:

- Überprüfung des Algorithmus zur Spursuche in den Spurkammern, Ermittlung der kinematischen Parameter und Berechnung der invarianten Masse von Spurpaaren.
- Überprüfung und Optimierung des Einflusses des Binnings, d.h. der durch den Einsatz von Lookuptables bedingten Rechenungenauigkeit.
- Untersuchung des zeitlichen Verhaltens des First Level Trigger Netzwerkes. In diesem Zusammenhang ist insbesondere die Latenzzeit des First Level Trigger Netzwerks von Interesse, also das Zeitintervall vom Auftreten eines Ereignisse bis zur Triggerentscheidung des First Level Trigger. Sie darf insgesamt nicht den Wert von 12,3 μs überschreiten, da die Datennahme ansonsten nicht totzeitfrei arbeiten kann.

Um diese verschiedenen Anforderungen erfüllen zu können, sind drei 'Generationen' von Simulationen vorgesehen. Diese Generationen unterscheiden sich in der Detailliertheit der Modellierung der Hardware. Der grundsätzliche Algorithmus des Triggers kann in einer Simulation untersucht werden, der von der Hardware weitgehend abstrahiert. Eine detailliertere Simulation hingegen erlaubt ein genaueres Studium der Effekte, die durch die Implementierung des Triggeralgorithmus durch die vorgesehene Hardware auftreten. Die Hardwarenähe der detailierten Simulationen schlägt sich in einem höheren Programmieraufwand und Ressourcenbedarf des Programms nieder. Daher ist es sinnvoll, zunächst eine Simulation zu verwenden, welche schneller fertiggestellt werden kann und den grundsätzlichen Triggermechanismus modelliert. Später können dann Simulationen entwickelt werden, die präzisere Untersuchungen des Triggers erlauben.
Im Rahmen des First Level Trigger von HERA-B werden drei Generationen von Simulationen entwickelt:

Logische Simulation (L1SIMU)

Diese Simulation gibt den logischen Berechnungsablauf des First Level Trigger wieder, ohne die Einschränkungen zu berücksichtigen, die sich durch die Realisierung in der Hardware ergeben. Alle Berechnungen werden mit Hilfe von Gleitkommazahlen vorgenommen, also mit voller Präzision durchgeführt, was in der Hardware nicht möglich ist. Diese Simulation verarbeitet die Daten jedes Ereignisses separat, ohne die vorhergehenden Ereignisse zu berücksichtigen, so daß Latenzzeiteffekte nicht erfaßt werden können.

Simulation mit Binning und Lookuptables (LUTSIM)

Die Simulation LUTSIM unterscheidet sich von der logischen Simulation darin, daß die Berechnungen nicht mit Gleitkommazahlen durchgeführt werden. Statt dessen werden die in der Hardware verwendeten Digitalisierungen der verwendeten Größen vorgenommen (Binning) und zur Berechnung die zur Erzeugung der Lookuptables vorgesehenen Funktionen verwendet. Bei der als Binning[1] bezeichneten Digitalisierung wird eine kontinuierliche Definitionsmenge auf eine diskrete Wertemenge abgebildet. Bildlich werden dabei bestimmte Intervalle der zu digitalisierenden Größe verschiedenen 'Behältern' zugeordnet, die jeweils einem Element der diskreten Wertemenge entsprechen (siehe auch Abschnitt 4.2.5). So können die Genauigkeitsverluste untersucht werden, die durch die Berechnungsmethodik in der Hardware auftritt. Auch diese Simulation arbeitet ereignisorientiert und kann Latenzzeiteffekte nicht berücksichtigen.

Simulation des Zeitverhaltens auf Taktebene (FLTSIM)

In FLTSIM wird nicht nur die Digitalisierung der betrachteten Größen und die Lookuptable-Funktionen zu deren Berechnung berücksichtigt, sondern es wird das zeitliche Verhalten der Hardware wiedergegeben. Dazu werden alle wesentlichen Komponenten der Hardware auf Taktebene simuliert. Die Simulation ist nicht mehr ereignisorientiert sondern taktorientiert, d.h. jeder Taktzyklus in der realen Hardware entspricht einem Verarbeitungsschritt in der Simulation. Die gegenseitige Beeinflussung von nacheinander verarbeiteten Ereignissen wird so berücksichtigt und die daraus resultierenden Effekte können untersucht werden. Ein Hauptziel dieser Simulation ist die Bestimmung der Latenzzeit des First Level Trigger.

Die logische Simulation (L1SIMU) und die Simulation mit Binning und Lookuptables (LUTSIM) sollen hier nicht näher betrachtet werden. Vielmehr sollen die Anforderungen und die Realisierung des programmiertechnischen Rahmens für FLTSIM und die Simulation der Trigger Decision Unit innerhalb dieses Rahmens dargestellt werden.

4.2 Eigenschaften der Hardware des First Level Trigger

In diesem Abschnitt soll auf die Eigenschaften der FLT-Hardware eingegangen werden, die das Verhalten des FLT hinsichtlich der Triggereffizienz beeinflussen. Diese Eigenschaften sind es, die bei einer vollständigen Simulation des FLT berücksichtigt werden müssen.

[1] von *bin*, engl. für 'Behälter'

4.2.1 Die Pipeline-Struktur

Eine grundlegende Eigenschaft der Realisierung des First Level Trigger liegt in der umfangreichen Verwendung von Pipeline-Strukturen[2]. Die Verwendung dieser Architektur wird durch den Umstand ermöglicht, daß nur ein fester Algorithmus in der Hardware implementiert werden muß. In einer Mikroprozessor Architektur, in der ein durch einen Programmcode gegebener Algorithmus abgearbeitet wird, wird jeder Schritt des Algorithmus nacheinander durch den zentralen Mikroprozessor durchgeführt. Der Mikroprozessor kann durch die Verwendung verschiedener Programmcodes beliebige Algorithmen umsetzen, ist aber auf einem Verarbeitungsschritt zur Zeit beschränkt.[3] In der Pipeline Architektur wird nicht ein zentraler Prozessor eingesetzt, sondern die Schritte des Algorithmus werden in einer Reihe von diskreten Verarbeitungseinheiten umgesetzt. Diese Verarbeitungseinheiten werden als Pipesteps[4] bezeichnet und arbeiten unabhängig voneinander, so daß jede einen Schritt des Algorithmus parallel verarbeiten kann. Der Name 'Pipeline' deutet an, wie der Datenfluß realisiert ist. Die Daten gelangen in die erste Verarbeitungseinheit, wo der erste Schritt des Algorithmus umgesetzt wird. Das Resultat der ersten Verarbeitungseinheit wird

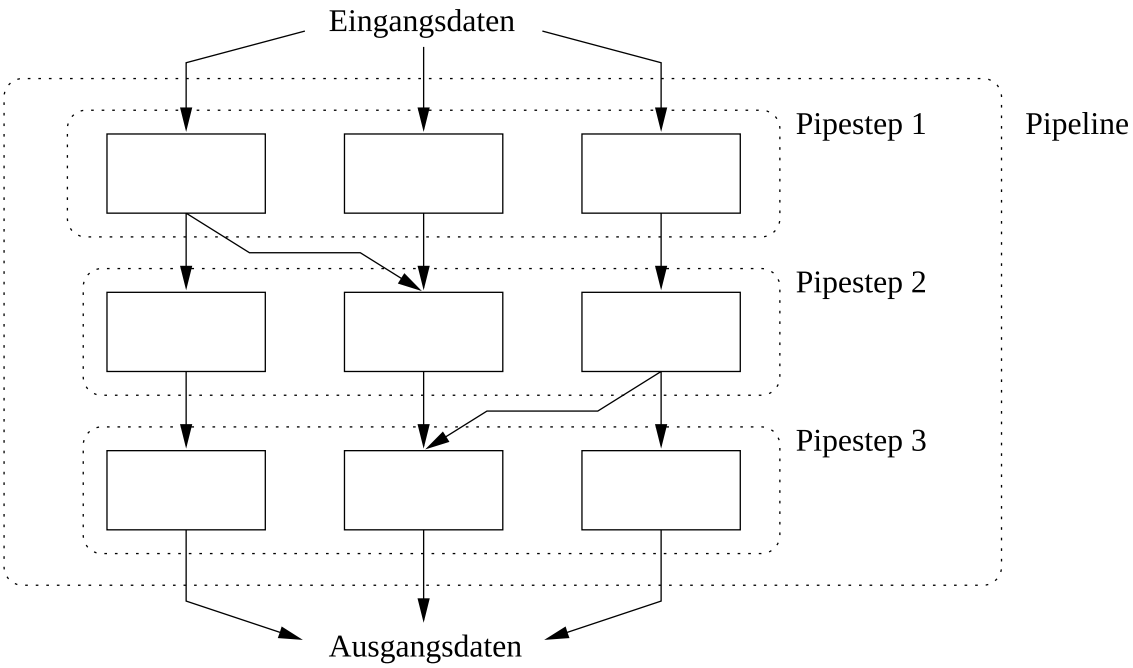

Abbildung 4.1: Einfaches Beispiel für den Aufbau einer Pipeline. Es sind drei Verarbeitungseinheiten (Pipesteps) eingezeichnet. Die Eingangsdaten erhält jede Pipestep von der jeweils vorhergehenden. Die Eingangsdaten der ersten Pipestep sind die Eingangsdaten der gesamten Pipeline und entsprechend sind die Ausgangsdaten der letzten Pipestep die Ausgangsdaten der Pipeline. Die Funktionalität der Pipesteps kann in parallel arbeitende Untereinheiten aufgeteilt werden, die hier durch Kästen angedeutet sind.

dann in die zweite Einheit übermittelt und dort verarbeitet. Gleichzeitig werden bereits

[2] *pipeline*, engl. für 'Rohrleitung'

[3] Für moderne Mikroprozessoren ist diese Aussage zu relativieren, da sie durch eine interne Pipelinearchitektur in der Lage sind, effektiv mehrere Arbeitsschritte parallel durchzuführen.

[4] engl. für 'Rohrabschnitt'

die nächsten Daten der ersten Verarbeitungseinheit bereitgestellt und dort verarbeitet. Auf diese Weise können gleichzeitig so viele Schritte des Algorithmus verarbeitet werden, wie Verarbeitungseinheiten vorhanden sind. Jede Verarbeitungseinheit besteht aus einer Reihe von Registern, welche die Synchronisation des Datentransports gewährleisten.

4.2.2 Die Register

Die Verarbeitungseinheiten oder Pipesteps einer Pipeline bestehen aus jeweils einer oder mehreren sogenannten Registern. Sie sorgen für das korrekte zeitliche Verhalten bei der Datenübertragung von Pipestep zu Pipestep. Die eigentliche Datenverarbeitung einer Verarbeitungseinheit wird durch Field Programmable Gate Arrays (kurz FPGA) und Lookuptables (kurz LUT) vorgenommen, die in den beiden folgenden Abschnitten vorgestellt werden. Die Datenverarbeitung in diesen Komponenten erfordert eine gewisse Zeit, die von Komponente zu Komponente unterschiedlich sein kann. Damit alle Signale einer Pipestep

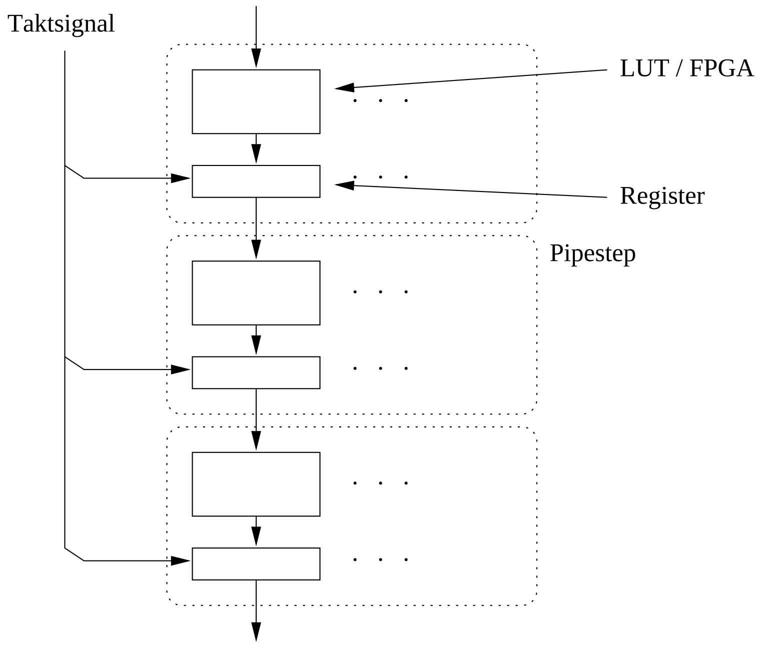

Abbildung 4.2: Funktionsweise von Registern. Mit jedem Taktsignal werden die Signale, die an den Eingängen der Register anliegen, an die Ausgänge kopiert. Die neuen Parameter werden in FPGAs oder LUTs (siehe Text) ermittelt und mit dem Taktsignal an den Ausgang des jeweiligen Registers übertragen.

synchron verarbeitet werden können, werden die Ausgänge der berechnenden Komponenten nicht direkt an die nachfolgende Stufe weitergereicht, sondern in einem Zwischenspeicher, das Register, geleitet. Das Kopieren des Eingangssignals an den Ausgang der Register wird durch ein Taktsignal gesteuert, welches alle Register empfangen. Das Zeitintervall zwischen zwei Taktflanken ist so bemessen, daß alle Komponenten ein stabiles Ausgangssignal liefern können, bevor die Signale an den Ausgang der Register kopiert werden. Nach dem Taktsignal liegen die aktualisierten Signale am Ausgang der Register, die Komponenten liefern neue Ausgangssignale, die beim nächsten Taktsignal wiederum an den Ausgang der Register kopiert werden. Auf diese Weise liegen mit jedem neuen Taktsignal synchron neue Werte an den Ausgängen der Register.

4.2.3 Die Programmierbaren Logikeinheiten

Viele Verarbeitungsschritte, mit denen in einer Pipeline ein Algorithmus verwirklicht wird, können mit einfachen logischen Gattern durchgeführt werden. Ein flexibles Werkzeug für diesen Zweck stellen Programmierbare Logikeinheiten dar, die auch FPGA[5] genannt werden. In einem FPGA sind neben Eingabe- und Ausgabebausteinen Logikelemente vorhanden, die durch eine variable Programmierung miteinander kombiniert werden können. Der Komplexität der Verbindung der Ein- und Ausgängen mit den Logikelementen sind durch die Größe des verwendeten Chips Grenzen gesetzt, so daß nicht beliebige Abbildungen von Ein- auf Ausgangswerte realisiert werden können, wie dies bei Lookuptables der Fall ist (siehe nächster Abschnitt). Der Vorteil von FPGAs ist hingegen, daß Exemplare mit einer hohen Anzahl von Eingangskanälen verfügbar sind.

4.2.4 Die Lookuptables

Sollen Verarbeitungsschritte angewandt werden, die sich nicht mit einem Netzwerk aus logischen Gattern begrenzter Komplexität verwirklichen lassen, so bieten Lookuptables[6] eine Alternative. Sie werden mit einem Speicherbaustein realisiert, dessen Speicherzellen mit den gewünschten Ausgangsdaten gefüllt werden. Die Werte der Eingangsdaten werden als Adresse innerhalb des Speicherbausteins verwendet. So werden die zu den Eingangsdaten gehörigen Ausgangsdaten im Speicher wie in einer Tabelle nachgeschlagen. Im Gegensatz zu den Programmierbaren Logikeinheiten können so beliebige Abbildungen von Eingangs- zu Ausgangswerten realisiert werden. Der Nachteil von Lookuptables gegenüber Programmierbaren Logikeinheiten liegt in der begrenzten Anzahl von Eingangskanälen, die durch die maximal verfügbare Größe von Speicherbausteinen gegeben ist. Je nach Anwendung bieten somit Programmierbare Logikeinheiten oder Lookuptables die geeignetere Lösung zur Verwirklichung der erforderten Verarbeitungsschritte.

4.2.5 Das Binning

Die Parameter, welche in der Pipeline verarbeitet werden, werden durch digitale Datenworte repräsentiert. Die Breite dieser Datenworte ist aufgrund der begrenzten Eingangsbreite der Lookuptables auf typischerweise maximal 10 bit begrenzt. Der Definitionsbereich der Parameter muß in geeigneter Weise auf die endliche Anzahl von Werten abgebildet werden, die sich mit dem Datenwort darstellen lassen. Die Abbildung $b\colon \mathcal{K} \to \mathcal{D}$ eines kontinuierlichen Parameters aus der Definitionsmenge $\mathcal{K}$ auf eine diskrete Menge $\mathcal{D}$ wird im folgenden als Binning oder Kodierung bezeichnet. Die Abbildung $r\colon \mathcal{D} \to \mathcal{K}$ von der diskreten Menge in den kontinuierlichen Parameter heißt Rebinning oder Dekodierung. Die Elemente aus $\mathcal{D}$ werden in der Hardware durch digitale Datenworte realisiert. Ein Datenwort von n bit kann 2^n verschiedene Werte annehmen. Der Definitionsbereich des zu kodierenden Parameters wird in $2^n - 1$ Intervalle aufgeteilt, die jeweils einem der diskreten Werte des Datenworts zugeordnet werden. Der verbleibende der 2^n Werte des Datenworts wird als ungültiger

[5] **F**ield **P**rogrammable **G**ate **A**rray, engl. für 'Feld programmierbarer Logikbausteine'

[6] engl. für 'Nachschlagtabelle'

Wert interpretiert, um Werte des Parameters, die nicht in einem der festgelegten Intervalle liegen, in einer definierten Weise behandeln zu können. [7] Bei der Dekodierung der Werte wird auf jedes Element des diskreten Definitionsbereichs ein geeigneter Wert aus dem ihm zugeordneten Intervall abgebildet. Hierdurch tritt unweigerlich eine Unsicherheit auf, die mit dem Kodieren und Dekodieren einhergeht. Diese Unsicherheit hängt zum einen von der Breite des zum Kodieren verwendeten Datenworts ab, da mit steigender Anzahl der zur Verfügung stehenden Intervalle die Breite derselben sinkt. Zum anderen ist es sinnvoll, das Binning so zu wählen, daß alle zur Verfügung stehenden Intervalle mit der gleichen Wahrscheinlichkeit verwendet werden. Dies läßt sich in der Praxis in guter Näherung mit zwei verschiedenen Binningmethoden erreichen, dem linearen und dem logarithmischen Binning. Beim linearen Binning sind alle Intervalle, die den Definitionsbereich des zu kodierenden Parameters aufteilen, von gleicher Größe. Sei N die Anzahl der Intervalle und $\mathcal{K} = [a, b)$ der Definitionsbereich des zu kodierenden Parameters, so ist ein lineares Binning $b_{lin} \colon [a, b) \to [0, N-1]$ gegeben durch

$$b_{lin}(x) = frac\left((N-1) \cdot \frac{x-a}{b-a}\right) \tag{4.1}$$

wobei $frac(x)$ definiert ist durch

$$frac(x) = \{x \in \mathcal{Z}, x - 1 < x \leq z\} \tag{4.2}$$

mit der Menge $\mathcal{Z}$ der ganzen Zahlen. Ein logarithmisches Binning $b_{log} \colon [a, b) \to [0, N-1]$ hingegen wird folgendermaßen implementiert:

$$b_{log}(x) = frac\left((N-1)\frac{\ln(a) - \ln(x)}{\ln(b) - \ln(a)}\right) \tag{4.3}$$

4.3 Anforderungen an FLTSIM

In diesem Abschnitt wird auf die Anforderungen der Simulation des First Level Trigger eingegangen, die das Zeitverhalten der realen Hardware berücksichtigt. Sie soll folgende Aspekte abdecken:

4.3.1 Simulation des Zeitverhaltens

Ein Hauptziel ist die Wiedergabe des Zeitverhaltens des First Level Trigger. Die für die Berechnungen verwendeten Größen werden in Registern gespeichert (siehe Abbildung 4.2). Die Register bestehen aus zwei Zwischenspeichern, in denen die Werte während der Berechnung bereitgestellt werden. Das Taktsignal sorgt für eine synchrone Aktualisierung der Werte. Zunächst wird unter Berücksichtigung der Eingangssignale der neue Wert bestimmt und in dem Eingangsspeicher abgelegt. Der neue Wert des Registers liegt zu dieser Zeit noch nicht am Ausgang vor, da die nachfolgenden Stufen noch den vorigen Wert für ihre Berechnungen benötigen. In einem zweiten Schritt wird der berechnete Wert von dem Eingangsspeicher in den Ausgangsspeicher übertragen. Der Taktzyklus ist abgeschlossen und der nächste beginnt.

[7] Aus diesem Mechanismus leitet sich die Bezeichnung 'Binning' ab. Jeder Wert des zu codierenden Parameters wird einem der durch die Intervalle definierten 'Behälter' oder Bins (*bin*, engl. für 'Behälter') zugeordnet.

Die Simulation soll diesen Mechanismus der getakteten Register wiedergeben. Eine weitergehende Simulation der realen Hardware wie z.B. die Berücksichtigung von Signallaufzeiten und anderen Effekten der Elektronik, die in kürzeren Zeitskalen als die des Taktsignals hingegen ist nicht vorgesehen, da keinen sie Einfluß auf die Triggereffizienz haben.

4.3.2 Modularität zur Simulation verschiedener FLT-Szenarien

Um verschiedene FLT-Szenarien mit der Simulation untersuchen zu können, ohne dazu tiefgreifende Änderungen an dem Programm vornehmen zu müssen, ist Modularität eine weitere Forderung an die Simulation. Zum einen sollen einzelne Bausteine des FLT durch einzelne Programmteile der Simulation wiedergegeben werden. Zum anderen sollen sich diese Programmteile analog zu den Bausteinen des FLT zusammensetzen lassen, um in der Simulation wie in der Hardware verschiedene Ausbaustufen des FLT zu untersuchen. Ein Beispiel für einen solchen Aufbau ist in Abbildung 4.3 zu sehen.

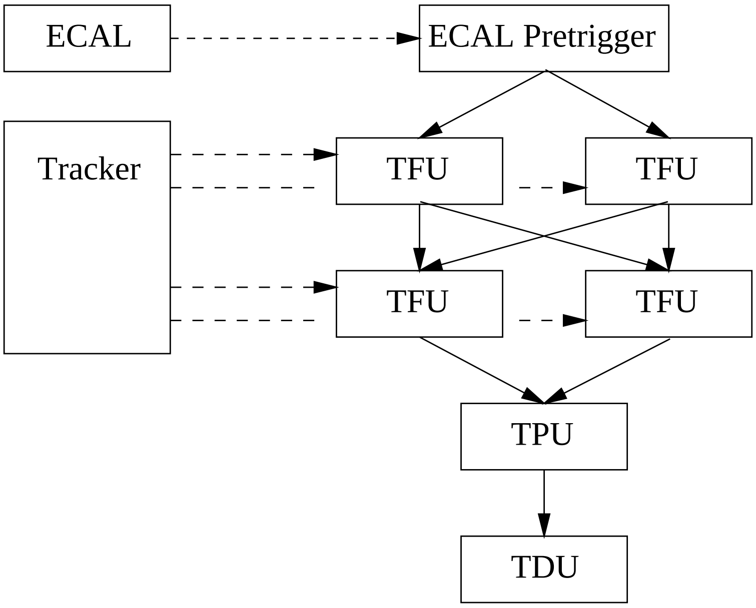

Abbildung 4.3: Beispiel eines zu untersuchenden Setups.

Dies ist ein typischer Aufbau des First Level Trigger in der Phase der Inbetriebnahme, in der das TFU-Netzwerk nach und nach vervollständigt werden soll, um ein detailiertes Verständnis des Systems gewinnen zu können.
Hierzu wird die Simulation in folgender Weise strukturiert:

- Trennung von programmiertechnischen Rahmen, welcher zur Wiedergabe hardwarespezifischer Eigenschaften dient, und FLT-spezifischen Programmabschnitten.
  Die Teile des Programms, die die oben beschriebene Funktionalität der getakteten Register bereitstellen, sollen unabhängig von der Modellierung der spezifischen Funktionen des First Level Trigger sein. Dazu ist ein programmiertechnischer Rahmen vorgesehen, der in dem Abschnitt 'Das tls-Framework', S. 54 beschrieben wird.

- Getrennte Behandlung verschiedener Bausteine des FLT.
  Die einzelnen Bestandteile des FLT werden alle als separate Programmelemente

behandelt. In der Sprache der objektorientierten Programmierung wird jedem in sich geschlossenem Bestandteil der Hardware eine Klasse zugeordnet. So werden die drei Pretriggersysteme, die TFUs, TPUs und TDU sowie die Messageboards, die die Datenübertragung zwischen diesen Elementen ermöglichen, separat modelliert.

4.4 Das tls-Framework

Das tls-Framework, welches den programmiertechnischen Rahmen der Simulation bildet, soll kurz in seiner Funktionalität vorgestellt werden. Zu dieser Beschreibung werden wir die Terminologie der objektorientierten Programmierung verwenden, die ein angemessenes Werkzeug für das vorliegende Problem darstellt. Technische Ausdrücke der objektorientierten Programmierung werden in diesem Abschnitt zur Kennzeichnung *kursiv* angegeben.

4.4.1 Grundzüge objektorientierter Programmierung

Das zentrale Konzept der objektorientierten Programmierung liegt darin, die in einem Programm behandelten Gegenstände in einer möglichst natürlichen Weise zu beschreiben. Hierzu sind sogenannte Klassen vorgesehen, die eine Beschreibung eines Gegenstandes darstellen. Die Beschreibung umfaßt die Eigenschaften, die ein Gegenstand besitzt. Zum anderen wird ein Gegenstand durch die Aktionen definiert, die man mit ihm durchführen kann. Hierzu bestehen Klassen aus den *Attributen* und *Methoden*. Attribute sind Variablen, die den Zustand eines Gegenstandes beschreiben. Methoden sind Funktionen, die eine Aktion mit dem Gegenstand wiedergeben. Die Menge von Attributen und Methoden definiert eine Klasse, die so eine Gegenstandsart charakterisert. Als *Objekte* oder *Instanzen* werden Manifestationen dieser durch die Klassen definierten Gegenstände bezeichnet. Jedes dieser Objekte hat seine eigenen Werte für die Attribute, die seinen Status beschreiben. Durch den Mechanismus der *Vererbung* können Klassen Attribute und Methoden von anderen Klassen übernehmen. In der betrachteten Simulation sollen z.B. Komponenten modelliert werden, die in Abhängigkeit eines Taktsignals operieren. Programmiertechnisch wird dazu eine Klasse verwendet, die eine Schnittstelle für den Empfang von 'Taktsignalen' bereitstellt, die in der Simulation durch den Aufruf von Methoden wiedergegeben werden. So können alle Klassen, die Hardwarekomponenten simulieren sollen die durch ein Taktsignal gesteuert werden, durch Vererbung mit der Fähigkeit ausgestattet werden, auf ein Taktsignal per Methodenaufruf zu reagieren.

Die Klasse, die in der Simulation die Schnittstelle zum Empfang von Taktsignalen bereitstellt, ist `tlsWrapper`[8,9]. Der Mechanismus zur Simulation des Zeitverhaltens ist im nächsten Abschnitt dargestellt. Um eine Modularisierung der Simulation zu ermöglichen, stellt die Klasse `tlsWrapper` auch die Möglichkeit zur Verfügung, Zeiger auf Variablen für andere Instanzen von `tlsWrapper` zur Verfügung zu stellen und Zeiger auf Variablen von anderen Instanzen zu erhalten. Dieser Mechanismus wird im übernächsten Abschnitt beschrieben.

[8]wrapper, engl. für 'Hülle', da diese Klasse für jede Klasse, die eine Hardwarekomponente simulieren soll, als Basis dient und somit die 'Hülle' für jede Hardwarekomponente darstellt.

[9]Bezeichner des Programmcodes der Simulation werden zur Kennzeichnung im folgenden in der Schriftart `Typewriter` angegeben.

4.4.2 Realisierung der Simulation des Zeitverhaltens

Um das Zeitverhalten der getakteten Register zu simulieren, ist eine Klasse `tlsClock` vorgesehen, die die Erzeugung und Verteilung eines Taktsignals modelliert. Alle Komponenten der Simulation, die von einem Taktsignal gesteuert werden sollen, registrieren sich mit einem speziellen Funktionsaufruf bei einer Instanz der Klasse `tlsClock`. Diese bildet so eine Liste aller Objekte, die das Taktsignal erhalten sollen. Bei der Simulation der Abarbeitung eines Taktzykluses wird die Instanz von `tlsClock` angewiesen, ein Taktsignal an alle bei ihr registrierten Objekt zu senden. Hierzu traversiert `tlsClock` die Liste der Objekte und ruft bei jedem Objekt eine Methode auf, die die Verarbeitungsschritte während eines Taktsignals umsetzt.
Die Daten, die dem Schema der getakteten Register unterliegen sollen, werden durch die Klasse `tlsRegister` gekapselt. Um das Zeitverhalten der Register wiederzugegeben, werden Wertzuweisungen an die Register zunächst zwischengespeichert. Wird der Wert eines Registers abfragt, so wird vor dem nächsten Taktereignis noch der alte Wert zurückgeliefert. Erst nachdem eine Instanz von `tlsClock` das Taktsignal verteilt hat, wird der neu zugewiesene Wert an den Ausgang kopiert und bei weiteren Wertabfragen zurückgeliefert.

4.4.3 Realisierung der Modularität

In der Hardware können die einzelnen Komponenten Informationen von anderen Komponenten empfangen und an andere senden. Bei der Simulation müssen entsprechend die verschiedenen Komponenten miteinander kommunizieren können. Jede Klasse, die eine Komponente des FLT modellieren soll, wird von der Klasse `tlsWrapper` abgeleitet. Um eine beliebige Strukturierung der Simulation zu ermöglichen, kann eine von `tlsWrapper` geerbte Klasse weitere solche Klassen enthalten, wobei beliebig tief geschachtelt werden kann.
Um den Datenaustausch zwischen den von `tlsWrapper` abgeleiteten Klassen zu ermöglichen, sind die Klassen `tlsRegister` und `tlsInput` vorgesehen. Neben der Simulation des Zeitverhaltens dient die Klasse `tlsRegister` dazu, Parameter innerhalb der Simulation zur Verfügung zu stellen. Jeder Instanz von `tlsRegister` wird dazu eine Zeichenkette zugewiesen, die diese Instanz identifiziert. Mit dieser Zeichenkette können andere Teile der Simulation diesen Parameter mit Hilfe einer Instanz der Klasse `tlsInput` anfordern. Gesteuert wird dieser Mechanismus von einer Instanz der Klasse `tlsController`, bei der alle Instanzen von `tlsRegister` mit ihren identifizierenden Zeichenketten registriert werden. Die Instanzen von `tlsInput` können dann die Adressen der Parameter von `tlsController` erhalten.

4.5 Simulation der Trigger Decision Unit

Mit Hilfe des tls-Frameworks ist eine Simulation der Trigger Decision Unit erstellt worden. Die Information über den internen Aufbau lieferte hierzu eine Simulation der TDU, die in der Programmiersprache C vorliegt [groe98]. Mittels dieser Simulation wurde zunächst ein Datenflußdiagramm der TDU erstellt, welches Grundlage der Beschreibung der TDU im vorhergehenden Kapitel gewesen ist. Eine vollständige Dokumentation vom Entwickler der Trigger Decision Unit lag zum Zeitpunkt der Fertigstellung dieser Arbeit noch nicht vor.

Das tls-Framework erlaubt einen modularen Aufbau der Simulation. Es gibt eine Klasse `tduTDU`, die sämtliche Module der TDU-Simulation zusammenfaßt und eine einheitliche Schnittstelle bietet. Da die Simulation des Zeitverhaltens nicht für jede Fragestellung von Bedeutung ist, wurde eine weitere Klasse `tlsQuickTDU` entwickelt, die eine zu `tduTDU` kompatible Schnittstelle bereitstellt. Die Klasse `tlsQuickTDU` läßt sich innerhalb der Simulation äquivalent zur Klasse `tduTDU` verwenden, wobei sich die Informationen über die internen Zustände nicht abrufen lassen.

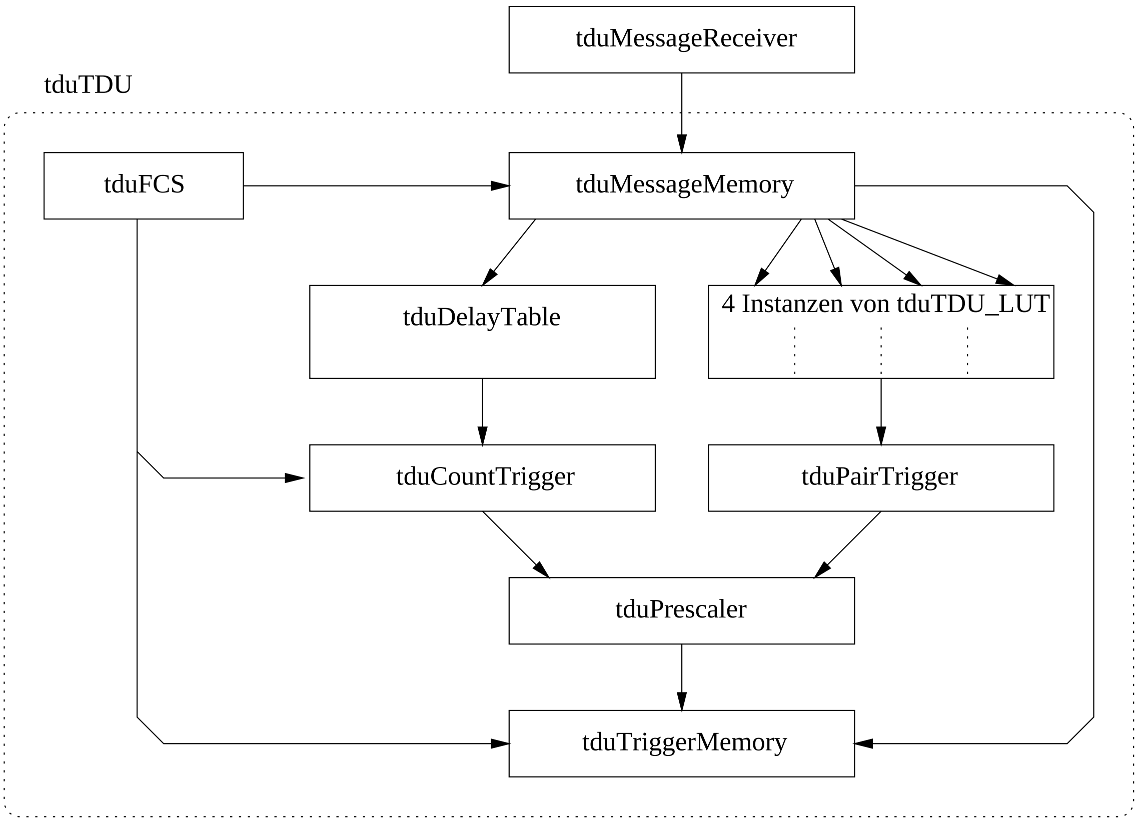

Abbildung 4.4: Innere Strukturierung der Klasse `tduTDU`. Jeder Kasten entspricht einer Instanz der angegebenen Klasse. Logisch zusammengehörige Bereiche der TDU werden durch entsprechende Klassen simuliert, um den Programmcode übersichtlich zu halten. Die Klasse `tduMessageReceiver` simuliert die Koordination von mehreren Datenquellen (in Form von TPUs) zu dem einen Eingang der TDU.

Die Klasse `tduTDU` enthält als interne Module Instanzen von Klassen, die einzelne Bereiche der Trigger Decision Unit simulieren. Eine Übersicht dieser Klassen (siehe Abbildung 4.4) zeigt daher klare Analogien mit der Strukturierung der TDU-Hardware (siehe Abbildung 3.5). Die Implementierung der Simulation der Trigger Decision Unit wird im Anhang A.7 vorgestellt.

4.6 Simulation des ECAL Pretriggers

Um einfache Untersuchungen des First Level Trigger in einer Ausbaustufe vorzunehmen, bei der lediglich die Information des elektromagnetischen Kalorimeters verwendet wird und die Spurrekonstruktion im TFU-Netzwerk übergangen wird, ist eine einfache Simulation des ECAL Pretriggers in dem tls-Framework verwirklicht worden. In der gegenwärtigen Form werden weder die Lookuptables zur Berechnung der Ausgangsparameter noch das interne Zeitverhalten berücksichtigt. Die Klasse `tlsEPT` enthält lediglich eine Schnittstelle zum tls-Framework, um sich in dessen Zeitschema einzufügen. Die Daten können direkt aus sogenannten ARTE-Tabellen gelesen werden. ARTE ist ein im Rahmen der HERA-B Kollaboration entwickeltes Programmpaket, welches zur Erzeugung und Prozessierung von Monte Carlo Ereignissen sowie zur Rekonstruktion von Monte Carlo oder realen Daten dient. Wenn die Simulation zusammen mit ARTE gebunden wird, so werden die Daten des elektromagnetischen Kalorimeters aus der ARTE-Tabelle `HITC` gelesen, welche die Energiedepositionen der einzelnen Kalorimeterzellen enthält. Alternativ können die entsprechenden Daten aus einer Datei gelesen werden, um die Simulation auch ohne Bindung mit ARTE verwenden zu können, was die Ausführungsgeschwindigkeit erhöht.

4.6.1 Der Clusteralgorithmus des ECAL Pretriggers

Die Energien der einzelnen Kalorimeterzellen können wahlweise dem Clusteralgorithmus unterzogen werden, welcher Energieanhäufungen zusammenfaßt und so der Tatsache Rechnung trägt, daß ein in das Kalorimeter einlaufendes Teilchen seine Energie in mehreren benachbarten Kalorimeterzellen deponiert.
Die Daten werden von der Klasse `tlsEPT` in zwei Formen zur Verfügung gestellt. Zum einen wird die Information in Form des 80 bit breiten Datenworts ausgegeben, welche von den Track Finding Units als Spurkeime verwendet werden. So kann diese Pretriggersimulation in Verbindung mit der Simulation des FLT-Netzwerks verwendet werden. Zum anderen werden zur Simulation des FLT ohne TFU-Netzwerk zusätzlich die Impulskomponenten der Spurkandidaten ausgegeben, die direkt von der TDU-Simulation verwendet werden können.

4.7 Anwendungbeispiel des tls-Frameworks

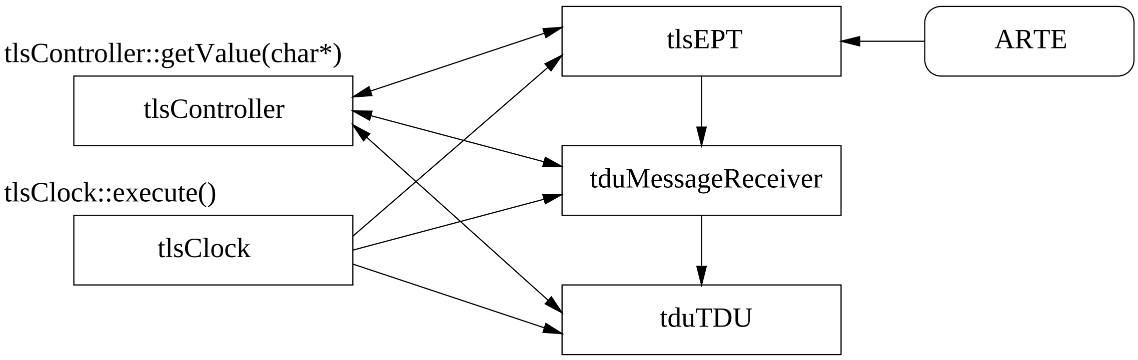

Abbildung 4.5: Beispiel für die Funktionsweise des tls-Frameworks.

An einem Beispiel soll nun kurz die Zusammenarbeit der einzelnen Module der Simulation erläutert werden. Der Aufbau ist in Abb. 4.5 angegeben. Die Simulation des ECAL Pretriggers `tlsEPT` erhält Daten aus dem mitgebundenen ARTE-Paket. Die Ausgangsdaten von `tlsEPT` werden durch die Simulation des Message Receivers `tduMessageReceiver` an die Simulation der Trigger Decision Unit `tduTDU` geleitet.
Darüber hinaus sind je eine Instanz der Klassen `tlsController` und `tlsClock` vorhanden. Bei der Initialisierung der Simulation werden zunächst Instanzen der Klassen `tlsController` und `tlsClock` konstruiert. Bei dem Aufruf der Konstruktoren[10] der Klassen `tlsEPT`, `tduMessageReceiver` und `tduTDU` melden diese ihre internen Instanzen von `tlsRegister` an. Über diese können die Objekte interne Werte nach außen exportieren, also innerhalb der Simulation global verfügbar machen. Anschließend wird die Methode `initializeAllDevices()` von tlsController aufgerufen. Nun werden alle bei `tlsController` angemeldeten Objekte aufgefordert, ihre Eingangsparameter anzufordern. Diese Anforderungen werden anhand der bei `tlsController` registrierten Objekte befriedigt. Sollte sich ein angeforderter Parameter nicht auf ein angemeldetes Objekt abbilden lassen, wird die Simulation mit einer entsprechenden Fehlermeldung abgebrochen. Nach dieser Prozedur sind die Module der Simulation miteinander verbunden und bereit zur Verarbeitung von Daten.
Bei dem Aufruf der Konstruktoren der Klassen, die die einzelnen Bestandteile des zu simulierenden Aufbaus repräsentieren, melden diese sich auch der der Instanz der Klasse `tlsClock` an. Anschließend werden durch Aufruf der Methode `execute()` von `tlsClock` die Daten bei allen angemeldeten Objekten prozessiert und so der Ablauf eines Taktzyklus in der Hardware simuliert.
Um Parameter auszuwerten, die innerhalb eines der bei `tlsController` registrierten Objekte vorliegen, kann die Methode `getValue(char*)` verwendet werden. Ihr wird eine Zeichenkette übergeben, die den gewünschten Parameter bezeichnet. So kann beispielsweise durch den Aufruf

```
controller.getValue("TDU:TriggerMemory:TM_VALID_XVII")
```

der Wert des Parameters `TM_VALID_XVII` ermittelt werden, welcher in der mit `TriggerMemory` bezeichneten Instanz der Klasse `tduTriggerMemory` definiert ist, welche wiederum in der Instanz `TDU` der Klasse `tduTDU` verwendet wird.

[10] In der objektorientierten Programmierung heißen die Funktionen, die bei der Initialisierung einer Klasse aufgerufen werden, Konstruktoren.

Kapitel 5

Einfluß des Binnings in den TDU Lookuptables

5.1 Einleitung

Die Trigger Decision Unit berechnet die invarianten Massen zu allen kombinatorisch möglichen Spurpaarungen eines Ereignisses. Hierzu werden in der Hardware Lookuptable Prozessoren verwendet, die in Abschnitt 3.7.3 vorgestellt worden sind. Entscheidend für eine optimale Präzision bei den Berechnungen in den Lookuptables ist ein geeignet gewähltes Binning, siehe Abschnitt 4.2.5.
In diesem Kapitel soll zunächst beschrieben werden, wie ein geeignetes Binning aufgefunden werden kann. Im Anschluß wird die Präzision der Berechnung durch die Lookuptables mit Hilfe von Monte Carlo Daten von myonischen J/ψ-Zerfällen untersucht.

5.2 Optimierung des Binnings

Es werden zwei Methoden des Binnings verwendet, das lineare und das logarithmische Binning (siehe Abschnitt 4.2.5). Für Parameter, deren Definitionsbereich den Wert 0 beinhaltet oder dessen Definitionsbereich mit einer ungefähr gleichmäßigen Wahrscheinlichkeitsverteilung von den auftretenden Werten abgedeckt wird, wird ein lineares Binning $b_{lin} : [a, b) \to [0, N-1]$ gewählt. Ist $[a, b)$ das zu kodierende Intervall, N die Anzahl der Bins und x der zu kodierende Wert, so wird die Funktion

$$b_{lin}(x) = frac\left((N-1) \cdot \frac{x-a}{b-a}\right) \tag{5.1}$$

verwendet. Zur Dekodierung $r_{lin} : [0, N-1] \to [a, b)$ des durch lineares Binning erhaltenen digitalen Wertes n wird die Funktion

$$r_{lin}(n) = a + \frac{2 \cdot n + 1}{2} \cdot (N-1) \tag{5.2}$$

verwendet. Es wird also das arithmetische Mittel des Intervalls verwendet, welches dem ganzzahligen Wert n zugeordnet ist.
Sollen Parameter kodiert werden, deren Definitionsbereich den Wert 0 nicht beinhaltet und deren Werte mit einer exponentiell abfallenden Wahrscheinlichkeitsverteilung auftreten, so

ist ein logarithmisches Binning $b_{log} : [a, b) \to [0, N-1]$ zweckmäßig. Es wird hierzu die Funktion

$$b_{log}(x) = frac\left((N-1)\frac{\ln(a) - \ln(x)}{\ln(b) - \ln(a)}\right) \tag{5.3}$$

verwendet. Die Dekodierung $r_{log}: [0, N-1] \to [a, b)$ wird durch die Funktion

$$r_{log}(n) = a\frac{e^{\frac{b}{a}\cdot\frac{n}{N-1}} + e^{\frac{b}{a}\cdot\frac{n+1}{N-1}}}{2} \tag{5.4}$$

realisiert.
Die Implementierung des linearen und logarithmischen Binnings erfolgt in dem Programmcode, der mit den Dateien `tlsBinning.h` und `tlsBinning.c` gegeben ist. Im Anhang, Abschnitt A.3, wird der Programmcode näher beschrieben.

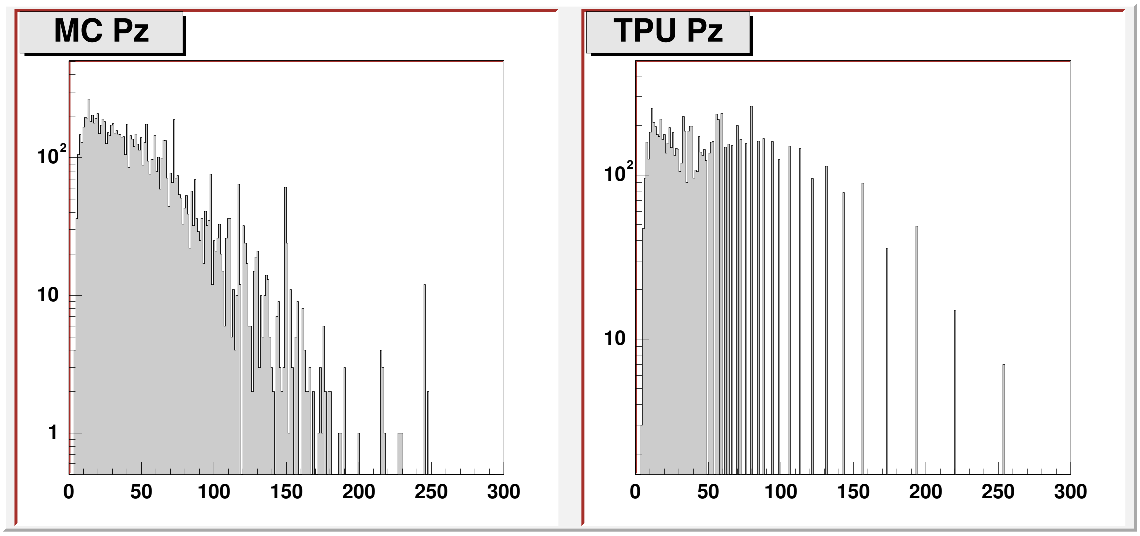

Abbildung 5.1: Histogramm der auftretenden Werte für den Parameter P_z bei myonischen J/ψ-Zerfällen. Links ist die Verteilung der Monte Carlo Impulse zu sehen. Rechts hingegen die Werte, die die Lookuptable Simulation der TPU ausgibt. In letzterem Histogramm ist gut die Quantisierung der Impulswerte durch das Binning der TPU Lookuptables zu erkennen.

Um das optimale Intervall $[a, b)$ für das Binning zu bestimmen, werden die Eingangsdaten für die gesuchten Ereignisse histogrammiert. Hierzu werden Monte Carlo Daten von dem Zerfallskanal $J/\psi \to \mu^+\mu^-$ herangezogen. Diese bieten gegenüber dem Kanal $J/\psi \to e^+e^-$ den Vorteil, daß keine Bremsstrahlungsverluste auftreten, die die zu untersuchenden Effekte überlagern. Neben den direkt von Monte Carlo Generator ermittelten Impulsen werden auch die durch die Lookuptable Simulation der Track Parameter Unit berechneten Impulse verwendet, um einen eventuellen Einfluß der TPU Lookuptables zu berücksichtigen. Anhand der Histogramme kann man das Intervall $[a, b)$ ermitteln, das mit dem Binning kodiert werden soll. So geben die Histogramme in den Abbildungen 5.1 und 5.2 die erforderlichen Definitionsbereiche für die Binningfunktionen $b(x) : [a, b) \to [0, N-1]$ der Eingangsparameter P_z, $\hat{P}_x$ und $\hat{P}_y$ wieder. Die gewählten Definitionsbereiche sind in Tabelle 5.2 angegeben. Das zum Binning gewählten Intervalle $[a, b)$ der weiteren Parameter,

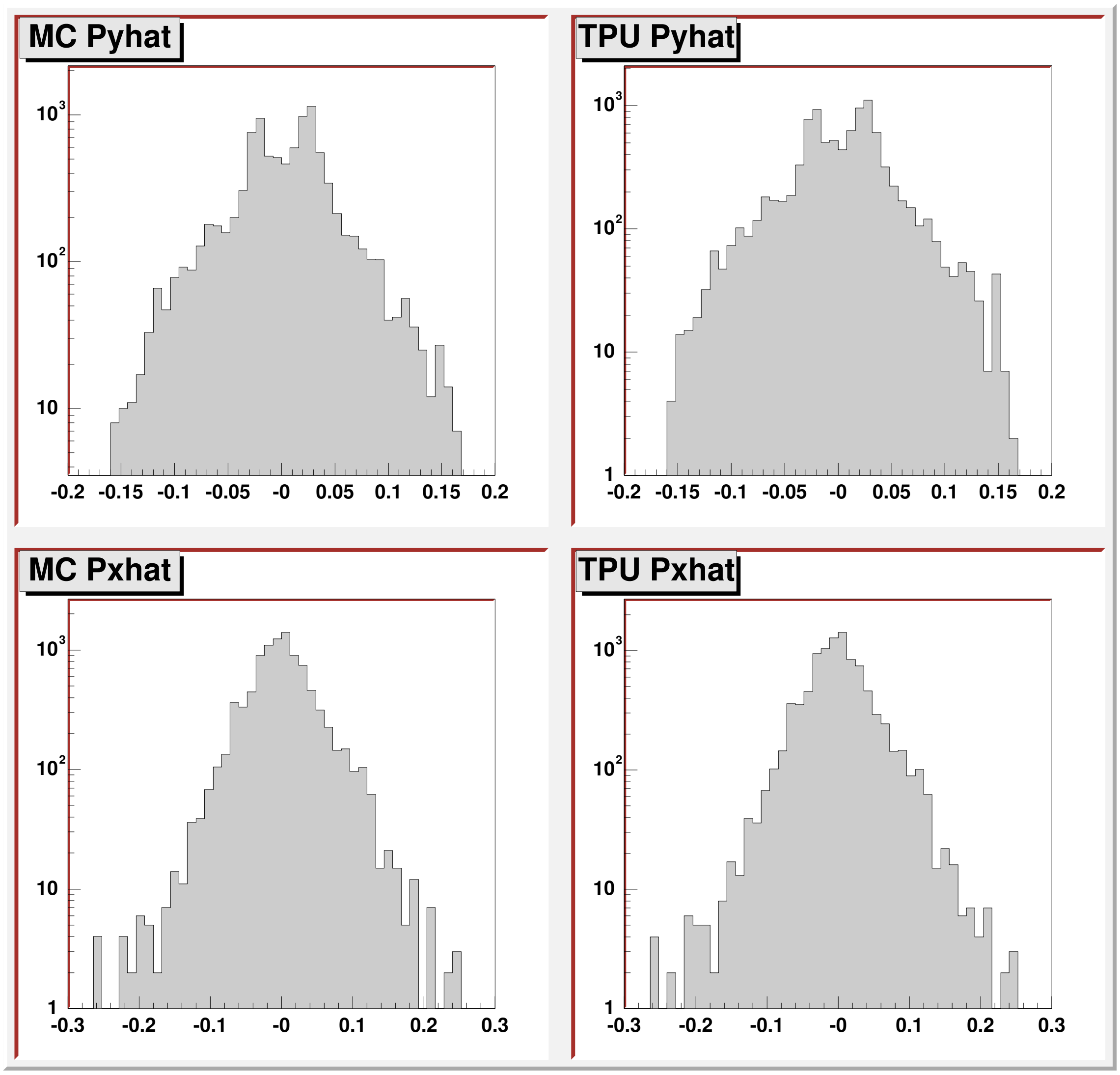

Abbildung 5.2: Histogramme der Werteverteilung für die Parameter $\hat{P}_x$ und $\hat{P}_y$ bei myonischen J/ψ-Zerfällen. Links sind die Werte gemäß den Monte Carlo Impulsen angegeben, rechts die Werte, die die Lookuptable Simulation der TPU liefert.

die in Tabelle 5.2 aufgeführt sind, werden in gleicher Weise bestimmt. Die entsprechenden Histogramme sind hier der Übersichtlichkeit halber nicht aufgeführt. Außerdem ergibt sich der Definitionsbereich der Binningfunktionen eines Parameters auch direkt aus denen der Größen, die zur Berechnung des Parameters herangezogen werden. Wie aus den Histogrammen ersichtlich wird, haben die Lookuptables innerhalb der TPU keinen Einfluß auf das Impulsspektrum am Eingang der TDU.

Es werden hier nur die Werte der Eingangsparameter für das gesuchte Signal $J/\psi \rightarrow \mu^+\mu^-$ berücksichtigt. Durch die Hintergrundereignisse treten auch Werte außerhalb der durch die Signalereignisse gegebenen Bereiche auf. Diese werden durch die in Abschnitt A.3 beschriebenen Binningfunktionen als ungültige Werte interpretiert. Hierzu liefert die Binningfunktion einen speziellen Wert, der die Nichtdefiniertheit des Wertes anzeigt. Die Lookuptables

Parameter	Binning	a	b	Bits
P_z	log	1,0	250,0	7
$\hat{P}_x$	lin	$-0{,}25$	$+0{,}25$	10
$\hat{P}_y$	lin	$-0{,}17$	$+0{,}17$	9
$\hat{P}_t$	log	0,005	0,320	8
$\Delta\hat{P}_x$	lin	0,00	0,30	8
$\Delta\hat{P}_y$	lin	0,00	0,25	8
P_t	log	0,005	80,000	3
E	log	3,0	250,0	7
E_{Prod}	log	6,0	250,0	8
ΔE	lin	0,0	500,0	8
θ	lin	0,00	0,27	8

Tabelle 5.1: Realisierung des Binnings der Parameter, die in den TDU Lookuptables verwendet werden. Die aufgeführten Parameter sind in Abschnitt 3.7.3 vorgestellt worden.

sind so programmiert, daß ebenfalls ein ungültiger Wert ausgegeben wird, sobald einer der Eingangsparameter einen nicht gültigen Wert annimmt. Dieses Verfahren bietet zwei Vorteile:

- Durch eine Minimierung der Intervallbreite $[a, b)$ bei den Binningfunktionen wird die Unsicherheit bei Signalereignissen minimiert.
- Die Interpretation von außerhalb der durch die Signalereignisse gegebenen Intervalle liegenden Werten als ungültige Werte wirkt wie ein Schnitt auf die entsprechenden Parameter und führt so zu einer Hintergrundreduktion.

5.3 Präzision der Massenberechnung in den Lookuptables

Zur Untersuchung der durch die Lookuptables der TDU hervorgerufenen Unsicherheit in der Berechnung der invarianten Masse von Spurpaaren werden die gleichen Monte Carlo Daten mit myonischen J/ψ-Zerfällen verwendet, wie im vorigen Abschnitt 5.2. Die invarianten Massen werden nach drei verschiedenen Methoden berechnet und es wird jeweils die Unsicherheit σ_M in der Masse bestimmt:

- Berechnung mittels der in den Lookuptables implementierten Beziehung

  $$M^2 = 2 \cdot E_1 \cdot E_2 \cdot \left(1 - \cos\left(\sqrt{(\Delta\hat{P}_x)^2 + (\Delta\hat{P}_y)^2}\right)\right) \qquad (5.5)$$

  Diese Beziehung, die in Abschnitt 3.7.3 motiviert worden ist, stellt eine Näherung dar. Daher tritt bereits ohne Binning eine intrinsische Unsicherheit bei der Berechnung der invarianten Masse auf. Um diese zu bestimmen, wird die invariante Masse

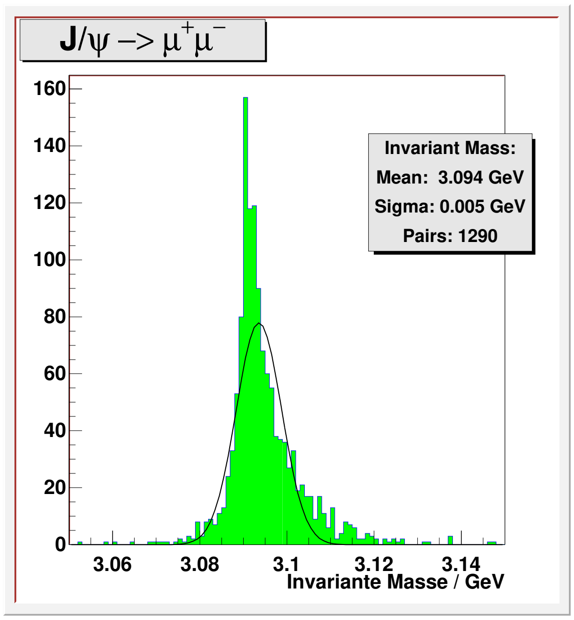

Abbildung 5.3: Massenspektrum bei Verwendung der in der TDU verwendeten Näherung zur Berechnung der invarianten Masse. Hieraus ergibt sich eine Unsicherheit von $\sigma_M \approx$ 5 MeV, die unabhängig vom gewählten Binning auftritt. Die Massenverteilung ist in diesem Fall nichtgaußisch.

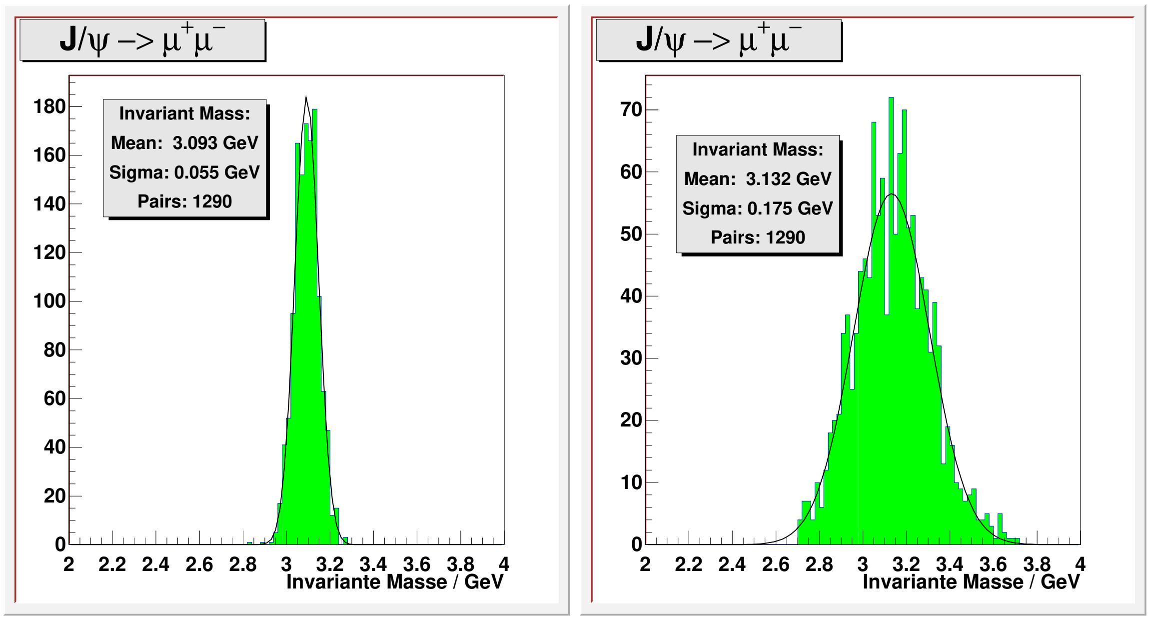

Abbildung 5.4: Das linke Diagramm zeigt das Massenspektrum von Monte Carlo Impulsen, welche mit den Lookuptables der TDU berechnet worden ist. Im rechten Diagramm wurden hingegen die Impulse verwendet, die die Simulation L1SIMU geliefert hat.

unter Verwendung von Gleichung 5.5 aus den vom Monte Carlo Generator gelieferterten Impulsen der Myonen aus dem $J/\psi \to \mu^+\mu^-$-Zerfall berechnet. Das Ergebnis ist in Abbildung 5.3 zu sehen. Die Verteilung der Masse ist nichtgaußisch.

- Zur Untersuchung des Einflusses des Binnings in der TDU wird Gleichung 5.5 unter Verwendung der Lookuptables und des in Abschnitt 5.2 vorgestellten Binnings angewandt. Auch hier werden die Impulse der Myonen des Monte Carlo Generators verwendet. Das linke Diagramm in Abbildung 5.4 gibt daher den Effekt der Näherung in Gleichung 5.5 zuzüglich des Binnings wieder. Die Unsicherheit aufgrund des Binnings ist ungefähr eine Größenordnung höher als die durch die Näherung in der Berechnung, so daß der Einfluß von letzterem keine Rolle spielt.

- Um den Einfluß des Binnings innerhalb der TDU mit dem innerhalb der TPU vergleichen zu können, wird im letzten Schritt die Ausgabe einer Lookuptablesimulation der TPU zur Berechnung der Impulskomponenten von Myonenspuren verwendet, die von der Simulation des TFU-Netzwerks innerhalb von L1SIMU[1] rekonstruiert worden sind. Die Simulation L1SIMU arbeitet ohne Lookuptables, so daß im rechten Diagramm von Abbildung 5.4 der nur Einfluß des Binnings von TPU und TDU auftritt.

Die Ergebnisse sind in Tabelle 5.3 zusammengefaßt. Es wird deutlich, daß die vom Bin-

Binning	σ_M
kein	5 MeV
in TDU	56 MeV
in TPU u. TDU	175 MeV

Tabelle 5.2: Unsicherheiten σ_M in der Berechnung der invarianten Masse in Abhängigkeit vom Binning in TPU und TDU.

ning in der Track Parameter Unit resultierende Unsicherheit größer als die der TDU ist. Dies liegt an den umfangreicheren Berechnungen in der TPU mit Lookuptables im Vergleich zu denen in der TDU. Die Unsicherheit durch die Näherung in Gleichung 5.5 und durch das Binning in der TDU sind daher als unkritisch zu bezeichnen. Es sind Optimierungsmöglichkeiten der Lookuptables der TPU untersucht worden [fla99].
Eine Untersuchung der Unsicherheit bei Berücksichtigung des Binnings im gesamten FLT-Netzwerk ist zum Zeitpunkt der Fertigstellung dieser Arbeit noch nicht vorgenommen worden.

[1] Siehe Abschnitt 4.1.

Kapitel 6

Untersuchungen zur Wahl der Triggerparameter

6.1 J/ψ Trigger ohne Spurrekonstruktion

Während der Wartungsperiode Mai/Juni '99 soll der innere Teil des elektronischen Kalorimeters fast vollständig mit Pretriggerboards ausgestattet werden [vil99]. Darüber hinaus steht ab Anfang August '99 eine TDU mit vier LUT-Prozessoren zur Verfügung. Somit eröffnet sich zum ersten Mal die Möglichkeit, mit dem First Level Trigger einen J/ψ-Trigger zu verwirklichen, der auf die invariante Masse von Teilchenpaaren anspricht.
Da im elektromagnetischen Kalorimeter neben der Ortsinformation auch die Energie eines eingehenden Teilchens bestimmt werden kann, ist es möglich, die kinematischen Parameter allein aus diesen Daten zu berechnen. Die Spurrekonstruktion, die im der in diesem Kapitel betrachteten Aufbauphase noch nicht implementiert ist, hat die Aufgabe, die von den Pretriggern gelieferten Spurparameter zu präzisieren und Hintergrundereignisse herauszufiltern. Letzteres wird erreicht, indem Energieeinträge durch fehlende Koinzidenz in den Spurkammern erkannt und nur vollständig rekonstruierte Spuren angenommen werden. Unter Inkaufnahme von geringerer Präzision und erhöhtem Hintergrund kann jedoch das elektromagnetische Kalorimeter auch allein für den J/ψ-Trigger verwendet werden. Die hierbei zu erwartende Effizienz und Hintergrundunterdrückung soll hier untersucht werden. Diese Ergebnisse sollen mit den zu erwartenden Werten bei Verwendung des Count Triggers verglichen werden, der lediglich auf einzelne Spurkandidaten mit hohem Transversalimpuls reagiert.
Der erforderliche Unterdrückungsfaktor des First Level Trigger ergibt sich aus der Ereignisrate des Experiments und der maximalen Eingangsrate des Second Level Trigger. Der Designwert von 50 kHz für die SLT-Eingangsrate kann in der betrachteten Ausbaustufe aus zweierlei Gründen noch nicht erreicht werden [med99]. Zum einen sind mit 80 Stück noch nicht alle vorgesehenen Farm-PCs installiert. Zum anderen kann der Second Level Trigger noch nicht die endgültig vorgesehenen Algorithmen verwenden. Für den endgültigen Betrieb ist vorgesehen, daß der Second Level Trigger die Spurkandidaten des First Level Trigger übernimmt und lediglich die Spurparameter präzisiert, durch das Feld des Spektrometermagneten in den Bereich des Vertexdetektors projiziert und dort bis zum Vertex verfolgt. Dazu müßte der Second Level Trigger nur auf die Informationen der Bereiche des Detektors zurückgreifen, die durch die Spurkandidaten des First Level Trigger als Regions

of Interest ausgewiesen sind. Da dies im Augenblick noch nicht möglich ist, verwendet der Second Level Trigger zusätzlich Daten des elektromagnetischen Kalorimeters und des gesamten Vertexdetektors. Dies begrenzt durch den hierzu erforderlichen höheren die maximale Eingangsrate des SLT auf momentan (August '99) maximal 5 kHz. Auch wenn die Verwendung des Pair Triggers der Trigger Decision Unit dies überflüssig machen kann, sobald ein hinreichend großer Bereich des Kalorimeters mit Pretriggerboards bestückt ist, steigt die maximale Eingangsrate des SLT nicht deutlich an. Dies liegt an der Auslese des Vertexdetektors, der komplett ausgelesen werden muß, solange nicht Spurkandidaten des FLT berücksichtigt werden können. Erst wenn der First Level Trigger mit Hilfe der Spursuche des TFU-Netzwerks hinreichend präzise Spurparameter an den SLT senden kann, kann der Designwert von 50 kHz an Eingangsrate des SLT erreicht werden. Im folgenden wird daher von einer maximalen SLT-Eingangsrate von 5 kHz ausgegangen, was bei einer Ereignisrate von 10 MHz einen Unterdrückungsfaktor von

$$\frac{10\,\mathrm{MHz}}{5\,\mathrm{kHz}} = 2000$$

für den First Level Trigger erfordert. Eine Ereignisrate von 10 MHz ist typisch für die Datennahme in dieser Phase der Inbetriebnahme des Detektors.
Eine weitere begrenzende Größe bei der Aufnahme von Ereignissen ist die maximale Aufzeichnungsrate, mit der die Daten auf Band gespeichert werden kann. Sie hängt stark von der Anzahl der während einer Datennahme verwendeten Subkomponenten des Detektors ab. Im folgenden wird von einer maximalen Speicherfrequenz von 25 Hz ausgegangen. Die Einflüsse der höheren Triggerstufen wird hier nicht betrachtet. Dies ist eine realistische Betrachtungsweise, da konzeptionell der First Level Trigger die Aufgabe hat, J/ψ-Zerfälle zu finden und damit den folgenden Triggerstufen eine Ausgangsbasis zur weiteren Rekonstruktion zu geben.

6.2 Signal-Hintergrundverhältnis und Signalsignifikanz

In diesem Abschnitt sollen Kriterien vorgestellt werden, mit denen die Leistung eines Triggers adäquat beschreiben werden können. Ein Trigger dient dazu, aus dem Ereignisstrom, den der Detektor liefert, die Ereignisse herauszufiltern, die eine gesuchte Reaktion enthalten. Als Ereignisstrom bezeichnen wir eine Folge von Ereignissen an einer beliebigen Stelle der Auswertungskette. Die gesuchten Ereignisse werden als *Signal* bezeichnet, die Gesamtheit aller Reaktionen, in der auch das Signal auftritt, heißt *Hintergrund*. Das *Signal-Hintergrundverhältnis* S/H eines Ereignisstroms ist der Quotient der Anzahl der gesuchten Signalereignisse und der der Hintergrundereignisse [igo97]. Das Signal-Hintergrundverhältnis des direkt von Detektor kommenden Ereignisstroms, welches neben den inelastischen Hintergrundreaktionen das Signal X enthalte, ergibt sich aus dem totalen Wirkungsquerschnitt des Signals σ_X und dem der inelastischen Ereignisse $\sigma_{inel.}$:

$$\frac{S}{H} = \frac{\sigma_X}{\sigma_{inel.}} \tag{6.1}$$

Ein Trigger dient dazu, das Signal-Hintergrundverhältnis zu erhöhen. Dazu wendet der Trigger geeignete Kriterien auf der Ereignisstrom an und verwirft aufgrund dieser Kriterien bestimmte Ereignisse. Ziel ist es, Signalereignisse möglichst nicht zu verwerfen und dagegen die Hintergrundereignisse zu unterdrücken. Um diesen Vorgang zu quantifizieren, werden die Triggereffizienz ϵ und der Hintergrundunterdrückungsfaktor R definiert. Die Triggereffizienz $\epsilon \in [0, 1]$ ist die Wahrscheinlichkeit für das Akzeptieren eines Signalereignisses und der Unterdrückungsfaktor $R \in [1, \infty[$ ist gleich der reziproken Wahrscheinlichkeit für das Akzeptieren eines Hintergrundereignisses. Betrachten wir nun den Ereignisstrom der von einem Trigger ausgewählten Ereignisse, so ergibt sich folgendes modifiziertes Signal-Hintergrundverhältnis:

$$\frac{S}{H} = \frac{\sigma_X}{\sigma_{inel.}} \cdot \epsilon \cdot R \tag{6.2}$$

Je höher das Produkt $\epsilon \cdot R$ von Triggereffizienz und Hintergrundunterdrückung, desto häufiger tritt die gesuchte Reaktion am Ausgang auf.
Eine weitere Größe zur Beurteilung der Leistung des Triggers ist die *Signalsignifikanz* $S/\sqrt{H}$. Damit ein interessantes Signal klar von den Hintergrundereignissen unterschieden werden kann, müssen sich die Signalereignisse gegenüber den statistischen Schwankungen der Hintergrundereignisse absetzen. Man kann hierbei zwei Fälle unterscheiden, die sich in der maximal verfügbaren Aufzeichnungsfrequenz der Ereignisse unterscheiden. Liegt die maximal verfügbare Aufzeichungsfrequenz stets über der Ausgaberate des Triggers, so paßt sich die Aufzeichungsfrequenz der Ausgabefrequenz an. Wenn hingegen die Ausgaberate des Triggers die maximale Aufzeichungsfrequenz stets übersteigt, müssen einzelne Ereignisse, die den Trigger passiert haben, verworfen werden, um die verfügbare Aufzeichungsfrequenz nicht zu überschreiten. Die Anzahl der primären Ereignisse N_0 und der aufgezeichneten Ereignisse N ergibt sich aus der Meßzeit t und der Ereignisrate ν bzw. Aufzeichnungsfrequenz f gemäß

$$N_0 = \nu \cdot t, \quad N = f \cdot t \tag{6.3}$$

Die beiden Fälle sind:

- Wenn alle Daten des Ereignisstroms, der am Ausgang des Triggers vorliegt, berücksichtigt werden können, hängen die Anzahl von Signalereignissen S und Hintergrundereignissen H von der Anzahl der Ereignisse am Eingang des Triggers N_0 in folgender Weise ab:

  $$S = \frac{\sigma_X}{\sigma_{inel.}} \cdot \epsilon \cdot N_0 \tag{6.4}$$

  $$H = \frac{1}{R} \cdot N_0 \tag{6.5}$$

  Das Signalsignifikanz beträgt in diesem Fall

  $$\frac{S}{\sqrt{H}} = \frac{\sigma_X}{\sigma_{inel.}} \cdot \epsilon \cdot \sqrt{R} \cdot \sqrt{N_0} = \frac{\sigma_X}{\sigma_{inel.}} \cdot \epsilon \cdot \sqrt{R} \cdot \sqrt{\nu \cdot t} \tag{6.6}$$

- In dem Fall, daß die Aufzeichungsfrequenz stets unter der Ausgaberate des Triggers liegt, ist die Anzahl der Signalereignisse abhängig von der der aufgezeichneten Daten N:

  $$S = \frac{\sigma_X}{\sigma_{inel.}} \cdot \epsilon \cdot R \cdot N \tag{6.7}$$

  Die Anzahl der Hintergrundereignisse hingegen ist gleich der Anzahl der aufgezeichneten Daten:

  $$H = N \tag{6.8}$$

Die Signalsignifikanz beträgt somit:

$$\frac{S}{\sqrt{H}} = \frac{\sigma_X}{\sigma_{inel.}} \cdot \epsilon \cdot R \cdot \sqrt{N} = \frac{\sigma_X}{\sigma_{inel.}} \cdot \epsilon \cdot R \cdot \sqrt{f \cdot t} \tag{6.9}$$

Die Unterscheidung hinsichtlich der Aufzeichungsfrequenz ist in den folgenden Betrachtungen von Belang, wenn die erforderliche Meßzeit zur Aufzeichnung eines J/ψ-Signals bestimmt werden soll. In beiden Fällen ist die Signalsignifikanz abhängig von der Wurzel der Meßzeit. So kann die Signalsignifikanz prinzipiell über die Meßzeit beliebig erhöht werden, doch es sind diesem Weg natürlich praktische Grenzen gesetzt.
Im folgenden sollen die besprochenen Parameter für prompte J/ψ, die in ein Elektron-Positron-Paar zerfallen, untersucht werden. Als prompte J/ψ-Ereignisse werden solche bezeichnet, denen keine weiteren Randbedingungen außer dem Auftreten eines J/ψ gefordert werden. Insbesondere werden nicht speziell J/ψ aus dem 'goldenen Zerfall' betrachtet. Bei dem Experiment HERA-B wird ein Wirkungsquerschnitt von $\sigma_{J/\psi} = 350\,\mathrm{nb}$ für prompte J/ψ-Ereignisse und $\sigma_{inel} = 13\,\mathrm{mb}$ für inelastische Ereignisse erwartet. Zusätzlich muß das Verzweigungsverhältnis für den Kanal $J/\psi \to e^+e^-$ berücksichtigt werden, da die J/ψ im Kalorimeter über Elektronen nachgewiesen werden. Dieses Verzweigungsverhältnis beträgt $BR(J/\psi \to e^+e^-) = 6 \cdot 10^{-2}$. Wenn man nur der First Level Trigger zur Selektion von J/ψ-Ereignissen verwendet und von einer maximalen Aufzeichnungsfrequenz f ausgeht, so hängt das Verhältnis $S/\sqrt{H}$ von der Meßzeit t in der folgenden Weise ab:

$$\frac{S}{\sqrt{H}} = \frac{\sigma_{pN \to J/\psi + X} \cdot BR(J/\psi \to e^+e^-)}{\sqrt{\sigma_{inel}}} \cdot \epsilon \cdot R \cdot \sqrt{f \cdot t} \tag{6.10}$$

Dabei sind ϵ und R Triggereffizienz und Unterdrückungsfaktor des FLT. Die erforderliche Meßzeit, um eine gegebene Signalsignifikanz $S/\sqrt{H}$ zu erzielen, ergibt sich dann aus:

$$t = \frac{1}{f} \cdot \left(\frac{S}{\sqrt{H}} \cdot \frac{\sigma_{inel.}}{\sigma_{J/\psi} \cdot BR(J/\psi \to e^+e^-)} \right)^2 \cdot \frac{1}{(\epsilon \cdot R)^2} \tag{6.11}$$

$$= \frac{1}{f} \cdot \left(\frac{S}{\sqrt{H}} \cdot 6{,}2 \cdot 10^5 \right)^2 \cdot \frac{1}{(\epsilon \cdot R)^2} \tag{6.12}$$

6.3 Verwendung eines Triggers auf hohen Transversalimpuls im Tower- und Cluster-Modus des ECAL-Pretrigger

Die einfachste Möglichkeit, Ereignisse mit einem prompten J/ψ zu selektieren, ist die Realisierung eines Triggers auf hohen Transversalimpuls. Hierzu werden einzelne Spuren auf hohen Transversalimpuls geprüft und zur Triggerentscheidung herangezogen. Da das J/ψ mit 3,1 GeV eine Masse besitzt, die groß gegen die der Zerfallsprodukte ist, haben diese einen hohen Transversalimpuls. Das Transversalimpulsspektrum der durch die inelastischen Hintergrundereignisse hervorgerufenen Spuren verläuft hingegen exponentiell fallend [rat99]. Durch Zurückweisung von Ereignissen, die keine Elektronenkandidaten mit einem Transversalimpuls oberhalb einer geeignet gewählten Schwelle aufweisen, ist eine wirksame Unterdrückung des Hintergrunds erreichbar.
Mit dem First Level Trigger ist es möglich, einen solchen Trigger auch ohne das Spurkammersystem zu realisieren, indem lediglich die Orts- und Energieinformation von Treffern im elektromagnetischen Kalorimeter verwendet werden.

Der Verzicht auf die Spursuche führt bei Verwendung des Count Triggers zu folgenden Nachteilen:

- Da die Ablenkung im Feld des Spektrometermagneten nicht bestimmt wird, können Elektronenkandidaten nicht von Positronkandidaten unterschieden werden. Dieser Umstand führt zu einem erhöhten kombinatorischen Hintergrund.
- Photonen werden im vollständigen System von Elektronen dadurch unterschieden, daß sie keine Treffer in den Spurkammern hervorrufen. Bei Verzicht auf die Spursuche im First Level Trigger erhöht sich daher der Hintergrund aufgrund der Photonen, die auch als Elektronkandidaten angesehen werden.

Im folgenden sollen Triggereffizienz ϵ und Hintergrundunterdrückung R für einen Trigger auf hohen Transversalimpuls mittels des First Level Trigger untersucht werden.

6.3.1 Einfluß des Magnetfeldes, der Parameter K_{trig}

Bei abgeschalteten Spektrometermagneten kann man den Transversalimpuls eines Elektronkandidaten im ECAL-Pretrigger allein aus den Daten des Kalorimeters bestimmen. In diesem Fall ist der Betrag des Impulses gleich der Energie, die im Kalorimeter registriert worden ist. Da keine Ablenkung im einem Magnetfeld erfolgt, ist der Impuls darüber hinaus parallel zu der Geraden, die den primären Vertex und den Ort der Energiedeposition im Kalorimeter schneidet. Der Betrag des Transversalimpulses ist dann die Projektion des Impulsvektors auf die Ebene senkrecht zur Protonstrahlrichtung, also

$$p_T = \frac{E}{\sqrt{x^2 + y^2 + z^2}} \cdot \sqrt{x^2 + y^2} \tag{6.13}$$

wobei x, y, z den Ort und E den Betrag einer Energiedeposition im Kalorimeter bezeichnen. Der Term $\sqrt{x^2 + y^2 + z^2}$ entspricht der Entfernung der Energiedeposition vom primären Vertex und $\sqrt{x^2 + y^2}$ der Distanz vom Protonstrahlrohr.
Ist hingegen der Spektrometermagnet eingeschaltet, so ist die Beziehung 6.13 nicht mehr gültig. Um den Einfluß des Magnetfeldes zu berücksichtigen, wird stattdessen der empirisch motivierte Parameter K_{trig} herangezogen [loh94], der die Ablenkung der Elektronen in der Ebene senkrecht zur Magnetfeldrichtung berücksichtigt:

$$K_{trig} = E \cdot \left(\frac{1}{\sqrt{x^2 + y^2}} + \frac{1}{\sqrt{x^2 + |y|^3}} \right)^{-1}$$

Da im folgenden stets von dem Fall des eingeschalteten Spektrometermagneten ausgegangen wird, wird der Transversalimpuls stets in Form der Größe K_{trig} angegeben.

6.3.2 Einfluß des K_{trig}-Cuts im ECAL-Pretrigger auf die Anzahl der Spurkandidaten pro Ereignis

Um die Hintergrundunterdrückung R des Triggers auf hohen Transversalimpuls zu bestimmen, werden Monte Carlo Daten von inelastischen Ereignissen mit eingeschalteten Magneten betrachtet.
Die Information über die im Kalorimeter aufgetretenen Energiedepositionen werden über die ECAL-Pretrigger Boards an den First Level Trigger weitergeleitet. Die geometrische

Akzeptanz ist somit abhängig von den installierten ECAL-Pretrigger Boards. In der Phase der Inbetriebnahme des Detektors sind noch nicht alle ECAL-Pretrigger Boards verfügbar, weshalb verschiedene Ausbaustufen betrachtet werden. Im folgenden werden das gesamte, der mittlere und innere und ausschließlich der innere Teil als drei typische Ausbaustufen betrachtet.
Zur Bestimmung der Anzahl der Spurkandidaten bei gegebenen K_{trig}-Cut werden Monte Carlo Daten von inelastischen Ereignissen mit eingeschalteten Magneten herangezogen[1]. Ein Monte Carlo Ereignis bestehe im folgenden stets aus genau einer Proton-Nukleon-Wechselwirkung, was im Mittel einer Wechselwirkungsrate von 8,5 MHz im Experiment entspricht. Für andere Wechselwirkungsraten müssen entsprechende Skalierungen vorgenommen werden.
Um die Wirkung der ECAL-Pretrigger zu simulieren, werden aus den Monte Carlo Daten die Energiedepositionen im elektromagnetischen Kalorimeter betrachtet. Vor der Wartungsperiode Mai/Juni '99 ist der Pretrigger so programmiert gewesen, daß er lediglich die Energiedeposition in jeder einzelnen Kalorimeterzelle zur Triggerentscheidung benutzt. Der Nachteil dieser Betriebsweise ist, daß systematisch Energie verlorengeht wenn sich die Energiedeposition eines Teilchens über eine Reihe von Kalorimeterzellen erstreckt. Aus dem gleichen Grund kann ein hochenergetisches Teilchen mehrere Trigger in benachbarten Zellen auslösen. Ab Mitte Juli '99 sollen die Pretrigger Boards hingegen so reprogrammiert werden, daß sie auf Cluster, also Anhäufungen von Energiedepositionen in benachbarten Zellen innerhalb eines Kalorimeterbereiches (inneres, mittleres und äußeres Kalorimeter), ansprechen kann. Hierdurch ist eine höhere Präzision zu erwarten, da Energiedepositionen, die sich über einige Kalorimeterzellen erstrecken, korrekt behandelt werden. Im folgenden sollen daher sowohl die Zellen- als auch die Clusterenergie betrachtet werden.

6.3.3 K_{trig}-Spektrum bei prompten J/ψ-Zerfällen

In diesem Abschnitt soll die Triggereffizienz ϵ des Triggers auf hohen Transversalimpuls bei der Suche nach prompten J/ψ-Ereignissen untersucht werden. Dabei werden wie bei der Betrachtung der Hintergrundunterdrückung nur die Daten des elektromagnetischen Kalorimeters verwendet. Als Datenquelle dienen 3000 Monte Carlo Ereignisse, bei denen jeweils ein $J/\psi \to e^+e^-$-Kanal vorhanden ist, um die Reaktion des elektromagnetischen Kalorimeters zu erhalten[2]. Der Spektrometermagnet ist bei den Monte Carlo Ereignissen eingeschaltet.
Es werden die Energiedepositionen in dem elektromagnetischen Kalorimeter betrachtet und wie bei den inelastischen Ereignissen sowohl die Energien der Kalorimeterzellen als auch die Gesamtenergien der Energieanhäufungen (Cluster) über mehrere Zellen unter-

[1] Eine Referenz auf Monte Carlo Daten zu geben ist stets unsicher, da die entsprechenden Daten veralten und daher ersetzt werden oder an eine anderen Stelle kopiert werden können. Außerdem können die Speicherorte nur in einer für das betrachteten Experiment prorietären Form angegeben werden. Dennoch wäre es unklug wegen dieser Gefahr Referenzen auf Monte Carlo Daten gänzlich zu unterschlagen, da man so a priori jeglicher Kontrollmöglichkeit beraubt wäre. Die hier betrachteten Monte Carlo Daten inelastischer Ereignisse befanden im Juni '99 in den Dateien `/acs/mc/gean-01-08-r3/det99b/inelast.g0802.geo99b.001.dst` bis `/acs/mc/gean-01-08-r3/det99b/inelast.g0802.geo99b.048.dst`.

[2] Es werden hierzu folgende Monte Carlo Ereignisse verwendet:
`/acs/mc3/99.0222.mag-0206r2.jp/99.0222.mag-0206r2.jp.0001.dst` bis
`/acs/mc3/99.0222.mag-0206r2.jp/99.0222.mag-0206r2.jp.0003.dst`.

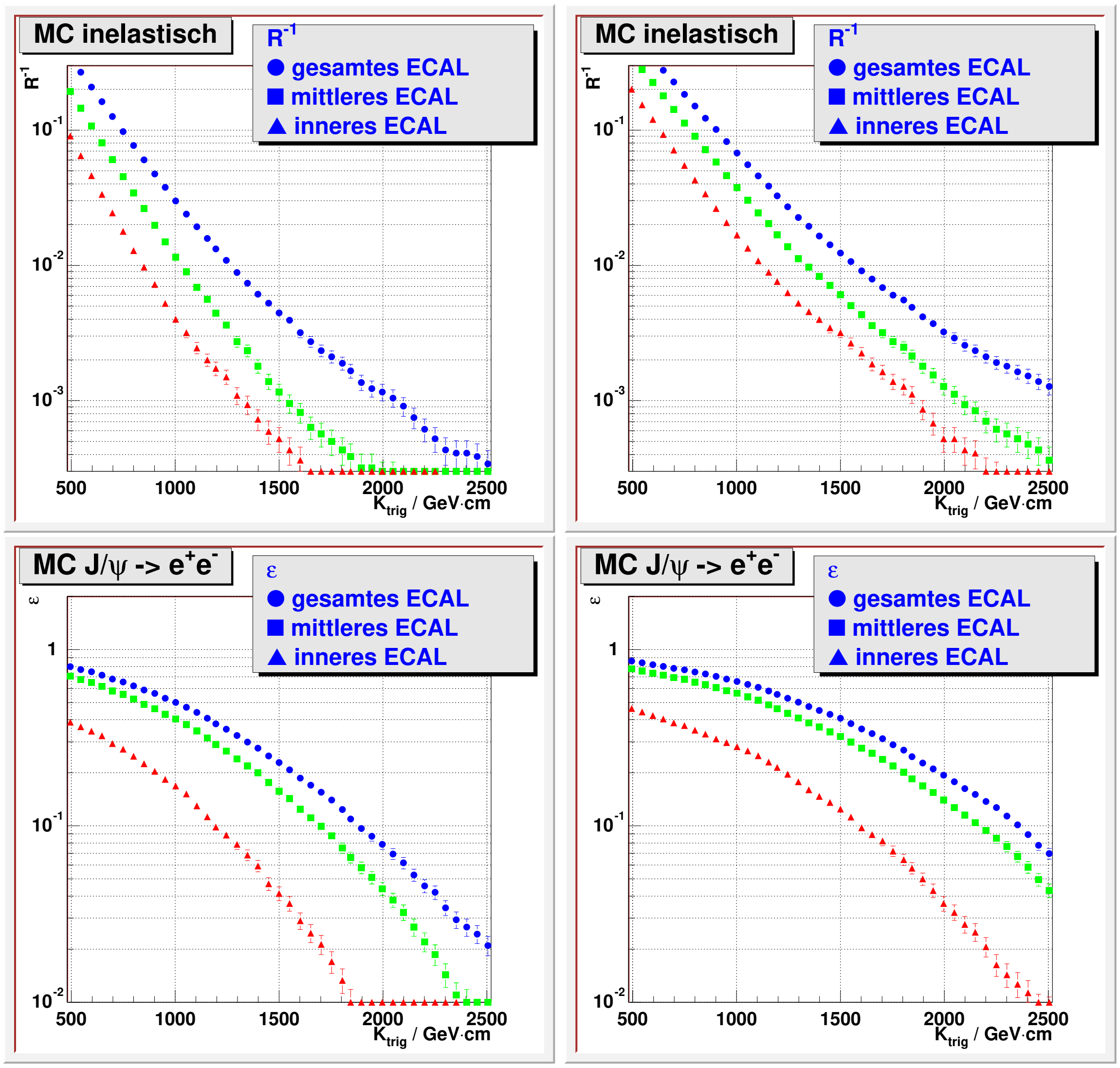

Abbildung 6.1: Reziproke Hintergrundunterdrückung R^{-1} (oben) und Triggereffizienzen ϵ (unten) bei Verwendung des Count Triggers. Die linken Diagramme verwenden die in den Kalorimeterzellen deponierte Energie, während die rechten die Energie nach Anwendung des Clusteralgorithmus berücksichtigen. Es sind die Ergebnisse für die drei betrachteten Ausbaustufe des Kalorimeter-Pretriggers im Clustermodus zusammengefaßt, von oben nach unten innerhalb der Diagramme sind die Werte für das gesamte Kalorimeter, den mittleren und inneren Bereich und ausschließlich den inneren Bereich eingetragen.

sucht. Damit eine Energiedeposition oder ein Cluster in die Auswertung einfließt, wird gefordert, daß sich die Energiedeposition oder der Cluster mit der Monte Carlo-Spur eines Elektrons oder Positrons aus dem $J/\psi \to e^+e^-$-Kanal assoziieren läßt.

6.3.4 Das Signal-Hintergrundverhältnis bei Verwendung des Count Triggers

Bei Berechnung des Signal-Hintergrundverhältnises ist zu beachten, daß nicht alle Ereignisse, die den Count Trigger auslösen, auch Einträge in dem untersuchten Massenbereich besitzen. Es sollen hier nur Massen ab $m > 2{,}0\,\mathrm{GeV}$ berücksichtigt werden. Dieser Schnitt liegt deutlich unter der Masse des gesuchten J/ψ, was bei der hohen Unsicherheit der Elektronenrekonstruktion bei Verzicht auf die Spursuche und Bremsstrahlungskorrektur notwendig ist, um eine befriedigende Effizienz zu erzielen. Daher tragen nicht alle Ereignisse, die von dem Trigger auf hohen Transversalimpuls akzeptiert werden, zum Hintergrund bei. Um das Signal-Hintergrundverhältnis bei Verwendung des Count Triggers bestimmen zu können, ist neben der Hintergrundunterdrückung $R_{K_{trig}}$, die von dem Schnitt auf hohen Transversalimpuls hervorgerufen ist, ein weiterer Unterdrückungsfaktor R_{Masse} zu berücksichtigen. Der Faktor R_{Masse} ist das Verhältnis zwischen der Anzahl von Ereignissen, die den Schnitt auf hohen Transversalimpuls passiert haben und der Anzahl der Einträge in dem untersuchten Massenbereich, die von diesen Ereignissen stammen. Da R_{Masse} von dem Schnitt auf den Transversalimpuls abhängt, wird zunächst die Abhängigkeit des Produkts $\epsilon \cdot R_{K_{trig}}$ von dem Parameter K_{trig} untersucht. Anschließend wird der Faktor R_{Masse} für den Wert von K_trig bestimmt, unter dem das Produkt $\epsilon \cdot R_{K_{trig}}$ maximal ist. Zusammenfassend ergibt sich maximal erreichbare Wert für das Produkt von Triggereffizienz und Hintergrundunterdrückung $\epsilon \cdot R = \epsilon \cdot R_{K_{trig}} \cdot R_{Masse}$.

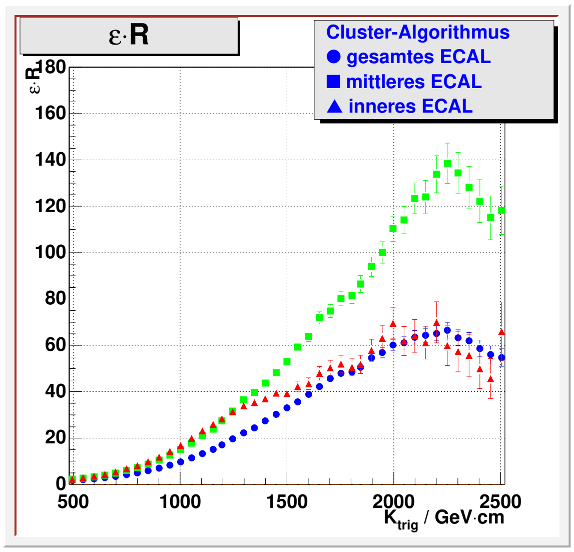

Abbildung 6.2: Produkt von Triggereffizienz und Hintergrundunterdrückung $\epsilon \cdot R$ bei Verwendung des Count Triggers bei prompten $J/\psi \rightarrow e^+e^-$ Ereignissen. Es sind die Ergebnisse für die geometrische Akzeptanz des gesamten, des inneren und mittleren und ausschließlich des inneren Kalorimeters angegeben.

Im ersten Schritt wird das Maximum des Produkts $\epsilon \cdot R_{K_{trig}}$ unter Variation des Parameters K_{trig} bestimmt, was dem Signal-Hintergrundverhältnis ohne Berücksichtigung des betrachteten Massenbereichs entspricht. Hierzu werden die in den Abschnitten 6.3.2 und 6.3.3 verwendeten Monte Carlo Daten verwendet, um den Faktor $\epsilon \cdot R_{K_{trig}}$ in Abhängigkeit von der Schnitt auf den Transversalimpuls K_{trig} zu bestimmen. Die Werte für die

Triggereffizienz ϵ und Hintergrundunterdrückung $R_{K_{trig}}$ aufgrund des Schnittes auf hohen Transversalimpuls werden für K_{trig} im Bereich von 500 GeV · cm bis 2500 GeV · cm bestimmt. Die Werte für das Produkt $\epsilon \cdot R_{K_{trig}}$ sind in Tabelle 6.2 aufgetragen. Es ergibt sich für die drei betrachteten geometrischen Akzeptanzen jeweils ein Maximum bei $K_{trig} = 2200$ GeV · cm. Auffallend ist, das für den Fall des inneren und mittleren Kalorimeters das Produkt $\epsilon \cdot R_{K_{trig}}$ mehr als den doppelten Wert als bei dem Fall des gesamten und inneren Kalorimeters erreicht. Dieser Effekt wird weiter unten in Abschnitt 6.5 betrachtet.

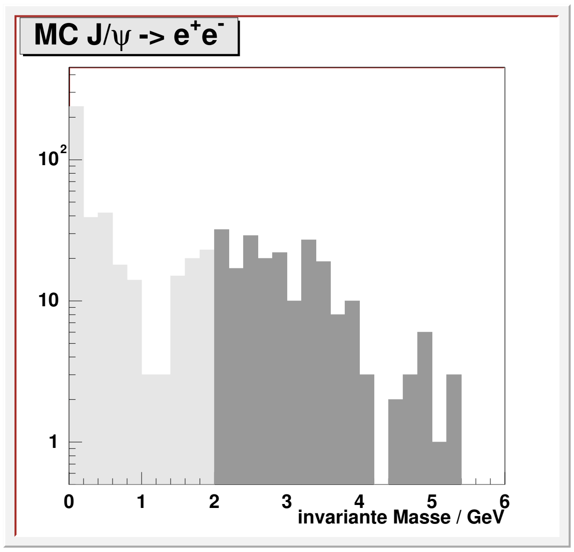

Abbildung 6.3: Massenspektrum von Ereignissen, die den Count Trigger bei einer Schwelle des Transversalimpulses von $K_{trig} > 2200$ GeV · cm passiert haben. Es wurde die geometrische Akzeptanz des gesamten Kalorimeters betrachtet.

Im zweiten Schritt werden die Unterdrückungsfaktoren R_{Masse} in die Betrachtung eingebracht. Hierzu werden die invarianten Massen der durch die beiden Monte Carlo Datensätze gegebenen Elektronenspurkeime berechnet und nur die Beiträge berücksichtigt, bei denen mindestens eine Energiedepositionen mit einem K_{trig} von mehr als 2200 GeV · cm vorliegt. Auf diese Weise erhält man die Massenspektren der Ereignisse, die den Count Trigger passiert haben. Dies wird separat für die drei betrachteten geometrischen Akzeptanzen durchgeführt. In Abbildung 6.3 ist beispielhaft das Spektrum für die Akzeptanz des gesamten Kalorimeters zu sehen. Die Häufung der Einträge für Massen nahe Null resultiert aus in der Fläche breit verteilten Energiedepositionen, die von dem ECAL Pretrigger als mehrere unabhängige Spurkandidaten aufgefaßt werden. Es sind in der TPU und TDU des First Level Trigger Filter vorgesehen, die doppelte Spuren unterdrücken sollen. Doch sind diese für den kompletten FLT mit Spursuche im TFU-Netzwerk vorgesehen, und reagieren daher nur auf Spurpaare mit deutlich kleineren Differenzen, als sie durch den eben beschriebenen Effekt auftreten. Das Verhältnis R_{Masse} von allen Einträgen in den Massenspektren zu denen, die 2 GeV überschreiten, ist in Tabelle 6.5 angegeben.
Die erforderliche Meßzeit, um eine Signalsignifikanz von $3\,\sigma$[3] zu erhalten, kann nun aus dem Produkt $\epsilon \cdot R_{K_{trig}} \cdot R_{Masse}$ unter Verwendung der Beziehung 6.12 berechnet werden.

[3] Die Angabe $3\,\sigma$ bedeutet, daß die Anzahl der Signaleinträge die statistische Unsicherheit der Anzahl der Hintergrundeinträge (σ_{inel}) um den Faktor drei übersteigen soll.

Ausbaustufe	$\epsilon \cdot R_{Ktrig}$	R_{Masse}	$\epsilon \cdot R_{Ktrig} \cdot R_{Masse}$	t/Tage
gesamter ECAL	65	4,7	306	17,1
mittlerer und inner ECAL	135	6,3	850	2,2
innerer ECAL	65	6,8	442	8,2

Tabelle 6.1: Erforderliche Meßzeit zur Gewinnung eines $3\,\sigma$ $J/\psi \to e^+e^-$-Signals bei Verwendung des Count Triggers. Die Zeit ist die effektive Meßzeit, ohne Berücksichtigung der Maschineneffizienz.

Die Resultate sind in Tabelle 6.5 aufgeführt. Hierbei ist eine Aufzeichnungsrate von $f =$ 25 Hz zugrunde gelegt worden. Auffallend ist, daß die erforderliche Meßzeit für einen $3\,\sigma$-Effekt nicht mit der geometrischen Akzeptanz sinkt, sondern für den Fall des inneren und mittleren Kalorimeters minimal wird. Dies hängt damit zusammen, daß die Elektronen aus dem gesuchten $J/\psi \to e^+e^-$-Zerfall bevorzugt im Bereich kleiner Winkel auftreten. Die aus den inelastischen Hintergrundereignissen resultierenden Energiedepositionen, die auch von Photonen herrühren, verteilen sich hingegen über den gesamten Kalorimeterbereich. Dies führt dazu, daß mit wachsender geometrischer Akzeptanz der Hintergrund stärker als das Signal ansteigt und somit das Produkt $\epsilon \cdot R$ sinkt. Dieser Zusammenhang wird in Abschnitt 6.5 genauer betrachtet.

6.4 Verwendung des Pair Triggers ohne Spurrekonstruktion

Die Trigger Decision Unit bietet neben dem Count Trigger die Option, mit dem Pair Trigger auf Zerfälle von Teilchen in Lepton-, Photon- oder Hadronpaare zu reagieren. Die Trigger Decision Unit kombiniert die Spuren eines Ereignisses und berechnet jeweils die invariante Masse von Spuren gleicher Teilchensorte, um so anhand der Masse das Mutterteilchen identifizieren und eine Triggerentscheidung treffen zu können. Hierzu muß ein Ereignis, welches den Trigger auslösen soll, mindestens zwei Spurkandidaten in einem Pretrigger auslösen. In obiger Betrachtung des Count Triggers führte hingegen bereits das Auftreten einer Spur mit hohem Transversalimpuls zu einer positiven Triggerentscheidung. Die Schnitte auf den Transversalimpuls müssen daher für vergleichbare Effizienz und Hintergrundunterdrückung bei Verwendung des Pair Triggers niedriger liegen als beim Count Trigger.

6.4.1 Berücksichtigung des Spektrometermagneten

Im elektromagnetischen Kalorimeter werden Ort und Energie eines eintreffenden Teilchens registriert. Da die Elektronen und Positronen im Feld des Spektrometermagneten je nach Ladung in unterschiedliche Richtungen abgelenkt werden, kann aus dem Ort und der Energiedeposition nicht in eindeutiger Weise auf den Impuls geschlossen werden. Es gibt je nach Ladung des Teilchens zwei verschiedene Lösungen für den Impuls in Abhängigkeit von Ort und Energie der Energiedeposition.
Das Feld des Spektrometermagneten führt zu einer Krümmung der Trajektorien. Verlängert man die Spurrichtung der Trajektorien hinter dem Spektrometermagneten in die Ebene des Wechselwirkungspunktes, so erscheint der Wechselwirkungspunkt aufgrund der Ablenkung im Magnetfeld um den Betrag

$$\Delta x = \frac{q}{|q|} \cdot \frac{300\,\mathrm{cm} \cdot \mathrm{GeV}}{E} \tag{6.14}$$

verschoben, wobei q die Ladung und E die Energie des Spurkandidaten ist, siehe Abbildung 6.4. Ohne Spursuche können die beiden Ladungen nicht unterschieden werden. Die Tatsache, daß es sich bei den interessanten Spurkandidaten um Teilchen mit hohem Impuls handelt, kann verwendet werden, um eine Hypothese über die Ladung eines Spurkandidaten zu erhalten. Bei den gesuchten Energiedepositionen aus dem Kanal $J/\psi \rightarrow e^+e^-$ ist die Wahrscheinlichkeit höher, daß ein Teilchen mit hoher Energie leicht vom Strahlrohr weg als ein Teilchen niedrigerer Energie stärker zum Strahlrohr hin gelenkt wird. Dies liegt an der höheren Spurdichte in der Nähe des Protonenstrahlrohrs, siehe Abbildung 6.4 [rat99]. Der ECAL Pretrigger liefert daher Spurkandidaten, die vom Protonenstrahlrohr weggelenkten Trajektorien entsprechen. Die Impulskomponenten $P_{x,y,z}$, die in der Track Parameter Unit aus den Spurkandidaten des ECAL Pretriggers berechnet werden, hängen also von dem Ort x, y, z und der Energie E einer Energiedeposition im Kalorimeter in folgender Weise zusammen:

$$P_z = E \cdot \frac{x - \frac{x}{|x|}\frac{300\,\mathrm{cm} \cdot \mathrm{GeV}}{E}}{\sqrt{x^2 + y^2 + z^2}} \tag{6.15}$$

$$P_y = E \cdot \frac{y}{\sqrt{x^2 + y^2 + z^2}} \tag{6.16}$$

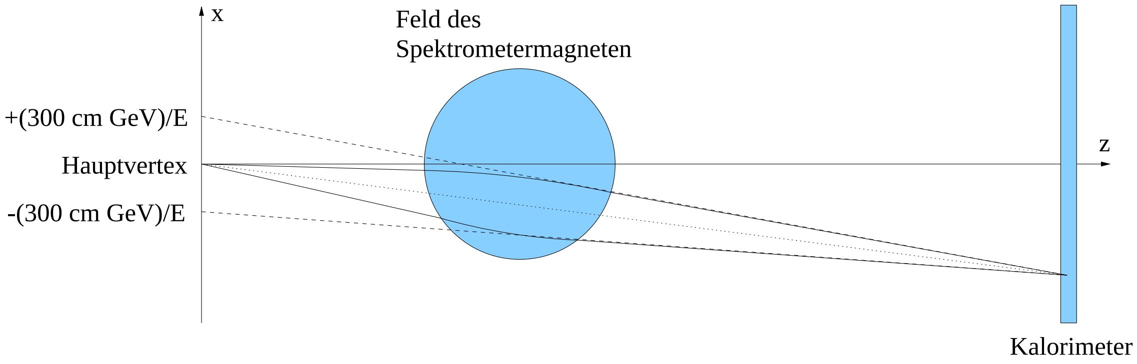

Abbildung 6.4: Ablenkung der Elektronenspuren im Spektrometermagneten. Es sind die beiden möglichen Trajektorien eingezeichnet, die eine gegebene Energiedeposition an einem gegeben Ort im Kalorimeter hervorrufen können.

$$P_z = E \cdot \frac{z}{\sqrt{x^2 + y^2 + z^2}} \tag{6.17}$$

6.4.2 Hintergrundunterdrückung bei Verwendung des Pair Triggers

Um die zu erwartenden Hintergrundunterdrückung R bei Verwendung des Pair Triggers zu bestimmen, werden 44000 inelastische Monte Carlo Ereignisse betrachtet. Die Ereignisse werden dazu mittels der schnellen TDU Simulation `tlsQuickTDU` prozessiert, die die invarianten Massen aus den kombinatorisch möglichen Spurpaaren bildet, die eine bestimmte K_{trig} -Schwelle überschritten haben. So kann die Hintergrundunterdrückung in Abhängigkeit von der K_{trig}-Schwelle bestimmt werden. Da des Transversalimpulsspektrum von inelastischen Ereignissen exponentiell fallend verläuft, ist wie im Fall der Untergrundunterdrückung bei dem Trigger auf hohen Transversalimpuls eine Unterdrückungsrate zu erwarten, die sich beschreiben läßt durch

$$R(K_{trig}) = a \cdot e^{b \cdot K_{trig}}$$

Die Größen a und b wurden mit 44000 Monte Carlo Ereignissen für den gesamten, den mittleren und inneren Teil sowie für den inneren Teil des Kalorimeters ermittelt. Die Ergebnisse sind in Abbildung 6.5 und Tabelle 6.4.2 angegeben.

Ausbaustufe	$a \cdot 10^{-5}$	$b \cdot 10^3 / GeV cm$
ganzer ECAL	$1{,}27 \pm 0{,}10$	$3{,}97 \pm 0{,}11$
mittlerer ECAL	$3{,}17 \pm 0{,}40$	$4{,}01 \pm 0{,}18$
innerer ECAL	42 ± 30	$4{,}02 \pm 1{,}06$

Tabelle 6.2: Häufigkeit für das Auslösen eines Pairtrigger pro Monte Carlo Ereignis. Es wurden jeweils 44000 Ereignisse mit je einer Proton-Nukleon Wechselwirkung prozessiert und der Clusteralgorithmus verwendet.

Die Hintergrundunterdrückung des Pair Triggers ist bei gleichem effektiver als die des Count Triggers. Dies erklärt sich unter anderem aus der Tatsache, daß mindestens zwei

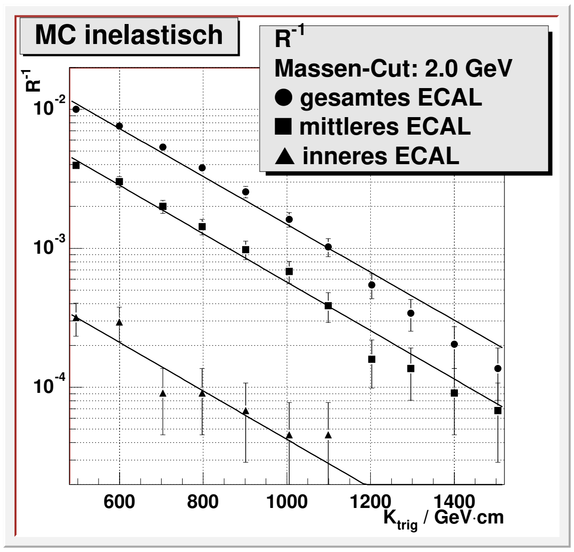

Abbildung 6.5: Unterdrückung R von inelastischen Ereignissen durch den Pair Trigger der TDU bei Verwendung des Cluster-Algorithmus. Es wurde eine Wechselwirkung pro Ereignis verwendet. Der Massen-Cut wurde auf 2 GeV gesetzt und K_{trig} über einen Bereich von 500 GeV $\cdot$ cm bis 1500 GeV $\cdot$ cm variiert. Die oberen, mittleren und unteren Einträge entsprechen dem ganzen, dem inneren und mittlerem und nur dem inneren Kalorimeter.

Spurkandidaten vorhanden sein müssen, um eine invariante Masse berechnen zu können. Bei Verwendung des gesamten Kalorimeters ist der geforderte Unterdrückungsfaktor von 2000 ab einem K_{trig}-Cut von 700 GeV $\cdot$ cm erreicht. Bei geringeren Abdeckungen des Kalorimeters durch Pretriggerboards ist die maximale Eingangsrate von 5 kHz des Second Level Trigger kein begrenzender Faktor mehr (s. Abschnitt 6.1).

6.4.3 Triggereffizienz bei Verwendung des Pair Triggers

Die Triggereffizienz des Pair Triggers für prompte J/ψ-Zerfälle wird mit 3000 Monte Carlo Ereignissen untersucht, die den Zerfall $J/\psi \rightarrow e^+e^-$ enthalten. Auch hier werden die beiden Programmieroptionen des ECAL Pretriggers berücksichtigt, entweder lediglich die Zellenenergie zu betrachten oder die Energie eines Cluster zu verwenden. Analog zum vorangehenden Abschnitt werden die Spurkandidaten eines Ereignisses der schnellen TDU-Simulation `tlsQuickTDU` zugeführt. Dabei werden nur die Energiedepositionen im Kalorimeter berücksichtigt, die sich auf ein Elektron des $J/\psi \rightarrow e^+e^-$-Zerfalls zurückführen lassen. So wird vermieden, daß Elektronen aus Hintergrundreaktionen zu einer irrtümlich zu hohen Effizienz führen. In Abbildung 6.6 ist die Massenverteilung zu sehen, die sich aus der Verwendung der Spurkandidaten mit $K_{trig} \geq 1000\,\mathrm{GeV}\cdot\mathrm{cm}$ ergibt. Die Einträge, die in der schnellen TDU-Simulation einen Trigger ausgelöst haben, sind gestrichelt umrandet. Wie in Abbildung 6.6 zu sehen ist, liegt die Triggereffizienz ϵ bei Verwendung des Clusteralgorithmus etwa doppelt so hoch als bei Verwendung der Zellenenergie. Wie schon in Abbildung 6.3 treten Einträge mit Masse nahe Null gehäuft auf. Dies ist wiederum die Folge von in der Fläche breit verteilten Energiedepositionen. Sie werden von der Pretriggerlogik des Kalorimeters in verschiedene Spurkandidaten aufgeteilt. Dieser Vorgang wird durch die hier nicht betrachtete Spursuche des TFU-Netzwerks unterdrückt, welches sicherstellt, daß nur Spurkandidaten berücksichtigt werden, die physikalischen Spuren korrespondieren.

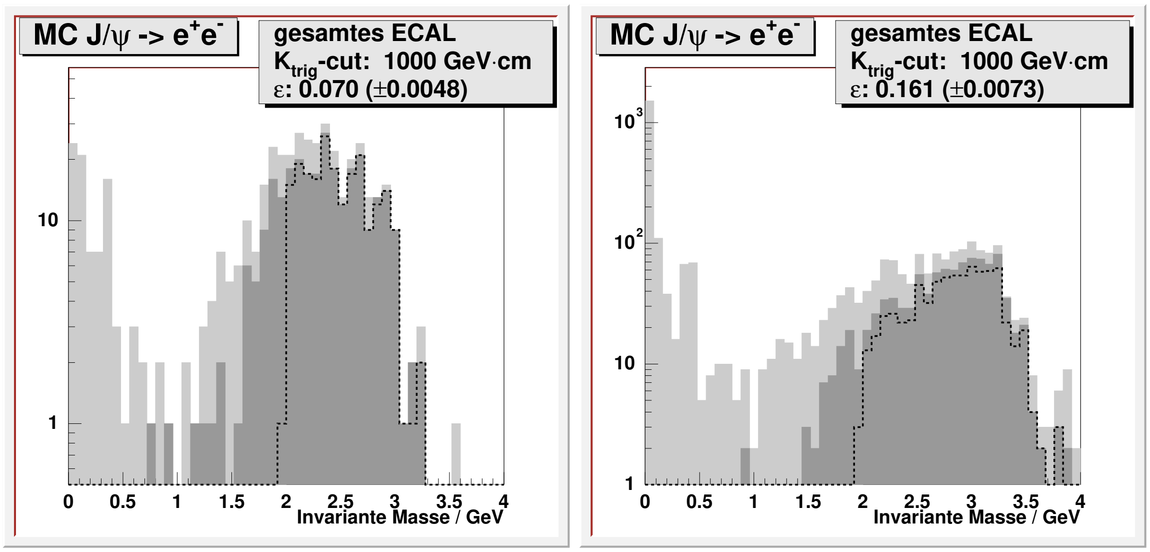

Abbildung 6.6: Massenspektrum prompter $J/\psi \rightarrow e^+e^-$Ereignisse für Spurkandidaten mit $K_{trig} \geq 1000\,\mathrm{GeV}\cdot\mathrm{cm}$. Die helle ausgefüllte Fläche gibt die Einträge aller Spurkandidaten an, die dunkle Fläche dagegen die nur Spurkandidaten von Energiedepositionen, die sich auf ein Elektron des $J/\psi \rightarrow e^+e^-$-Zerfalls zurückführen lassen. Die gestrichelte Linie bezeichnet die von der TDU-Simulation akzeptierten Einträge. Das linke Histogramm verwendet die Energien der Kalorimeterzellen, das rechte die durch den Clusteralgorithmus ermittelten Spurkandidaten.

In der Abbildung 6.7 ist die Triggereffizienz für die drei betrachteten Ausbaustufen der ECAL Pretriggerboards dargestellt. Hier ist deutlich zu sehen, daß die Effizienz durch die

Verwendung des Clusteralgorithmus um gut das doppelte ansteigt. Der Grund hierfür ist die Verteilung der von einem Elektron im Kalorimeter deponierten Energie über mehrere Kalorimeterzellen. Betrachtet man nur einzelne Zellen, so werden systematisch zu geringe Energien für die Berechnung der invarianten Masse herangezogen, was sich in einem Effizienzverlust äußert.

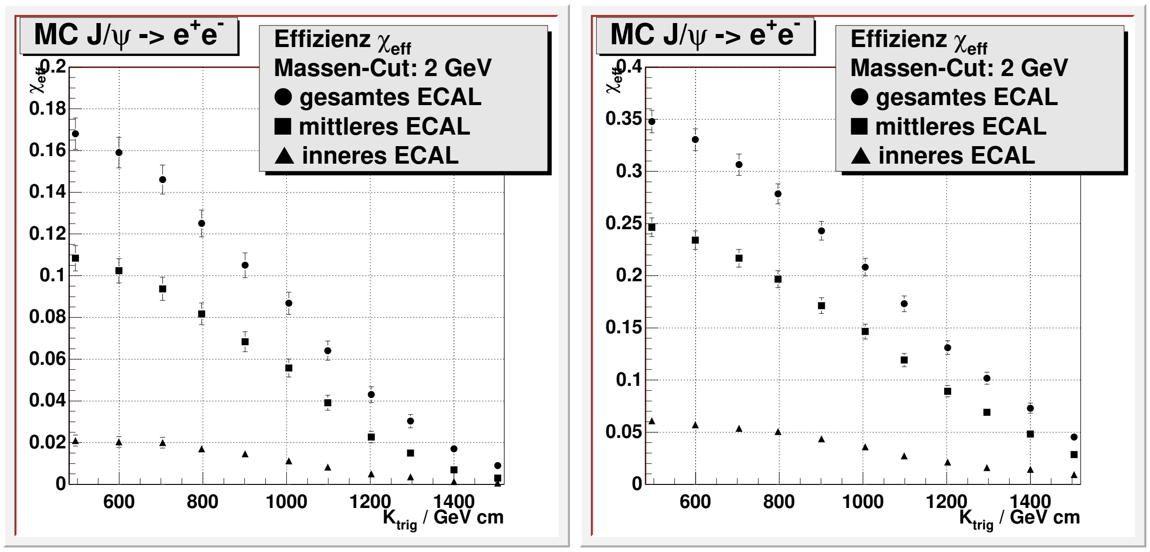

Abbildung 6.7: Triggereffizienz des Pair Triggers ϵ bei prompten $J/\psi \rightarrow e^+e^-$ Ereignissen. Es wurde eine Wechselwirkung pro Ereignis betrachtet. Das linke Diagramm verwendet direkt die Energien der einzelnen Kalorimeterzellen während im rechten Diagramm der Clusteralgorithmus des ECAL-Pretriggers herangezogen wird. Der Massen-Cut wurde auf 2,0 GeV gesetzt und K_{trig} über einen Bereich von 500 GeVcm bis 1500 GeVcm variiert. Die oberen, mittleren und unteren Einträge entsprechen dem ganzen, dem inneren und mittlerem und nur dem inneren Kalorimeter.

6.4.4 Signal-Hintergrundverhältnis bei Verwendung des Pair Triggers

Bei Verwendung des Pair Triggers zum Auslösen einer Triggerentscheidung müssen notwendig zwei Spurkandidaten zu einem Ereignis gefunden werden. Die Wahrscheinlichkeit, daß ein Hintergrundereignis zur einer positiven Triggerentscheidung führt, ist entsprechend geringer als im Fall des Count Triggers. Um dennoch mit einer begrenzten Anzahl von inelastischen Monte Carlo Ereignissen eine Aussage über das Signal-Hintergrundverhältnis zu erhalten, wurde die Triggereffizienz der prompten $J/\psi \to e^+e^-$-Ereignisse mit der Unterdrückungsrate in Beziehung gesetzt, die sich aus der Relation

$$R(K_{trig}) = a \cdot e^{b \cdot K_{trig}}$$

ergibt (s. Abschnitt 6.4.2). Dabei wurden die in Tabelle ?? angegebenen Werte für a und b gemäß des betrachteten Akzeptanzbereiches des Kalorimeters verwendet.

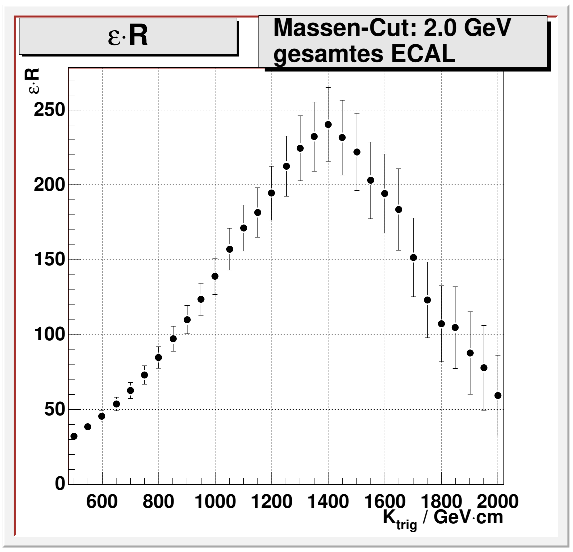

Abbildung 6.8: Das Produkt $\epsilon \cdot R$ bei Verwendung des Pair Triggers.

Ausbaustufe	$\epsilon \cdot R$	t/Tage
gesamter ECAL	240	27,8
mittlerer und inner ECAL	420	9,1
innerer ECAL	1500	0,7

Tabelle 6.3:

In Tabelle 6.5 sind die ermittelten Werte für das maximal erreichbaren Wert des Produkts $\epsilon \cdot R$. Weiterhin sind die erforderlichen Meßzeiten angegeben, die für eine Signalsignifikanz von $3\,\sigma$ erforderlich sind. Die erforderliche Meßzeit steigt mit größerer geometrischen Akzeptanz an. Dies liegt an dem mit dem Akzeptanzbereich wachsenden Hintergrund. Bei den Hintergrundereignissen ist die Wahrscheinlichkeit für das Auslösen eines Triggers

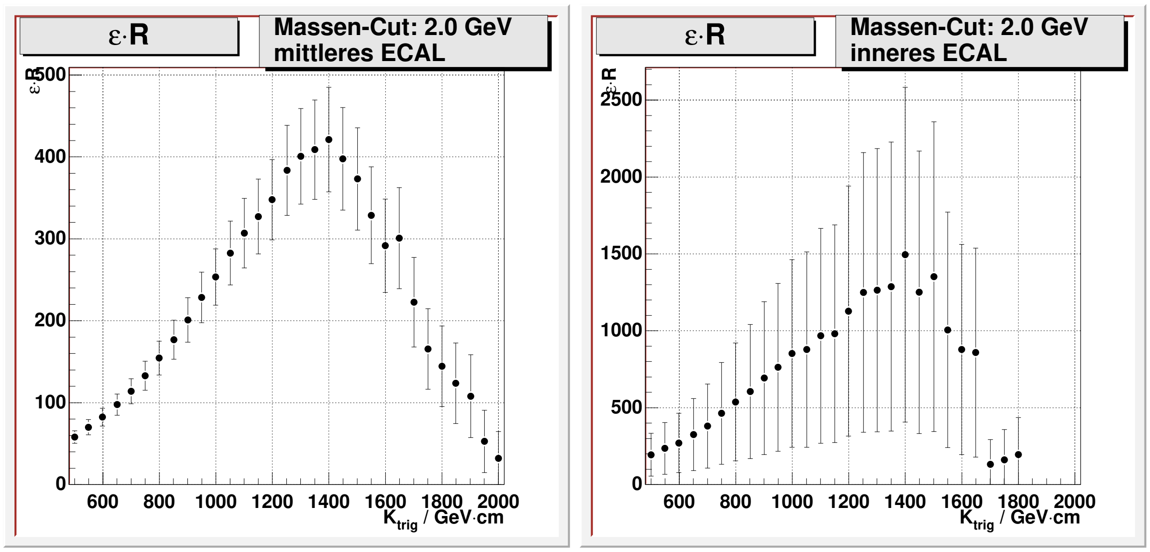

Abbildung 6.9: Das Produkt $\epsilon \cdot R$ bei Verwendung des Pair Triggers.

abhängig vom Quadrat der geometrischen Akzeptanz, da mindestens zwei Spurkandidaten gefunden werden müssen und keine Korrelation zwischen den Spurkandidaten besteht. Die Signalereignisse hingegen sind korreliert, so daß die Abhängigkeit von der geometrischen Akzeptanz in diesem Fall nicht so stark ist.

6.5 Interpretation der Monte Carlo-Untersuchungen

In diesem Abschnitt sollen die Ergebnisse dieses Kapitels zusammengefaßt werden. Vergleicht man die Ergebnisse für die erforderliche Laufzeit zur Erzielung eines $3\,\sigma$-Effekts in den Tabellen und , so zeigt sich:

- Die kürzeste Meßzeit für eine Signalsignifikanz von $3\,\sigma$ erhält man mit dem Pair Trigger und bei ausschließlicher Verwendung des inneren Kalorimeters.

- Die kürzeste Meßzeit bei Verwendung des Count Triggers ergibt sich mit dem inneren und mittleren Kalorimeter.

Das Überraschende an diesem Ergebnis ist, daß eine höhere geometrische Akzeptanz nicht zu einer Verkürzung der erforderlichen Meßzeit führt. Dies liegt an dem Umstand, daß eine vergrößerte geometrische Akzeptanz nicht nur die Triggereffizienz ϵ erhöht, sondern auch die Hintergrundunterdrückung R verringert. Die Energiedepositionen von Signal- und Hintergrundereignissen verteilen sich über das Kalorimeter nicht in der gleichen Weise. Dadurch ist das Produkt $\epsilon \cdot R$ abhängig vom betrachteten geometrischen Bereich.
Dieses Verhalten kann bei Betrachtung des im Kalorimeter gemessenen Energieflusses qualitativ verstanden werden. Hierzu werden die Monte Carlo Datensätze herangezogen, die in obiger Untersuchung verwendet worden sind. Dies sind neben inelastischen Hintergrundereignissen Zerfälle von direkt am primären Wechselwirkungspunkt erzeugten J/ψ, die in ein Elektron-Positron Paar zerfallen. Trägt man die im Kalorimeter pro Fläche gemessene Energie gegen den Abstand der Energiedeposition vom Protonenstrahlrohr auf, so

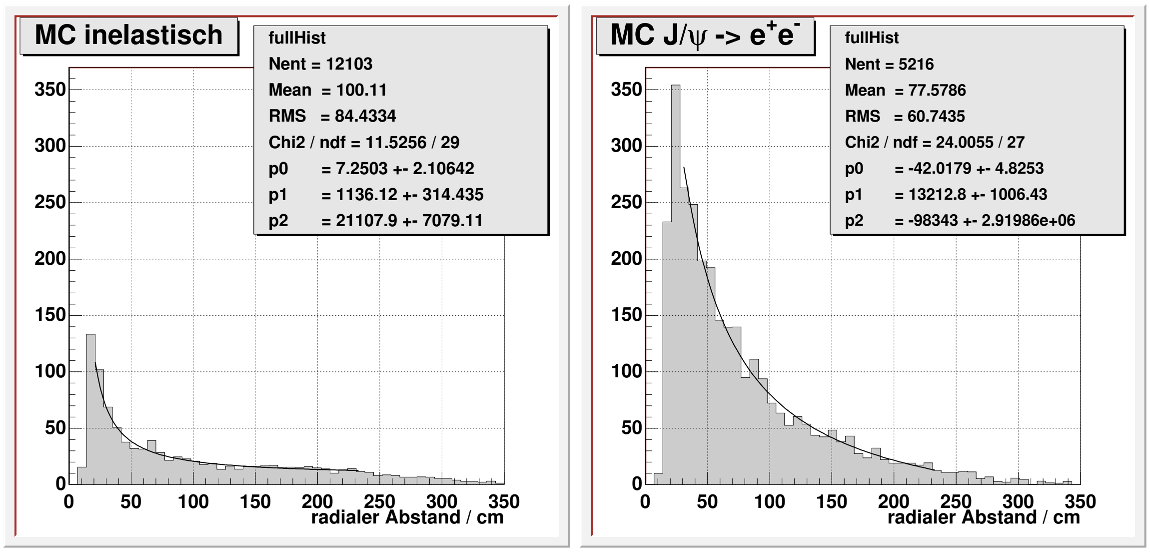

Abbildung 6.10: Im Kalorimeter deponierte Energie pro Fläche in willkürlichen Einheiten. Es ist die Abhängigkeit vom radialen Abstand vom Protonenstrahlrohr angegeben. Das linke Diagramm zeigt die vollständige Energiedeposition pro inelastischer Wechselwirkung. Im rechten Diagramm sind nur die Energien der Elektronen des $J/\psi \to e^+e^-$-Zerfalls berücksichtigt worden. Es sind in beiden Diagrammen Anpassungen an die Funktion $p_0 + p_1\frac{1}{r} + p_2\frac{1}{r^2}$ vorgenommen worden.

erhält man die in Abbildung 6.10 aufgetragenen Zusammenhänge. Der in den Diagrammen sichtbare abrupte Abfall bei Abständen unter 20 cm ist auf das Protonenstrahlrohr zurückzuführen. Beide Histogramme sind an die Funktion $p_0+p_1\frac{1}{r}+p_2\frac{1}{r^2}$ angepaßt worden. Die Werte für die Parameter $p_{0\ldots2}$ sind in Abbildung 6.10 enthalten. Es werden hierbei nur Histogrammeinträge im Bereich $25\,\mathrm{cm} < r < 234\,\mathrm{cm}$ berücksichtigt, in dem weder Protonenstrahlrohr noch äußere Begrenzung die Akzeptanz des Kalorimeters begrenzen.
Aus den Diagrammen wird deutlich, daß für die Hintergrundereignisse die untersuchte Abhängigkeit bei Abständen von $r > 100\,\mathrm{cm}$ relativ flach verläuft. Auch im äußeren Bereich des Kalorimeters treten Energiedepositionen auf, die von Hintergrundereignissen hervorgerufen werden. Für die von dem Kanal $J/\psi \to e^+e^-$ hervorgerufenen Energiedepositionen liegt hingegen eine stärkere Abhängigkeit vom radialen Abstand vor. Die Elektronen des gesuchten Zerfalls bedecken bevorzugt den inneren Teil des Kalorimeters und treffen im äußeren Bereich vergleichsweise selten auf.
Für das hier in diesem Kapitel untersuchte Signal-Hintergrundverhältnis ergibt sich hieraus folgendes Bild. Im inneren Bereich des Kalorimeters ist der durch ein Signalereignis hervorgerufene Signalfluß deutlich höher als der durch ein Hintergrundereignis erzeugte. Dieses Verhältnis wird bei wachsenden Abstand vom Protonenstrahlrohr zunehmend ungünstiger. Dies resultiert in einem geringeren Signal-Hintergrundverhältnis bei Einbeziehung äußerer Bereiche des Kalorimeters, wie es in den Tabellen und zum Ausdruck gebracht wird.
Es sei nochmals bemerkt, daß dieses Verhalten aus der alleinigen Verwendung des Kalorimeters zur Auffindung von Spurkandidaten für den First Level Trigger resultiert. Bei Einbeziehung der Spursuche im TFU-Netzwerk wird der in den hier untersuchten Szenarien auftretende Hintergrund signifikant unterdrückt werden.

Zusammenfassung und Ausblick

Im Rahmen dieser Diplomarbeit sind die Grundlagen für eine Simulation der ersten Triggerstufe des HERA-B Experiments mit Berücksichtigung des Zeitverhaltens gelegt worden. Zum einen ist ein programmiertechnischer Rahmen (Framework) in der Programmiersprache C++ entwickelt worden, welcher Hilfsmittel für die Simulation des Zeitverhaltens und der Verbindung verschiedener Module bereitstellt. Mit diesem Framework ist eine Simulation der Trigger Decision Unit des FLT entwickelt worden, welche die Funktionsfähigkeit des Frameworks demonstriert. Somit ist die Grundlage für eine Simulation des gesamten First Level Trigger auf Taktebene realisiert worden.
Eine Untersuchung des Einflusses des Binnings in den Lookuptables in der TDU hat gezeigt, daß die vom Binning hervorgerufenen Präzisionsverluste tolerierbar sind. Die an dieser Stelle auftretenden Unsicherheiten liegen deutlich unter denen, die durch die übrigen Komponenten des FLT hervorgerufen werden, namentlich die der Spursuche im TFU-Netzwerk und der Berechnungen in der TPU.
Mit dieser Simulation der TDU sind Triggereffizienzen untersucht worden, die bei der Ausbaustufe der ersten Triggerstufe zum Zeitpunkt der Fertigstellung dieser Arbeit (ab August '99) zu erwarten sind. Ein wesentliches Ergebnis dieser Untersuchung ist, daß das Signal-Hintergrundverhältnis bei Verwendung des Count Triggers und des Pair Triggers der Trigger Decision Unit stark von der geometrischen Akzeptanz abhängt, wenn nur die Daten des elektromagnetischen Kalorimeters verwendet werden und die Spursuche nicht implementiert ist. Da die inelastischen Ereignisse im gesamten Kalorimeter gleichmäßig Energiedepositionen hervorrufen, verringert eine höhere geometrische Akzeptanz die Hintergrundunterdrückung. Die Energiedepositionen, die von dem gesuchten Signal $J/\psi \rightarrow e^+e^-$ hervorgerufen werden, treten hingegen am intensivsten im inneren Bereich des Kalorimeters auf, so daß die Effizienz zum Erkennen dieses Signals mit der geometrischen Akzeptanz nur wenig zunimmt. Dies führt dazu, daß bei Verzicht auf die Spursuche im TFU-Netzwerk das Signal-Hintergrundverhältnis für den Count Trigger maximal wird, wenn nur der innere und mittlere Teil des Kalorimeters verwendet werden. Für den Pair Trigger, der die Berechnung invarianter Massen von Spurkandidaten aus dem Kalorimeter für seine Triggerentscheidung zugrunde legt, ist das Signal-Hintergrundverhältnis maximal bei ausschließlicher Verwendung des inneren Kalorimeters. Bei Einbeziehung der Spursuche in dem TFU-Netzwerk des FLT wird der Hintergrund im Kalorimeter wirksamer unterdrückt. Die Energiedepositionen, die von Photonen stammen und daher keine Koinzidenzen in den Spurkammern hervorrufen, werden von dem TFU-Netzwerk zurückgewiesen.
Die nächsten Schritte in der Fertigstellung einer Simulation der Elektronik des FLT auf Taktebene bestehen in der Integration der Track Finding Units und der Track Parameter Units sowie der Myon- und High-p_T-Pretrigger in die Simulation. Es wird dann möglich

sein, die einzelnen Simulationsmodule analog zur Hardware miteinander beliebig zu kombinieren, um verschiedene Ausbaustufen der ersten Triggerstufe und Triggerszenarien zu untersuchen.

Anhang A

Übersicht des bereitgestellten Programmcodes

A.1 Einleitung

In diesem Kapitel wird ein kurzer Überblick der im Rahmen der Diplomarbeit entwickelten Software gegeben werden. Der überwiegende Teil des Codes ist in der Programmiersprache C++ formuliert. Es wurde stets versucht, die Möglichkeiten auszuschöpfen, die C++ zur objektorientierten Programmierung bietet. Bei der Beschreibung der Programme werden daher Begriffe der objektorientierten Programmierung verwendet. Eine verständliche Einführung in die objektorientierte Programmierung bietet beispielsweise [jos94]. Die angegebenen Quelltexte sind mit dem AFS-Dateisystem weltweit unter
`/afs/desy.de/user/m/mino/public/`
verfügbar. Der unter dem Punkt 'Ordner' in den folgenden Abschnitten angegebene Pfad bezieht sich relativ auf den Startpunkt
`/afs/desy.de/user/m/mino/`.

A.2 Simulation des ECAL Pretriggers

Auf den Wunsch der Physikgruppe von HERA-B ist eine einfache Simulation des ECAL Pretriggers entwickelt worden, das in das ARTE-Paket CARE integriert worden ist. ARTE ist das bei HERA-B entwickelte Softwarewerkzeug zur Erzeugung und Prozessierung von Monte Carlo-Daten und zur Untersuchung von realen Daten.
Ordner: `mino/public/EcalPretrigger/`

`EcalPretrigger.h EcalPretrigger.C ECPT.inc kaecpt.C`

Die in den Dateien `EcalPretrigger.h` und `EcalPretrigger.C` implementierte statische Klasse `EcalPretrigger` greift auf die ARTE Tabelle HITC zu, in der die Energiedepositionen der einzelnen Kalorimeterzellen gespeichert sind. Die Dateien `ECPT.inc` und `kaecpt.C` bieten eine Schnittstelle zu ARTE, so daß die Funktionen der Pretriggersimulation direkt im interaktiven Modus von ARTE aufgerufen werden können oder in einer Skriptdatei verwendet werden können.
Zur Initialisierung der Simulation dient das Kommando

`ECPT 'init'` *magnetOn cut geoMode*
Dabei werden die Parameter *magnetOn, cut* und *geoMode* übergeben. Durch *magnetOn* wird entschieden, ob der Transversalimpuls unter Einbeziehung des Spektrometermagneten bestimmt werden soll. Für $magnetOn = 0$ wird p_T, für $magnetOn = 1$ der Parameter K_{trig} gemäß Abschnitt 6.3.1 zur Bestimmung des Transversalimpulses einer Energiedeposition im Kalorimeter verwendet. Als Schwelle auf den so spezifizierten Parameter wird der Wert *cut* herangezogen. Mit dem Wert von *geoMode* kann die geometrische Akzeptanz eingestellt werden. Für $geoMode = 0$ wird die volle geometrische Akzeptanz des gesamten Kalorimeters berücksichtigt. Bei der Angabe $geoMode = 1$ wird die geometrische Akzeptanz verwendet, die durch eine Reihe von rechteckigen Bereichen in der Kalorimeterebene definiert wird. Hierzu dient das Kommando
`ECPT 'rect'` *x1 y1 x2 y2*
durch das ein rechteckiger Bereich mit den Eckkoordinaten $(x1, y1)$ und $(x2, y2)$ angegeben wird. Es können beliebig viele solcher Bereiche angegeben werden. Als geometrische Akzeptanz wird dann die Vereinigungsmenge aller spezifizierten Bereiche verwendet. Weiterhin kann mit $geoMode = 2$ die geometrische Akzeptanz durch die installierten Pretriggerboards des ECAL Pretriggers angegeben werden. Dazu werden mit dem Kommando
`ECPT 'fdid'` *id*
die Identifikationsnummern *id* der zu berücksichtigen Pretriggerboards angegeben.
Um die Anzahl der Kalorimeterzellen zu erhalten, die bei einem Ereignis die spezifizierte Schwelle des Transversalimpulses überschritten haben, kann das Kommando
`ECPT 'tows'`
verwendet werden. Die gleiche Information unter Berücksichtigung des Clusteralgorithmus wird mit dem Kommando
`ECPT 'clus'`
angefordert. Der hierbei verwendete Clusteralgorithmus ist in Abschnitt 3.3.1 beschrieben worden.
Auf die Simulation kann auch aus dem Programmcode der Analysesoftware zugegriffen werden. Mit der statischen Methode
`int EcalPretrigger::processTowers()`
wird die Anzahl der Kalorimeterzellen ermittelt, die die spezifizierte Transversalimpulsschwelle erreichen. Die Methode
`int EcalPretrigger::processCluster()`
gibt die entsprechende Anzahl nach Anwendung des Clusteralgorithmus an.

A.3 Funktionen zum Binning der Lookuptables der TDU

Das Verfahren des Binnings, also des Kodierens eines kontinuierlichen Intervalls in eine diskrete Menge ist in Abschnitt 4.2.5 und Kapitel 5 behandelt worden. Hier sollen die Funktionen vorgestellt werden, mit denen das Binning in der Simulation durchgeführt worden ist. Die in diesem Abschnitt vorgestellten Quelltexte sind im Gegensatz zu allen anderen in diesem Kapitel vorgestellten Codes nicht in C++ sondern in der Programmiersprache C geschrieben. Daher können sie nicht nur in der Simulation, sondern auch direkt auf der TDU-Platine in dem dort vorhandenen Mikroprozessor aufgeführt werden, für den nur ein C-Compiler zur Verfügung steht.
Ordner: `mino/public/TDU_online/`

`TDU_Binning.h TDU_Binning.c`

Das für die TDU Lookuptables verwendete Binning ist in folgenden Funktionen implementiert. Dabei ist stets $[a, b)$ das zu kodierende Intervall und N die Anzahl der bits, die zum Binning zur Verfügung stellen. Die Kodierungsfunktionen `tlsLinBin` und `tlsLogBin` liefern für den Fall $x \notin [a, b)$ den Rückgabewert $2^N - 1$, der einen undefinierten Wert bezeichnet. Wird umgekehrt eine Dekodierungsfunktion `tlsLinFloat` oder `tlsLogFloat` mit dem Wert $2^N - 1$ aufgerufen, so wird die in `tlsBinning.h` definierte Konstante `double tbxFLOAT_INF` zurückgegeben. Alle hier aufgerufenen Funktionen arbeiten mit maximal $N \leq 32$ bit. Ein Wert $N > 32$ ebenso wie der Fall $a \geq b$ führt zu einer Fehlermeldung und zum Abbruch des Programms. Das gleiche Verhalten zeigen auch die Dekodierungsfunktionen `tlsLinFloat` und `tlsLogFloat`, wenn sie mit einem kodierten Wert von $n \geq 2^N$ aufgerufen werden.

```
unsigned long tlsLinBin(double x, double a, double b, unsigned long N)
```

Diese Funktion führt eine Kodierung des Parameters x nach dem linearen Binning $b_{lin} : [a, b) \rightarrow [0, N - 1]$ durch.

```
unsigned long tlsLinFloat(unsigned long n, double a, double b,
                          unsigned long N)
```

`tlsLinFloat` liefert den dekodierten Wert für den kodierten Parameter n zurück.

```
unsigned long tlsLogBin(double x, double a, double b, unsigned long N)
```

Das logarithmische Binning $b_{log} : [a, b) \rightarrow [0, N-1]$ wird von dieser Funktion implementiert.

```
unsigned long tlsLogFloat(unsigned long n, double a, double b,
                          unsigned long N)
```

Diese Funktion realisiert die Dekodierung $r_{log} : [0, N - 1] \rightarrow [a, b)$.

`tduLUTParameter.h`

Dieser Quellcode enthält Definitionen von Konstanten, die den in Tabelle 5.2 aufgeführten Binningparametern entsprechen.

`TDU_Code_Lib.h TDU_Code_Lib.c`

In diesen Dateien werden die Binningfunktionen für die Parameter definiert, die in den TDU Lookuptables verwendet werden. Dies geschieht mit Hilfe der in den Dateien `TDU_Binning.h` `TDU_Binning.c` definierten Binningfunktionen und den in der Datei `tduLUTParameter.h` definierten Parametern.

A.4 Das tlsFramework

Die in diesem Abschnitt vorgestellten Klassen bilden den programmiertechnischen Rahmen oder Framework zur Simulation von digitaler Elektronik in taktgesteuerter Pipeline-Architektur, siehe Kapitel 4. Die zentrale Rolle hierbei spielen die Klassen `tlsClock` und `tlsController`, die das zeitliche Verhalten bzw. die Verteilung der Information innerhalb der Simulation steuern.
Ordner: `mino/public/tls_src/tls/tlsFRAMEWORK/`

`tlsBasics.h tlsBasics.cc`

In diesen Dateien werden die Klasse `tlsClock` und die abstrakte Klasse `tlsLogObject` implementiert. Sie dienen zur Modellierung des zeitlichen Verhaltens der Hardware. Das Taktsignal wird dabei von einer zentralen Instanz der Klasse `tlsClock` an Instanzen der Klasse `tlsLogObject` oder abgeleiteter Klassen weitergegeben.
Hierzu müssen zunächst alle Instanzen der Klasse `tlsLogObject` oder abgeleiteter Klassen, die ein Taktsignal in der Simulation erhalten sollen, bei der zentralen Instanz mit dem Aufruf
`tlsClock::addClockListener(tlsLogObject*` *logobject*`)`
registriert werden. Der Zeiger *logobject* verweise dazu auf das zu registrierende Objekt. Das Taktsignal besteht in der Simulation aus zwei Methodenaufrufen, die der steigenden und fallenden Impulsflanke des physikalischen Taktsignals in der Hardware entsprechen. Hierzu sind die Methoden
`tlsClock::rise()`
und
`tlsClock::fall()`
vorgesehen, die bei allen registrierten Instanzen von `tlsLogObject` die Methoden
`tlsLogObject::onClockRise()`
beziehungsweise
`tlsLogObject::onClockFall()`
aufrufen. Hierzu sind die Methoden `onClockRise()` und `onClockFall()` in der Klasse `tlsLogObject` als virtuelle Funktionen implementiert. Es werden also stets die Implementationen von `onClockRise()` und `onClockFall()` der von `tlsLogObject` abgeleiteten Klassen aufgerufen um ein spezifisches Verhalten bei Erhalten der Taktereignisse zu ermöglichen. Die Klasse `tlsClock` bietet weiterhin die Methode
`tlsClock::execute()`
die die Funktionen von `tlsClock::rise()` und `tlsClock::fall()` vereinigt.

`tlsController.h tlsController.cc`

In diesen Dateien wird die Klasse `tlsController` implementiert, die neben `tlsClock` die zentrale Steuerung der Simulation übernimmt. Jede in der Simulation vorkommende Komponente, die durch eine von `tlsWrapper` (siehe Erläuterungen zu `tlsDevice.h` und `tlsDevice.cc`) abgeleitete Klasse dargestellt wird, registriert sich bei einer zentralen Instanz der Klasse `tlsController`. Dabei wird eine Zeichenkette übergeben, die die Instanz der von `tlsWrapper` abgeleiteten Klasse eindeutig identifiziert und bei dem Aufruf dessen Konstruktors angegeben wird. Anhand dieser Bezeichner kann über die zentrale Instanz

von `tlsController` auf die in der Simulation vorkommenden Komponenten zugegriffen werden. Dies wird für den Informationsaustausch der Komponenten untereinander verwendet. Soll eine eine von `tlsWrapper` abgeleitete Klasse B Parameter der (ebenfalls von `tlsWrapper` abgeleiteten) Klasse A verwenden, so muß Klasse A den entsprechenden Parameter 'exportieren', d.h. innerhalb der Simulation zur Verfügung stellen. Dazu verwendet Klasse A eine Instanz der unten vorgestellten Klasse `tlsRegister`, die ebenfalls mit einem Bezeichner in Form einer Zeichenkette bei der zentralen Instanz von `tlsController` registriert wird. Auf diese Weise ist der der von Klasse A exportierte Parameter innerhalb der Simulation über `tlsController` verfügbar. Wenn der Parameter von Klasse A beispielsweise mit dem Bezeichner `'Param1'` exportiert worden ist, so kann dessen Wert mit den Anweisung
`unsigned long tlsController::getValue(char*` *path*`)`
abgefragt werden, wobei die Zeichenkette *path* den Bezeichner des gewünschten Parameters angibt. Der Parameter `'Param1'` würde von einer mit `'a'` bezeichneten Instanz der Klasse A mit der Zeichenkette `'a:Param1'` abgefragt werden. Es wird also der 'Pfad' zu dem Parameter angegeben, eine Schachtelung wie z.B. `'TDU:PairTrigger:Mass_A'` ist möglich, wenn entsprechend das Objekt `'TDU'` ein Objekt `'PairTrigger'` verwendet.
Damit ein von Klasse A exportierter Parameter in einer Klasse B verwendet werden kann, muß Klasse B den Parameter 'importierten'. Dazu verwendet Klasse B eine Instanz der Klasse `tlsInput`. Dieser Instanz wird der Pfad zu dem von Klasse A exportierten Parameter übergeben.
In der Simulation läuft dann folgender Mechanismus ab. Zunächst werden die zentralen Instanzen von `tlsClock` und `tlsController` gebildet. Danach werden die von `tlsWrapper` abgeleiteten Klassen konstruiert. Dabei registrieren sie sich bei den zentralen Instanzen von `tlsClock` und `tlsController`. Zusätzlich werden die Instanzen von `tlsRegister` bei `tlsController` registriert. Danach sind alle Register bei `tlsController` bekannt und verfügbar. Wird ein Register mit einem Bezeichner registriert, unter dem schon ein anderer Parameter verzeichnet ist, so wird das Programm mit einer entsprechenden Fehlermeldung abgebrochen. Nun wird die Methode
`void tlsController::initializeAllDevices()`
aufgerufen. Die Instanz von `tlsController` ruft darauf hin die Methode
`void tlsWrapper::initializeByController()`
aller registrierten Instanzen von `tlsWrapper` auf. Diese fordern die durch Instanzen von `tlsInput` definierten Parameter von `tlsController` an. Sie erhalten die Zeiger auf die Speicherplätze, in denen die entsprechenden Parameter gespeichert werden.[1] Wird ein Parameter angefordert, zu dem keine Registrierung vorliegt, so wird die Simulation mit einer entsprechenden Fehlermeldung abgebrochen.
Nach dieser Prozedur ist die Simulation vollständig initialisiert. Durch einen Aufruf von
`tlsClock::execute()`
wird jeweils die Ausführung eines Taktes in der Hardware simuliert.

`tlsDevice.h tlsDevice.cc`

In diesen Quelltexten werden die Klassen `tlsInput` und `tlsWrapper` implementiert, deren Funktionsweise im vorigen Abschnitt erläutert worden ist. Die Klasse `tlsInput` dient zum

[1] Dies widerspricht dem Konzept der Datenkapselung. Dennoch wurde diese Methode gewählt, um die Ausführungsgeschwindigkeit der Simulation zu maximieren.

Importieren von innerhalb der Simulation bereitgestellten Parametern. Ein Objekt der Klasse `tlsInput` wird in der Initialisierungsliste der Klasse initialisiert, die es verwendet. Hierzu dient der Konstruktor
`tlsInput(tlsWrapper*` *wrapper*`, const char*` *path*`)`
Soll zum Beispiel ein Parameter mit dem Pfad `'TDU:PairTrigger:Mass_A'` mit dem Objekt `mass_a` der Klasse `tlsInput` importiert werden, so muß in der Initialisierungsliste die Zeile
`mass_a(this, "TDU:PairTrigger:Mass_A")`
stehen. Innerhalb der Klasse, die ein Objekt von `tlsInput` verwendet, kann der Wert mit dem Methode
`unsigned long tlsInput::val()`
abgefragt werden.

`tlsRegister.h tlsRegister.cc`

Die Klasse `tlsRegister` modelliert ein Register, wie es in Abschnitt 4.2.2 vorgestellt worden ist. Es dient darüber hinaus zum Bereitstellen eines Parameters innerhalb der Simulation, wie es bei der Beschreibung der Klasse `tlsController` dargelegt worden ist. Initialisiert wird ein Objekt der Klasse `tlsRegister` in der Initialisierungsliste einer von `tlsWrapper` abgeleiteten Klasse. Hierzu wird folgender Konstruktor von `tlsRegister` verwendet:
`tlsRegister(tlsWrapper*` *wrapper*`, char*` *name*`, long unsigned` *bitwidth*`)`
Hier wird der Zeiger *wrapper* auf die Instanz von `tlsWrapper`, die Zeichenkette *name*, die den Parameter bezeichnen soll und die Anzahl der zur Darstellung des Parameters nötigen bits *bitwidth* übergeben. Soll in einer von `tlsWrapper` abgeleiteten Klasse beispielsweise der Parameter `'Param1'` exportiert werden, so ist in dessen Initialisierungsliste die Zeile
`Param1(this, "Param1", 8)`
einzutragen, wenn der Parameter eine Breite von 8 bit haben soll und die Instanz von `tlsRegister` 'Param1' heißt.[2] Innerhalb der von `tlsWrapper` abgeleiteten Klasse kann dann auf den Wert einer eigenen Instanz der Klasse `tlsRegister` mit der Methode
`unsigned long tlsRegister::val() const`
zugegriffen werden. Um dieser Instanz einen neuen Wert zuzuweisen, ist die die Methode
`void tlsRegister::set(unsigned long` *value*`)`
vorgesehen. Der so zugewiesene Wert wird bei dem nächsten Aufruf der Methode
`tlsRegister::execute()`
zwischengespeichert und ist danach über `tlsRegister::val()` verfügbar. Auf diese Weise wird das zeitliche Verhalten der getakteten Register in der Hardware simuliert, siehe Abschnitt 4.2.2.

A.5 tlsFCS

Die Klasse `tlsFCS` hat wie sein Pendant in der Hardware, das Fast Control System, die Aufgabe, die Zeitpunkte der Ereignisse und eine fortlaufende Nummer, die jedes Ereignis kennzeichnet, für das System zur Verfügung zu stellen.

[2] Der Bezeichner der Instanz von `tlsRegister` und der Bezeichner des durch ihn exportierten Parameters müssen selbstverständlich nicht identisch sein, doch erhöht dies die Lesbarkeit des Programmcodes und ist vom Autor der Simulation intendiert.

Ordner: `mino/public/tls_src/tls/tlsFCS/`

`tlsFCS.h tlsFCS.cc`

Die Klasse tlsFCS exportiert im Rahmen des tls-Frameworks zwei Parameter, die in Tabelle A.5 aufgeführt sind. Der Parameter *FCS_BX* stellt die aktuelle Ereignisnummer bereit.

Parameter	Bedeutung	Bezeichner	Breite/bits
FCS_BX	Aktuelle Ereignisnummer	`FCS_BX`	8
BX_Occured	Ereignis ist aufgetreten	`BX_OCCURED`	1

Tabelle A.1: Von der Klasse `tlsFCS` exportierte Parameter.

Ist ein neues Ereignis aufgetreten, so hat der Parameter *BX_Occured* den Wert 1, ansonsten 0. Das in der Simulation verwendete Taktsignal, welches von der Klasse `tlsClock` erzeugt wird, entspricht einer Taktfrequenz von 50 MHz. Die Klasse `tlsFCS` liefert entsprechend alle 4,8 Taktsignale eine neue Ereignisnummer.

A.6 tlsEPT

Die Klasse `tlsEPT` bietet eine Simulation des ECAL Pretriggers im Rahmen des tls-Frameworks. Sie bietet die Möglichkeit, die Informationen der Energiedepositionen im Kalorimeter entweder aus einer Datei zu lesen[3] oder in Verbindung mit ARTE die Tabelle RCCL auszulesen, die die Energieanhäufungen im Kalorimeter enthält. Zur Bestimmung der Tabelle RCCL wird in ARTE ein optimierter Algorithmus verwendet. In der in Abschnitt A.2 vorgestellten Simulation des Pretriggers wird hingegen die Methode angewandt, die in der Hardware vorgesehen ist.
Ordner: `mino/public/tls_src/tls/tlsEPT/`

`tlsEPT.h tlsEPT.cc`

Die in den Quelldateien `tlsEPT.h tlsEPT.cc` implementierte Klasse `tlsEPT` bietet eine einfache Einbindung des ECAL Pretriggers in die Simulation. Die Daten werden entweder aus einer Datei gelesen, die `tlsEPT` mit mittels Methode
`tlsEPT::assignDataFile(FILE*` *dataFile*`)`
zugewiesen worden ist. Dabei ist *dataFile* der Zeiger auf eine bereits geöffnete Dateistruktur. Alternativ kann das Programm, sofern es mit dem Programmpaket ARTE gebunden wird, direkt auf die Daten innerhalb von ARTE zurückgreifen. Wird keine Datei mit `tlsEPT::assignDataFile(FILE*` *dataFile*`)` zugewiesen, so liest die Simulation die ARTE Tabelle RCCL aus (siehe oben). Die Klasse `tlsEPT` exportiert im Rahmen des tlsFrameworks die in Tabelle A.6 aufgeführten Parameter.

[3] Eine solche Datei kann zum Beispiel unter Verwendung der in Abschnitt A.2 vorgestellten Simulation des ECAL Pretriggers erzeugt werden.

Parameter	Bedeutung	Bezeichner	Breite/bits
BX	Ereignisnummer	`BX`	8
ID	Teilchensorte	`ID`	2
P_z	Impuls	`Pz`	7
$\hat{P}_x$	Impuls	`Pxhat`	8
$\hat{P}_y$	Impuls	`Pyhat`	10

Tabelle A.2: Von der Klasse `tlsEPT` exportierte Parameter.

A.7 Simulation der Trigger Decision Unit

Die Simulation der Trigger Decision Unit macht von der von dem tlsFramework unterstützten Modularität Gebrauch. Die TDU wird in logische Abschnitte eingeteilt, um den resultierenden Programmcode übersichtlich gestalten zu können. Zusammengefaßt werden diese Module in der Klasse `tduTDU`, die die Schnittstelle der TDU Simulation darstellt. Die interne Aufbau der TDU-Simulation ist in Abbildung A.1 angegeben und entspricht

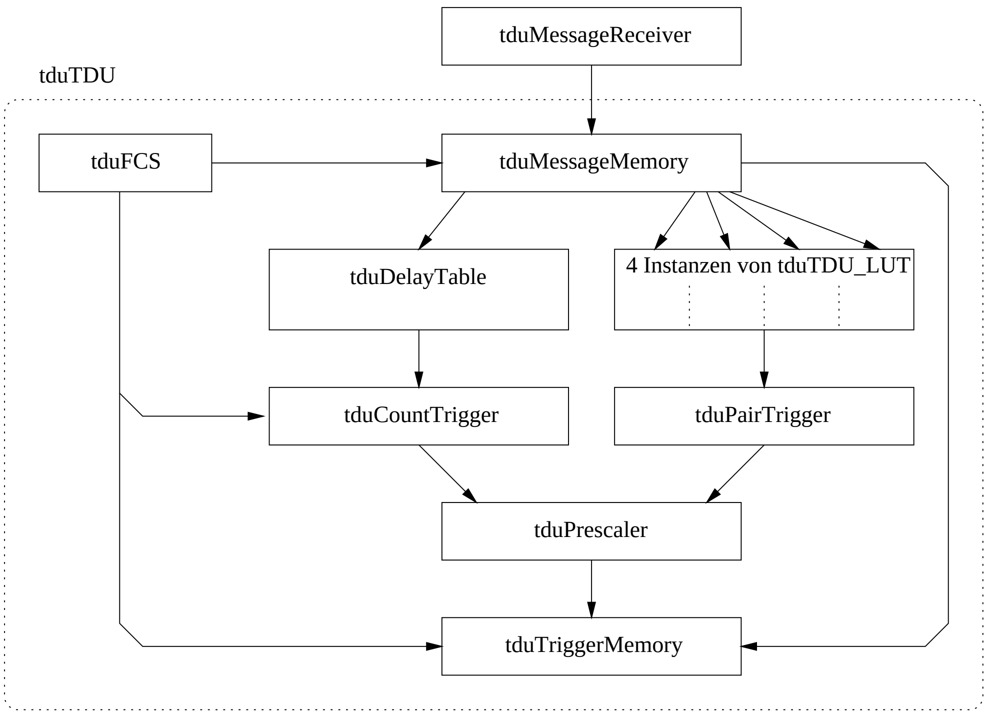

Abbildung A.1: Innerhalb der Klasse `tduTDU` verwendete Instanzen von Klassen, die einzelne Abschnitte innerhalb der TDU simulieren.

dem in Abbildung 3.5 vorgestellten Datenfluß innerhalb der TDU.

`tduTDU.h tduTDU.cc`

Um die im folgenden vorgestellten Teilsimulationen zusammenzufassen, ist die Klasse `tduTDU` vorgesehen. Bei ihrer Konstruktion werden Instanzen der Teilsimulationen gebildet und miteinander verbunden. Die Instanz von `tduTDU` kapselt die internen Objekt, so daß beispielsweise innerhalb der Simulation nur die Instanz von `tduTDU` bei `tlsClock` registriert werden muß. Das Taktsignal wird dann von `tduTDU` an die internen Instanzen der Subkomponenten der TDU Simulation weitergereicht.

`tduMessageReceiver.h tduMessageReceiver.cc`

Die Simulation `tduMessageReceiver` ist in der Lage, zwei Datenströme von TPU Ausgangsdaten in einen Datenstrom zur Eingabe in eine TDU zusammenzuführen.

`tduMessageMemory.h tduMessageMemory.cc`

Die Klasse `tduMessageMemory` modelliert neben dem Message Memory, in dem die Parameter der in dem TFU-Netzwerk rekonstruierten Spuren gespeichert werden, die Funktionen von Messagecounter und -sequencer. In der Hardware wird zwischen Track Memory und Message Memory unterschieden, wobei das Track Memory Daten speichert, die auch im Message Memory abgelegt werden. Diese Redundanz ist in der Hardware erforderlich, um eine unabhängige Auslese dieser Speicher zu ermöglichen.

`tduDelayTable.h tduDelayTable.cc`

Die Klasse `tduDelayTable` dient lediglich zur Verzögerung der Signale, die im Datenflußschema parallel zu den Lookuptable Modulen liegen. Darüber hinaus ist hier keine Funktionalität implementiert.

`tduCountTrigger.h tduCountTrigger.cc`

Die Klasse `tduCountTrigger` simuliert den Count Trigger der TDU. Um die Zählschwellen, bei denen ein Trigger ausgelöst werden soll, zu programmieren, verfügt diese Klasse über die Methode

`void setCTThreshold(const char*` *identifier*`, unsigned long` *threshold*`)`

der der Bezeichner der Zählschwelle und der Wert derselben übergeben wird. Die mit der Zeichenkette *identifier* bezeichnete Zählschwelle wird auf den Wert *threshold* gesetzt. Die Zeichenkette *identifier* muß dazu einen der Werte `'N0'...'N13'` oder `'K9'..'K13'` besitzen, siehe Abschnitt 3.7.2.

`tduTDU_LUT.h tduTDU_LUT.cc`

Die Simulation der Lookuptable Module der TDU wird mit der Klasse `tduTDU_LUT` vorgenommen. Sie berechnet die invarianten Massen von den Spurpaarungen eines Ereignisses. Das Ergebnis wird in Form der sogenannten Triggernummer mit dem Wertebereich von 0 bis 255 geliefert. Der Wert 0 entspricht dem Fall, daß keine positive Triggerentscheidung gefällt werden konnte. Die Werte 1 bis 255 hingegen zeigen die Erfüllung der Triggerbedingungen an und lassen sich verschiedenen Massenbereichen zuordnen. Zusätzlich können weitere Kriterien berücksichtigt werden, wie die Gleichheit der Ladungsvorzeichen beider Spuren, die Pretriggerquelle, der Transversalimpuls der einzelnen Spuren sowie die asymmetrische Energie $E_{Asym} = \frac{E_1 - E_2}{E_1 + E_2}$ beider Spuren.

`tduPairTrigger.h tduPairTrigger.cc`

Die Klasse `tduPairTrigger` erhält die Informationen von den vier Lookuptable Modulen, die durch Instanzen der Klasse `tduTDU_LUT` simuliert werden. Diese Information enthält zum einen die Triggernummern, die gefundenen Trigger wiedergeben und zum anderen die Information über doppelt aufgetretene Spuren. Wird zu einem Spurkandidaten eine doppelte Spur gefunden, so werden die zu diesem Spurkandidaten gefundenen Triggernummern verworfen.

`tduPrescaler.h tduPrescaler.cc`

In dieser Klasse sind die Funktionalität von Count Trigger- und Pair Trigger-Prescaler integriert. Die Skalierungsfaktoren für den Count Trigger werden über die Methode `setCT_PS` eingestellt. Die Skalierungsfaktoren für den Pair Trigger sind zur Zeit nicht justierbar, da eine Herunterskalierung der relativ seltenen Pair Trigger-Ereignisse bei der Untersuchung von Monte Carlo Daten uninteressant ist.

`tduTriggerMemory.h tduTriggerMemory.cc`

Diese Klasse enthält die Simulation des Trigger Memorys, in dem alle gefundenen Trigger der letzten 128 Ereignisse gespeichert werden. Da in dieser Klasse auch die Verbindung zum Second Level Trigger modelliert wird, gibt es wie in Abb. A.1 ersichtlich eine Verbindung von Message Memory zum Trigger Memory.

`tduManualTDUInput.h tduManualTDUInput.cc`

Um eine Simulation allein der Trigger Decision Unit ohne das gesamte Netzwerk zu ermöglichen, wurde weiterhin die Klasse `tduManualInput` bereitgestellt, welcher direkt beliebige Eingangsparameter für die TDU übergeben werden können.

A.8 Schnelle Simulation der TDU

Die vollständige Simulation der TDU hat den Nachteil eines relativ hohen Rescourenbedarfs. Für Untersuchungen, bei denen das zeitliche Verhalten des FLT nicht von Interesse ist, wie zum Beispiel bei den Effizienzbestimmungen, die in Kapitel 6 vorgestellt worden sind. Solange nicht auf interne Parameter der TDU zugegriffen wird, sind die TDU Simulationen der Klassen `tduTDU` und `tlsQuickTDU` austauschbar. Der Unterschied liegt dann nur darin, daß bei Verwendung von `tlsQuickTDU` keine Latenzzeit auftritt.

`tlsQuickTDU.h tlsQuickTDU.cc`

Mit folgenden Methoden kann die Simulation der TDU konfiguriert werden:
`void setCTThreshold(const char*` *identifier*`, unsigned long` *threshold*`)`
Die mit der Zeichenkette *identifier* bezeichnete Zählschwelle wird auf den Wert *threshold* gesetzt. Die Zeichenkette *identifier* muß dazu einen der Wert `'N0'...'N13'` oder `'K9'..'K13'` annehmen, siehe Abschnitt 3.7.2.
`void setHighPtCut(unsigned long` *id*`, double` *cut*`)`

Für jede mit *id* spezifizierte Teilchensorte kann ein Schwellwert für den Transversalimpuls angegeben werden, der in den Lookuptables der TDU berechnet wird.[4] Das Überschreiten

id	Pretriggerquelle
0	Kalorimeter (Elektron)
1	Myonsystem
2	High-p_T
3	Kalorimeter (Photon)

Tabelle A.3: Definierte Werte für den Parameter *id* zur Spezifikation der Pretriggerquelle.

der Schwelle wird dem Count Trigger mitgeteilt, der die entsprechende Spur als High-p_T-Teilchen einordnet.
`void setInvMassPattern(unsigned long` *massPattern*`, double` *lowerCut*`,`
`double` *upperCut*`)`
Die Lookuptables der TDU geben bei der Berechnung der invarianten Masse von Spurpaaren einen vier bit breiten Parameter, den *massPattern*[5], zurück. Der Wert 0 entspricht einer negativen Triggerentscheidung, die Werte 1 bis 15 können beliebigen Massenbereichen zugeordnet werden. Mit der Methode `setInvMassPattern(...)` können den mit *massPattern* spezifizierten Werten die Massenintervalle [*lowerCut*, *upperCut*) zugewiesen werden. Wir einem Wert von *massPattern* kein Massenbereich zugeordnet, so wird der Wert bei einer positiven Triggerentscheidung nicht berücksichtigt. Überlappen sich die Massenbereiche, so wird bei einer mehrdeutigen Triggerentscheidung stets der niedrigst mögliche Wert für *massPattern* ausgegeben.

[4] Dieser Schwellwert ist unabhängig von den in der Pretriggern vorgenommenen Schnitten, die zur Auswahl der Spurkandidaten dienen.
[5] engl. für 'Massenmuster'

Literaturverzeichnis

[alb95] H. Albrecht et al.
HERA-B. An Experiment to Study CP Violation in the B System Using an Internal Target at the HERA Proton Ring. Design Report.
DESY-PRC 95/01.

[bak99] G. Baki 1999
Persönliche Mitteilung.

[buc99] V. Buchholz 1999
Persönliche Mitteilung.

[fla98] J. Flammer 1998
Bei der Untersuchung der Präzision der TDU-Lookuptables mittels Monte Carlo generierter $J/\psi \to \mu^+\mu^-$-Zerfälle hat mir J. Flammer wichtige Hilfestellung geboten.

[fla99] J. Flammer 1999
Simulation des First Level Triggers (FLT) von HERA-B zur Untersuchung der Triggerperformance
Vortrag auf der DPG Frühjahrstagung Heidelberg 1999

[ful96] T. Fuljahn 1996
Tracking through the magnet
FLT-internes Dokument
zu finden unter:
`http://www.desy.de/ftp/pub/herab/trigger/flt/tracking_new.ps`

[ful99] T. Fuljahn 1999
Aufbau und Charakterisierung des schnellen Kontrollsystems für das Experiment HERA-B
Dissertation zur Erlangung des Doktorgrades des Fachbereichs Physik der Universität Hamburg

[ger98] E. Gerndt et al. 1998
Description of Messages and Process-Logic for the First Level Trigger of HERA-B
FLT-internes Dokument

[groe98] A. Gröpl 1998
Eine Beschreibung der Trigger Decision Unit liegt lediglich in Form einer Standalone-Simulation in C vor.

[groe99] A. Gröpl 1999
Persönliche Mitteilung

[hil99] M. Hildebrandt 1999
Entwicklung und Bau der Detektoren für das Innere Spurkammersystem bei HERA-B
Dissertation

[igo97] O. Igonkina, R. Spighi, M. Villa, A. Zoccoli 1997
ECAL performance for J/ψ reconstruction
HERA-*B* internes Dokument: Hera-B 97-150, ECAL 97-006

[itt97] H. Itterbeck 1997
Datenflußdiagramm der TDU-LUT-Prozessoren
FLT-internes Dokument

[itt99] H. Itterbeck 1999
The Status and first experiences with the First Level Trigger of HERA-B
Vortrag auf der IEEE Real Time 99 in Santa Fe, NM, USA

[jos94] N. Josuttis 1994
Objektorientiertes Programmieren in C++: von der Klasse zur Klassenbibliothek

[kay91] B. Kayser 1991
Testing the Standard Model of CP Violation in all Manner of B Decay Modes
Nach einem Vortrag auf dem Second International Symposium on Particles, Strings and Cosmology (PASCOS-91), Boston, März 1991

[kob73] M. Kobayashi, T. Maskawa 1973
Progress in Theoretical Physics 49 (659)

[kre99] P. Kreuzer 1999
Persönliche Mitteilung.

[lan99] A. Lanyov 1999
Beschreibung der Geometrie der äußeren Spurkammern.

[loh94] T. Lohse et al.
HERA-B. An Experiment to Study CP Violation in the B System Using an Internal Target at the HERA Proton Ring. Proposal.
DESY-PRC 94/02.

[med99] M. Medinnis
Persönliche Mitteilung

[mic99] A. Michetti
Persönliche Mitteilung

[rat97] F. Ratnikov 1997
Concept of the First Level Trigger for HERA-B experiment
Veröffentlicht in IEEE Trans. Nucl. Sci. 45 (1998), 1782ff

[rat99] F. Ratnikov 1999
Persönliche Mitteilung.

[ress95] D. Ressing 1995
FLT-internes Dokument ohne Titel
zu finden unter:
`http://www.desy.de/ftp/pub/herab/trigger/flt/geometry.ps`

[ress96] D. Ressing 1996
First level trigger for HERA-B
Nuclear Instruments and Methods in Physics Research A 384 (1996) 131-135

[rie98] K. Riechmann 1998
Einsatz von Siliziumdetektoren in der Hochstrahlungsumgebung von HERA-B
Dissertation
MPI-PhE/98-18, Hera-B 98-217, VDET 98-012

[sam98] D. Samtleben 1998
Studies for a Charge Tag at HERA-B
HERA-*B* internes Dokument: Hera-B 98-080, Physics 98-001

[vil99] M. Villa 1999
Persönliche Mitteilung.

[wol83] L. Wolfenstein 1983
Phys. Rev. Lett. 51 (1945)

[zim99] R. Zimmermann 1999
Zeitmeßelektronik für den HERA-B-Detektor
Dissertation

Erklärung

Ich versichere hiermit, die vorliegende Diplomarbeit selbstständig verfaßt und nur die angegebenen Quellen und Hilfsmittel verwendet zu haben.

Hamburg im August 1999

(Michael Nörenberg)