

修士学位論文  
ATLAS 実験前後方ミュオントリガシステム用 Sector Logic  
の開発

神戸大学自然科学研究科物理学専攻  
一宮 亮

平成 13 年 2 月

## 概要

2006 年から実験開始が予定されている ATLAS 実験は、CERN 研究所において建設中の次世代加速器 LHC(Large Hadron Collider) に設置される大型の汎用検出器である ATLAS 検出器を用いた実験プロジェクトであり、標準理論で予言されている粒子の中で唯一未発見の Higgs 粒子の発見をはじめとした、TeV 領域における多くの物理的成果を達成する事が期待されている。

ATLAS 実験では、生成事象の大部分を多量のバックグラウンドが占める。そのため、その中から目的とする物理現象を含む実験データ(イベント)を短時間で効率的に選び出すトリガシステムの可否が、実験全体の成功の鍵を握ることになる。中でもミュオンの信号はきれいに取り出せることからその重要性は高く、トリガシステムの初段を構成する Level-1(LVL1)トリガシステムの中核をなしている。

ATLAS 測定器のエンドキャップ部に設置される TGC(Thin Gap Chamber) は、トロイダルマグネットの磁場を用いてミュオンの横運動量 ( $P_T$ ) を観測することの出来るトリガチェンバである。TGC の出力はミュオントリガシステムに送られ、そこで  $P_T$  の構築とトリガ処理が行なわれる。また、その結果はデータ収集システム (DAQ) への読みだしも行われる。

この Sector Logic 回路はミュオントリガシステムの主要な回路要素であり、Trigger Sector と呼ばれる領域の中で、最終的にミュオンの横運動量 ( $P_T$ ) を構築し、その値を用いてトリガ処理を行なう機能を持つ。一方、ミュオントリガシステムが Level-1 トリガシステムに位置することから、Sector Logic 回路には、厳しい時間的制約が科せられる。このような特徴を持つ Sector Logic 回路は、非常に大規模な論理回路となるため、最新の設計技法を用いる事が必要となる。

今回の研究では、仕様に基づいて Sector Logic 回路の基礎設計から開発を行った。そして、基礎設計で決定された各機能ブロックについて、実際にプロトタイプ回路を製作し、要求される性能の達成可能性について評価を行った。その結果、要求される仕様を十分に達成出来る事を確認した。その結果を踏まえ、Sector Logic のほぼ全機能を実装したプロトタイプモジュールの設計を行なった。これは、今年 9 月に行なわれる、トリガシステムの統合テストで使用される予定となっている。

# 目次

第 0 章	イントロダクション	5
第 1 章	ATLAS 実験	7
1.1	LHC 計画	7
1.2	ATLAS の目指す物理	8
1.2.1	標準理論 Higgs 粒子	8
1.2.2	超対称性粒子 (SUSY)	9
1.2.3	超対称性 Higgs 粒子	10
1.3	ATLAS 検出器	11
1.3.1	概要	12
1.3.2	内部飛跡検出器 (Tracker)	12
1.3.3	カロリメータ	13
1.3.4	ミュオン・スペクトロメータ	13
1.3.4.1	MDT(Monitored Drift Tube): 精密検出器	14
1.3.4.2	CSC(Cathode Strip Chamber):精密検出器	14
1.3.4.3	RPC(Resistive Plate Chamber):トリガ用検出器	15
1.3.4.4	TGC(Thin Gap Chamber):トリガ用検出器	15
第 2 章	ATLAS 実験のトリガとデータ収集システム	17
2.1	トリガとデータ収集システムの構成	17
2.1.1	トリガシステム	17
2.1.1.1	トリガシステムのスキーム	17
2.1.2	データ収集システム (DAQ)	18
2.1.3	ディテクタコントロールシステム (DCS)	18
2.2	Level-1(LVL1) トリガシステム	18
2.2.1	Central Trigger Processor(CTP)	19
2.2.2	Muon CTP Interface(MUCTPI)	19
2.2.3	Timing, Trigger and Control distribution(TTC)	20
2.3	前後方ミュオントリガシステム	21
2.3.1	前後方ミュオントリガシステムの構成とレイアウト	21
2.3.2	Trigger Sector	22
2.3.3	R- $\phi$ Coincidence Logic の特徴	22
2.3.4	トリガエレクトロニクス	24
2.3.5	モジュールの配置	27
2.3.6	Latency	27
2.3.7	構成要素の補足的説明	28
2.3.7.1	Amp-Shaper-Discriminator(ASD)	28
2.3.7.2	Hi-P <sub>T</sub>	29

2.3.7.3	Star Switch . . . . .	29
<b>第 3 章</b>	<b>Sector Logic</b>	<b>30</b>
3.1	Sector Logic の 基本設計 . . . . .	30
3.1.1	Sector Logic の構成 . . . . .	31
3.1.2	Sub Sector Cluster(SSC) . . . . .	32
3.1.3	TGC 境界の問題 . . . . .	33
3.1.4	Sector Logic の機能ブロック . . . . .	36
3.1.4.1	Decoder . . . . .	37
3.1.4.2	R- $\phi$ Coincidence Logic . . . . .	40
3.1.4.3	De-multiplexer . . . . .	41
3.1.4.4	Track Selection Logic(Track Pre-Selector, Track Selector) . . . . .	42
3.1.4.5	Encoder . . . . .	44
3.2	Track Selection Logic のプロトタイプとテスト . . . . .	45
3.2.1	Track Selection Logic の Verilog-HDL による記述と論理シミュレーション . . . . .	45
3.2.1.1	Primary/Secondary Selector の実装 . . . . .	47
3.2.1.2	結果 . . . . .	48
3.2.2	pt3 モジュールの仕様 . . . . .	49
3.2.3	プロトタイプの仕様と設計 . . . . .	50
3.2.4	プロトタイプの動作検証 . . . . .	51
3.3	R- $\phi$ Coincidence Logic の設計とテスト . . . . .	52
3.3.1	プロトタイプの仕様と設計 . . . . .	54
3.3.1.1	低電圧化への対応 . . . . .	54
3.3.1.2	VME コントロール CPLD と VME . . . . .	55
3.3.1.3	IO Emulator ブロック FPGA(F0) . . . . .	55
3.3.1.4	R- $\phi$ Coincidence Logic ブロック (F1) . . . . .	57
3.3.2	プロトタイプの動作検証 . . . . .	59
3.3.2.1	SRAM を用いた設計 . . . . .	59
3.3.2.2	SRAM 版の結果 . . . . .	59
3.3.2.3	BlockRAM を用いた設計 . . . . .	60
3.3.2.4	BlockRAM 版の結果 . . . . .	61
3.3.2.5	どちらを使うか? . . . . .	62
3.4	Prototype-0 の設計 . . . . .	62
3.4.1	要求される仕様と構成 . . . . .	62
3.4.2	VME コントロール CPLD . . . . .	63
3.4.3	入出力ブロック . . . . .	64
3.4.3.1	Hi-P <sub>T</sub> からの入力 . . . . .	64
3.4.3.2	MUCTPI への出力部 . . . . .	64
3.4.3.3	TTCrx と Readout Buffer . . . . .	64
3.4.4	FPGA-0(XCV405E-6BG560C × 2) . . . . .	64
3.4.5	FPGA-0(XCV400E-6BG432C × 1) . . . . .	65
<b>第 4 章</b>	<b>まとめ</b>	<b>66</b>
	参考文献	69

# 目次

1.1	LHC のレイアウト	7
1.2	標準理論 Higgs 粒子の生成断面積	8
1.3	標準理論 Higgs 粒子の代表的な生成過程: (a)gluon-gluon 融合, (b)WW(ZZ) 融合, (c) $q\bar{q}$ 対消滅, (d) $t\bar{t}$ を含む生成	9
1.4	標準理論 Higgs 粒子の各崩壊過程への分岐比	10
1.5	ATLAS 検出器の全体図	11
1.6	中央飛跡検出器	12
1.7	カロリメータ	13
1.8	ミュオン・スペクトロメータの R-Z 断面図	14
1.9	MDT(Monitored Drift Tube)	14
1.10	CSC(Cathode Strip Tube)	15
1.11	RPC(Resistive Plate Chamber)	15
1.12	TGC(Thin Gap Chamber) の構造	16
1.13	TGC(Thin Gap Chamber) Doublet と Triplet	16
2.1	ATLAS 実験のトリガシステムのスキーム	17
2.2	Level-1 トリガシステムのスキーム	19
2.3	CTP(Central Trigger Processor) のブロックダイアグラム	20
2.4	MUCTPI のフロントエンド (MIOCT) のプロトタイプ	20
2.5	TGC の配置の R-z 平面図	21
2.6	Trigger Sector	22
2.7	ミュオンの横運動量測定用トロイダル・マグネット群	23
2.8	均等な 6GeV のミュオンをトロイダル・マグネットの磁場中に入射した時の Pivot Plane における通過位置	23
2.9	前後方ミュオントリガシステムの構成	25
2.10	Patch Panel(PP), Slave Board(SB), Hi- $P_T$ (HP), Sector Logic(SL) との間の信号の接続関係	26
2.11	各モジュールの設置場所	27
2.12	ASD Board	28
2.13	Hi- $P_T$ における Track Selection	29
3.1	パイプライン構造 (上) と CPU 等の並列処理	31
3.2	Sector Logic のブロックダイアグラム (概念図)	32
3.3	SSC の構造	33
3.4	入力の構造と Trigger Sector の SSC への分割	34
3.5	TGC 境界と、それを含む SSC	35
3.6	TGC 境界を含む SSC の分割	36
3.7	Sector Logic の機能ブロック別の構成	37
3.8	SSC への入力信号	41

3.9	EI(EndCap Inner), FI(Forward Inner) TGC の R- $\phi$ 平面図 . . . . .	41
3.10	De-multiplexer の入力信号と出力信号 . . . . .	42
3.11	Track Pre-Selector の構造と入出力信号 . . . . .	43
3.12	Track Selector の構造と入出力信号 . . . . .	44
3.13	Primary/Secondary Selector の実装 . . . . .	45
3.14	MUCTPI へ送るデータのフォーマット . . . . .	46
3.15	Track Selection Logic の論理シミュレーションの結果 . . . . .	48
3.16	pt3 モジュールの写真 . . . . .	49
3.17	pt3 モジュールのブロック図 . . . . .	49
3.18	Track Selection Logic の 4 個の FPGA への振り分け . . . . .	50
3.19	Track Selection Logic の 4 個の FPGA 間のデータの配線 . . . . .	51
3.20	pt3 による Track Selection Logic の動作波形 (20MHz) . . . . .	52
3.21	pt3 による Track Selection Logic の動作波形 (40MHz) . . . . .	53
3.22	SSC の halfSSC への分解 . . . . .	53
3.23	R- $\phi$ Coincidence Logic の実装評価用プロトタイプ (slm1) . . . . .	54
3.24	slm1 モジュールのブロック図 . . . . .	55
3.25	R- $\phi$ Coincidence Logic の LUT 周りの回路 . . . . .	58
3.26	R- $\phi$ Coincidence SRAM 版 (40MHz) . . . . .	60
3.27	R- $\phi$ Coincidence BlockRAM 4bit 版 (40MHz) . . . . .	61
3.28	R- $\phi$ Coincidence BlockRAM 4bit 版 [最適化](60.6MHz) . . . . .	62
3.29	Prototype-0 の主要部のブロックダイアグラム . . . . .	63
4.1	Sector Logic の開発・量産スケジュール . . . . .	67

# 第0章 イントロダクション

現代物理学において、標準模型 (Standard Model) は全ての素粒子とその相互作用を極めて良い精度で説明する、20 世紀最大の成功の一つである。標準理論の予言する現象は実験によって検証され、いずれも実験精度の限界まで予言値と一致している。しかし、標準模型が予言する素粒子の中で、唯一未発見な粒子が Higgs 粒子である。Higgs 場の自発的対称性の破れ (相転移) により、真空は期待値を持ち、その結果、物質を構成しているフェルミオン (クォークとレプトン) や弱い力を媒介するゲージボソン (W, Z) に質量を与えている。標準模型の中でこのような重要な役割を果たす Higgs 粒子は、理論的に質量の上限が 1TeV とされており、実験的発見が待ち望まれている。

CERN 研究所において建設中の次世代加速器 LHC(Large Hadron Collider) は、このエネルギー領域を隈無く探索することの出来る最初の加速器であり、その衝突点の一つに設置される検出器に ATLAS 検出器がある。ATLAS 検出器は、Higgs 粒子探索を始めとした新しい物理現象の発見や、様々な物理定数の精密測定を行うことの出来る汎用検出器であり、その目的を達成するために、多くの高精度な検出器が設置される。そして、検出器全体から収集されるデータ量は膨大なものとなる。

そのため、目的とする物理現象を含む実験データ (イベント) のみを選択的・効率的に収集することの出来る、優れたトリガシステム・データ収集システム (DAQ) が不可欠になる。ATLAS 検出器のトリガシステムは、3 段階の階層構造を持ち、データ収集システム (DAQ) と組み合わせられて、柔軟かつ効率的なデータ選択を行うように設計されている。

本論文で扱う Sector Logic 回路は、この ATLAS 検出器トリガシステムのための回路である。その中でも、Sector Logic 回路はミュオンを用いるミュオントリガの一部であり、Trigger Sector と呼ばれる領域内部のミュオンの横運動量 ( $P_T$ ) を構築し、その値を用いてトリガ処理を行う機能を担っている。また、このミュオントリガは、ATLAS トリガシステムの初段 Level-1(LVL1) に位置するため、検出器から 40.08MHz の高周波で送られて来るデータをリアルタイムで、デッドタイムレスで処理する必要があり、さらに短時間で処理を完了する必要がある。

これらの処理を行う Sector Logic 回路は非常に大規模な回路であり、柔軟性を持ちつつ高速処理を行う必要がある。このような大規模開発に適した最新の技法にハードウェア記述言語 (HDL) を用いた実装技法がある。これは、既に論理回路 (デジタル回路) の ASIC(特定用途向け集積回路) の設計では標準的な技法であり、今や大規模なデジタル回路の設計には不可欠となっている。

ところで、高速論理回路のための半導体デバイスである FPGA(Field Programmable Gate Array) の、近年の性能向上 (回路規模・速度) は目覚ましいものがある。FPGA は、デバイスを基板に実装したままで、設計変更した回路をダウンロードすることにより、新しい回路として動作させることの出来る特徴を持ったデバイスであり、設計変更に対応しつつ柔軟な回路設計を行う必要のある Sector Logic 回路には、まさに最適であると言える。さらに、大規模化した FPGA の性能を十分に活かしつつ効率的な設計を行うには、HDL を用いる事が必須となる。そのため、Sector Logic 回路では HDL を用い、FPGA を全面採用することにした。

第1章では、ATLAS 実験の概要を述べる。ATLAS 実験のプラットフォームである LHC 計画と実験全体のあらまし、ATLAS 実験の目的とする物理対象を述べたあと、ATLAS 検出器を構成する検出器について概観を与える。

第2章は、ATLAS 実験のトリガシステムとデータ収集システム (DAQ) について説明する。まず、トリガシステムとデータ収集システム (DAQ) の概要を述べた後、本論文の主題である Sector Logic 回路の属する Level-1(LVL1) トリガシステムの役割とその機能、構成が記述される。

第3章は、本論文の核心部である。Sector Logic の基礎設計の詳細について述べた後、プロトタイプによる  $R\text{-}\phi$  Coincidence Logic と Track Selection Logic の検証について詳述する。その後、ほぼ全ての機能を持った Sector Logic のプロトタイプである、Prototype-0 の設計について述べる。

第4章は、ATLAS 実験で実際に使われる本番用の Sector Logic の製作に向けて、今後どのような開発過程が必要となるか、見通しと予定を説明する。



# 第1章 ATLAS 実験

## 1.1 LHC 計画

スイスのジュネーブの郊外にフランスとの国境を挟んで設置されている CERN(欧州素粒子物理研究所)では、世界最高の重心系エネルギーを持つ大型陽子陽子衝突型加速器 (LHC: The Large Hadron Collider) が 2005 年の運転開始を目指して現在建設が進められている。LHC は、大型電子陽電子衝突型加速器 (LEP: The Large Electron Positron Collider)<sup>1</sup>のトンネル (周長 27km) 内に建設されており、14TeV の重心系衝突エネルギーを持つ。また、バンチ交差頻度が 24.95ns と従来の加速器に比べて極めて大きく、一回のバンチ交差あたり 平均 23 回の陽子陽子衝突が予測されている。

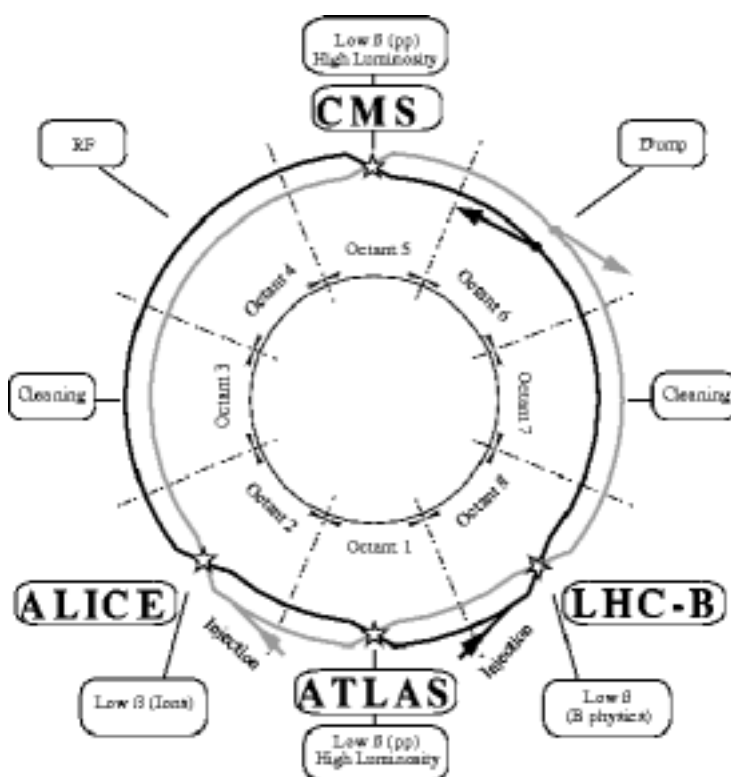


図 1.1: LHC のレイアウト

衝突点は 4ヶ所設置され、それぞれの衝突点に、汎用検出器の ATLAS(A Toroidal LHC Apparatus) と CMS(The Compact Muon Solenoid)、B-physics に特化した LHC-B、重イオン衝突実験用の ALICE(A Large Ion Collider Experiment) が設置される予定である。

ATLAS 実験は、LHC 加速器の持つ 14TeV の重心系衝突エネルギーを活用し、TeV 領域の素粒子の探索を行う、エネルギーフロンティアの実験である。そのなかでも、現在標準理論で唯一未発見の粒子であるヒッグスボソンの検出は ATLAS 実験で最も期待されているテーマである。

<sup>1</sup>2000 年 11 月 2 日、LEP は 11 年間に渡る電子陽電子衝突のエネルギーフロンティアを切り開いて来た運転を終了した。

この他にも、ATLAS 検出器では超対称性粒子 (SUSY) の発見や、TeV 領域に存在する未知の新粒子の発見が期待されている他、LHC 加速器の持つ豊富な luminosity を活かした b クォークや W/Z などのゲージ・ボソンの質量や崩壊過程の精密測定や QCD の精密研究が可能となる。

## 1.2 ATLAS の目指す物理

ここでは、ATLAS 実験の目的とする物理について述べる。

### 1.2.1 標準理論 Higgs 粒子

標準理論は、一つの中性 Higgs 粒子  $H_{SM}^0$  の存在を予言しており、真空の安定性の条件から、その質量は 1TeV 以下であると考えられている。図 1.2 に LHC における Higgs 粒子の生成断面積を示す。また、図 1.3 は、陽子陽子衝突における主要な標準理論 Higgs 粒子の生成過程である。

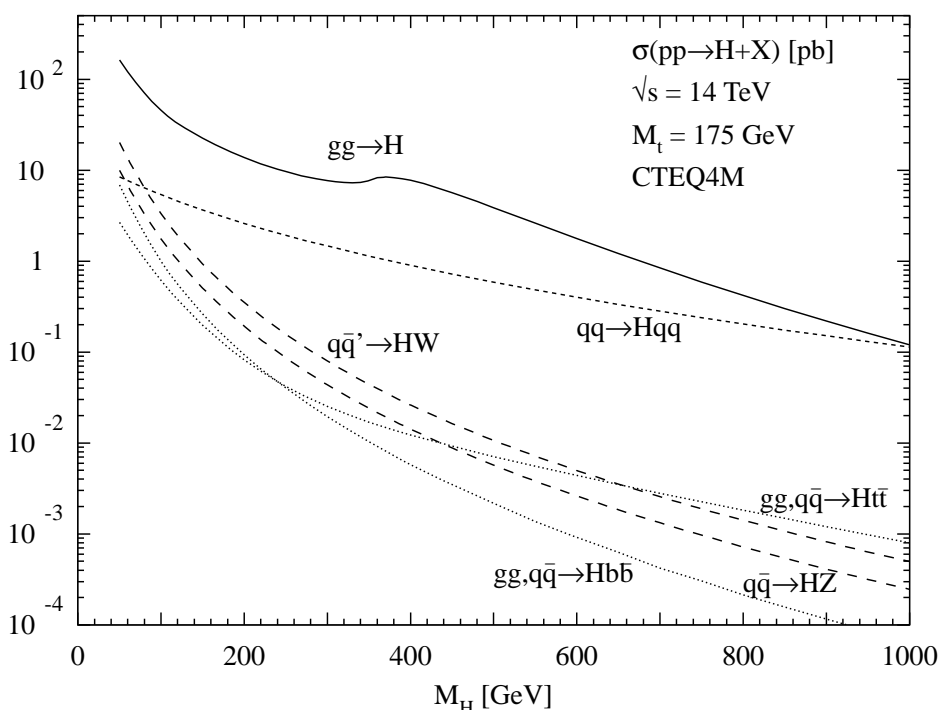


図 1.2: 標準理論 Higgs 粒子の生成断面積

図 1.4 に各崩壊過程への崩壊分岐比を示す。図から判るように、Higgs 粒子の生成断面積及び崩壊分岐比は  $H_{SM}^0$  の質量  $m_H$  に依存する。ATLAS 検出器では、以下のように  $H_{SM}^0$  により領域を分け、各領域ごとに感度の高い検出方式を用い、Higgs 粒子の探索を行う。 $m_H$  がおおむね 80GeV から 1TeV の領域で探索が可能である。

$80\text{GeV} < m_H < 130\text{GeV}$   $m_H < 2 m_W$  の領域では、 $b\bar{b}$  崩壊が支配的であるが、QCD バックグラウンドが大きいためこれを有効にトリガする方法がない。したがって、分岐比は  $\sim 10^{-3}$  と低い  $H \rightarrow \gamma\gamma$  を用いる。この Higgs 粒子崩壊に由来する 2 つの  $\gamma$  から不変質量  $M_{\gamma\gamma}$  を組むと Higgs 粒子の質量に相当する位置に鋭いピークを検出する。QCD による  $\gamma\gamma$  が連続したバックグラウンドとして存在

するため、ピークを検出するためには、エネルギー及び角度分解能の優れた  $\gamma$  測定器 (電磁カロリメータ) が要求される。

$130\text{GeV} < m_H < 700\text{GeV}$  この領域では、最も信頼性の高い探索モードである  $H \rightarrow ZZ^{(*)} \rightarrow l^+l^-l^+l^-$  (gold plated events) を主に用いる。このモードは、独立した2組のレプトン対の不変質量  $m_{l\bar{l}} \sim m_Z$  を事象 (イベント) 選択条件に用いることで大幅にバックグラウンドを落せるため、最もきれいな信号が得られる。本論文の主題である、Sector Logic 回路もミュオントリガシステムの一部として、主にこの領域を対象にトリガ条件 (ミュオンの  $P_T$  値) が設定されている。 $150\text{GeV} < m_H < 180\text{GeV}$  の領域では  $H \rightarrow WW$  の崩壊モードが開くものの、 $H \rightarrow ZZ$  への崩壊モードはまだ開かないため、やや分岐比は下がる。また、 $m_H > 800\text{GeV}$  になると、共鳴幅が大きく広がるため、Higgs 粒子の明確なピークを検出するのが困難になって来る。

$700\text{GeV} < m_H < 1\text{TeV}$  この領域では Higgs 粒子の崩壊幅が広がるため、分岐比の最も大きい  $WW$  崩壊過程を用いて、イベント数を稼ぐことが重要になって来る。この領域で有効な探索モードの一つに、 $WW$  融合 (図 1.3(b)) によって生成された Higgs 粒子の生成がある。このイベントは、前方にクォークによる2つのジェットが生じるのが特徴であり、これをタグすることにより同定することが可能である。

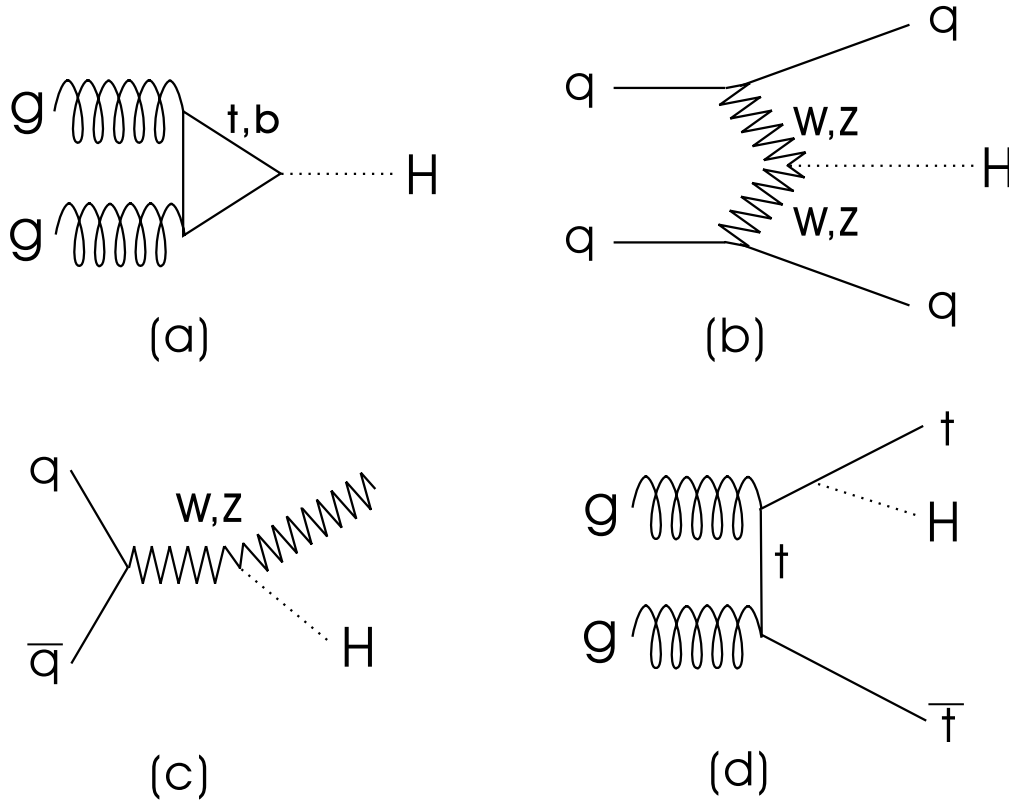


図 1.3: 標準理論 Higgs 粒子の代表的な生成過程: (a)gluon-gluon 融合, (b)WW(ZZ) 融合, (c)  $q\bar{q}$  対消滅, (d) $t\bar{t}$  を含む生成

### 1.2.2 超対称性粒子 (SUSY)

超対称性変換は、ボソンとフェルミオンを交換する。超対称性変換によって結ばれたボソンとフェルミオンはスーパーパートナーと呼ばれる。例えば、クォークとレプトンのスーパーパートナーはボソンのス

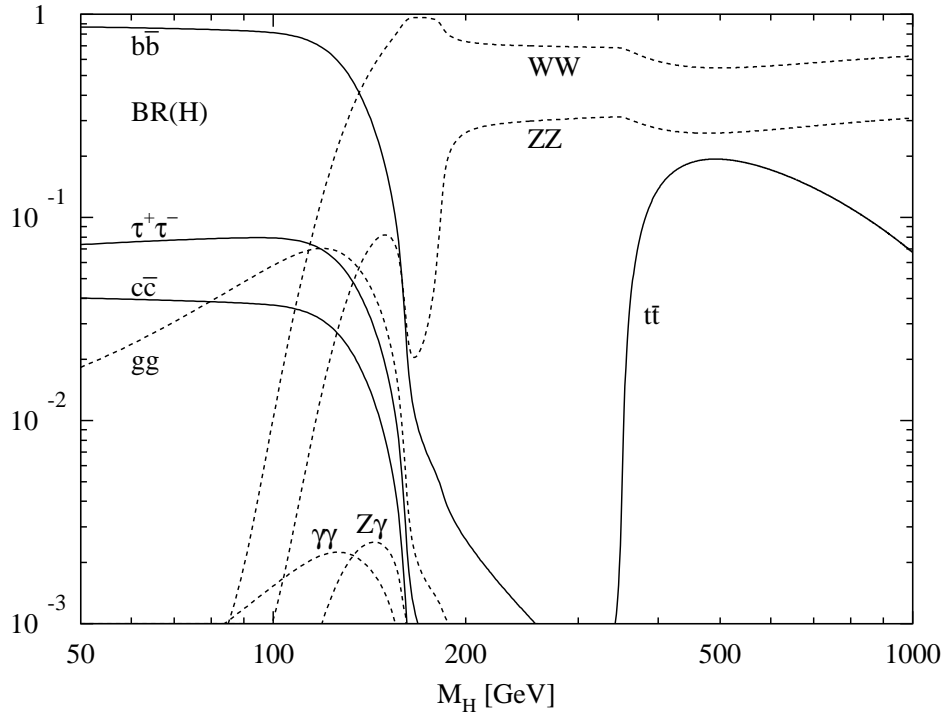


図 1.4: 標準理論 Higgs 粒子の各崩壊過程への分岐比

クォークとスレプトンとなる。

LHC では、強い相互作用をするスクォークやグルイーノ対が大量に生成される。これらは、非 SUSY 粒子から生成されているため、R パリティ<sup>2</sup>の保存を要請すると、SUSY 粒子は必ず対になって生成する。生成された SUSY 粒子は軽い粒子への崩壊を繰り返し、最終的に最も軽い SUSY 粒子 (LSP: Lightest SUSY Particle) に崩壊する。LSP は、電荷やカラーを持たない中性の粒子であり直接観測にかからないため、途中の崩壊過程でのジェットやレプトン及び消失エネルギーの観測によって検出する。

### 1.2.3 超対称性 Higgs 粒子

超対称性理論のなかでも最も簡単な MSSM (Minimal supersymmetric extension of Standard Model) では、スーパーチャージ  $Y = 1, -1$  の higgs 二重項が必要であり、最終的に  $H^\pm, A^0, H^0, h^0$  の 5 種類の MSSM Higgs が生じる。但し、 $A^0$  は CP が負 (odd) のボソンであり、 $H^0$  と  $h^0$  は CP が正 (even) のボソンで、 $m_H > m_h$  とする。

したがって、ATLAS 検出器で Higgs 粒子が 2 個以上検出された場合、超対称性 Higgs 粒子のような標準理論の枠を越えた物理現象が存在することを示唆することになり、現代物理学に対して大きな影響を与えることになる。

さらに、これらの 5 個の Higgs の質量は Tree level のみで考えると、2 個のパラメータ  $\tan \beta, m_A$  で表され、 $m_h < m_z$  となる Higgs 粒子が存在する。ただ、SUSY 粒子や t による放射補正を入れると実質的にはフリーパラメータになるものの、ほとんどの場合  $h^0$  の存在する上限は 150 GeV であろうと考えられている。そのため、この領域で Higgs 粒子がされた場合、さらに高いエネルギー領域に残りの Higgs 粒

<sup>2</sup>B をバリオン数、L をレプトン数、S をスピン量子数とすると  $R = (-1)^{3(B-L)+2S}$  で表される。非 SUSY 粒子は、 $R=1$ 、SUSY 粒子は  $R=-1$  である。

子が存在するかどうかがこの問題の鍵となる。

### 1.3 ATLAS 検出器

ATLAS 検出器は、LHC の持つ世界最高の重心系衝突エネルギーと高ルミノシティを最大限に利用して、観測対象を特定の物理現象に限定せず、エネルギーフロンティア領域のあらゆる素粒子物理現象を捉えることが出来るように設計された、巨大な検出器である。ATLAS 検出器は、LHC の高ルミノシティ下においても、電子や  $\gamma$  線等の電磁シャワー、ジェットによるハドロンシャワー、ミューオンなど、出来るだけ多くのシグナルをバランス良く取り出し、精度の良い測定が出来るよう、測定器が配置されている。直径は 22m、長さ 44m の円筒形をしており、総重量は 7000t である。このような巨大な検出器を開発するため、35ヶ国、2000 名以上の研究者による国際共同実験として開発・建設が進められている。

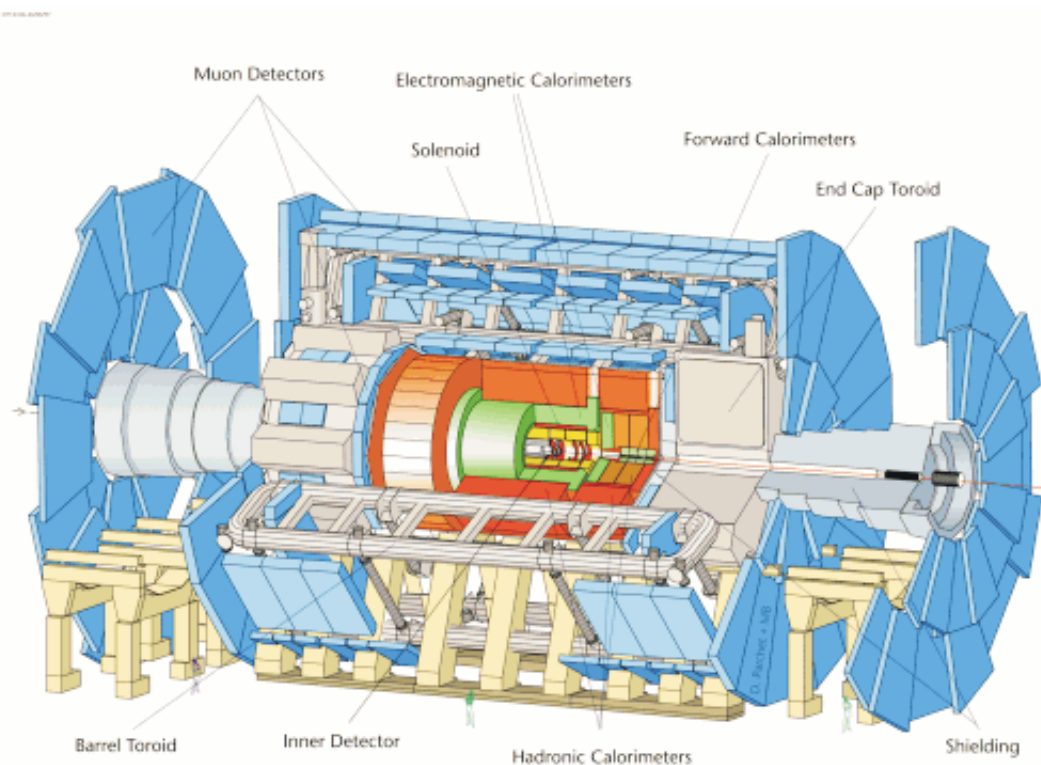


図 1.5: ATLAS 検出器の全体図

#### ATLAS で用いられる座標系と用語の定義

ATLAS 検出器は、ビーム軸によって貫かれた円筒形をした検出器であり、基本的に円筒座標を用いる。即ち、ビーム軸を  $Z$  軸、これと直交する半径方向を  $R$  軸、円周方向を  $\phi$  方向と定義する。このほか、円筒の端面 ( $R$ - $\phi$  平面に並行) を EndCap 部、円筒の胴部を Barrel 部と定義する。

また、衝突点と任意の点を結ぶ直線と  $Z$  軸との間の角度  $\theta$  の代わりに  $\eta = -\ln(\tan(\theta/2))$  と定義される pseudo rapidity(擬ラピディティ)  $\eta$  を用いる。これは、衝突型加速器での高エネルギー衝突反応における終状態粒子の角分布は、 $Z$  軸方向にブーストされるため、 $dN/d\eta$  がほぼ一定になることによる。

### 1.3.1 概要

ATLAS 検出器を構成する検出器は、以下のような種類に分けられる。

- 内部飛跡検出器 (Tracker)
- カロリメータ (EM, hadron)
- ミューオン・スペクトロメータ

以下、それぞれの検出器について概観する。

### 1.3.2 内部飛跡検出器 (Tracker)

荷電粒子のトラッキングを行う内部飛跡検出器は、衝突点に最も近い位置に設置され、中央ソレノイドの約 2T の磁場によって運動量の測定を行う。

中央飛跡検出器は、次の検出器から構成される。

**Pixel Detector** 最内層にある半導体検出器。1pixel が  $50\mu\text{m} \times 300\mu\text{m}$  の位置分解能を持つ。

**SCT(SemiConductor Tracker)** シリコンマイクロストリップと称される、細長い有感領域を持った半導体検出器である。

**TRT(Transition Radiation Tracker)** 半径 4mm のストローチューブ検出器であり、トラッキングの他に遷移放射を利用した電子識別を行う。最外層に設置される。

図 1.6 に中央飛跡検出器の全体図を示す。

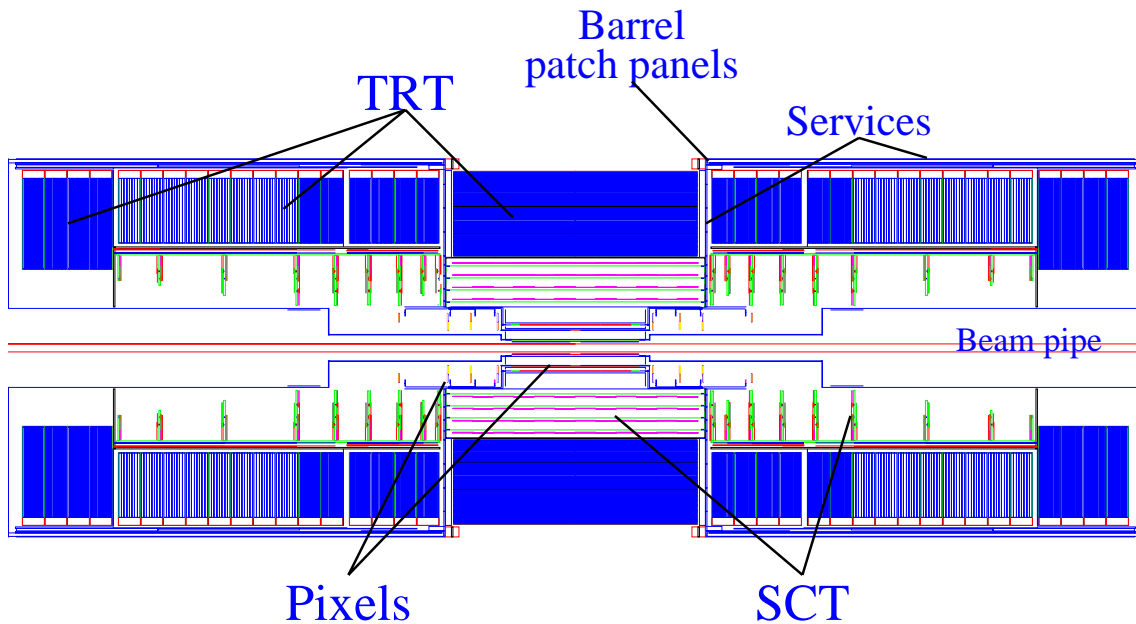


図 1.6: 中央飛跡検出器

### 1.3.3 カロリメータ

カロリメータは、カスケードシャワーを起こさせて入射粒子のエネルギーを測定する検出器である。電子や $\gamma$ 線に電磁シャワーを起こさせる電磁カロリメータとハドロンを吸収してハドロンシャワーを起こさせるハドロンカロリメータの2種類がある。

ATLAS 実験では、放射線耐久性に優れた LAr(液体アルゴン) カロリメータを主として用いている。カロリメータは次の検出器から構成される。

アコーディオン型電磁カロリメータ 鉛の吸収体がアコーディオン構造の LAr 電磁カロリメータ。

Hadronic LAr カロリメータ (EndCap 部) 銅の吸収体による LAr ハドロンカロリメータ。

Hadronic Tile カロリメータ (Barrel 部) 鉄とシンチレータによるハドロンカロリメータ。

Forward LAr カロリメータ 超前部 ( $3.2 < |\eta| < 4.9$ ) をカバーする、LAr カロリメータ。

図 1.7 にカロリメータの全体図を示す。

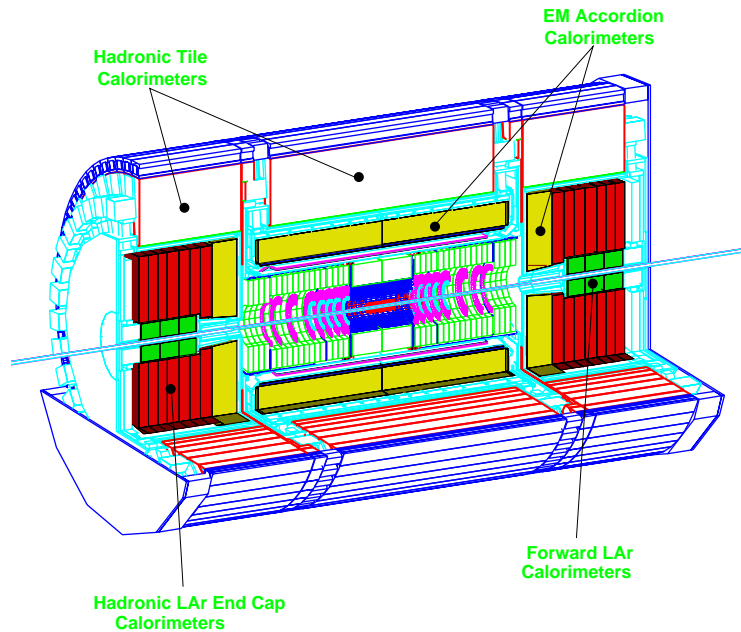


図 1.7: カロリメータ

### 1.3.4 ミューオン・スペクトロメータ

ミューオンは、物質中の通過力が高くハドロンカロリメータを突き抜けてくるため、最外層に設置したミューオン・スペクトロメータで測定を行う。

ATLAS 実験のミューオン・スペクトロメータでは、Barrel 部および EndCap 部のそれぞれに設置してある超伝導空心トロイダル磁石 ( $\phi$  方向に 8 回対称に設置されている。図 2.7 参照) の発生する  $\phi$  方向の磁場によって曲げられたミューオンの曲率を測定することにより、その運動量を測定する。

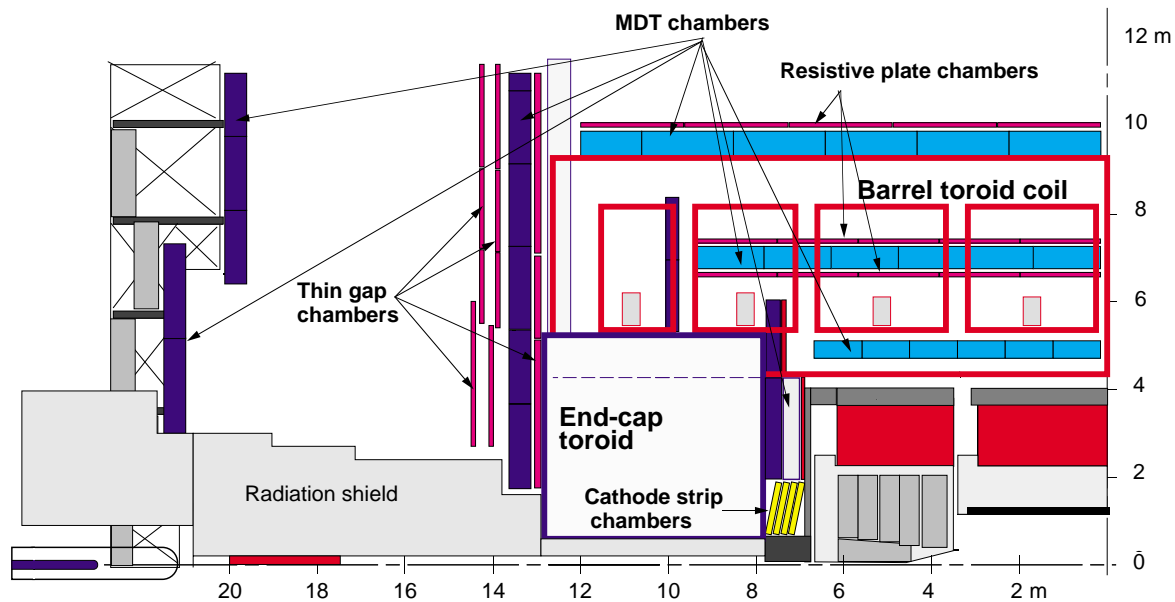


図 1.8: ミューオン・スペクトロメータの R-Z 断面図

ミューオン・スペクトロメータは、精密測定用とトリガ用の 2 種類のミューオン検出器から構成される。以下、それぞれのミューオン検出器について述べる。

#### 1.3.4.1 MDT(Monitored Drift Tube): 精密検出器

MDT は図 1.9 に示すように、チューブ径直径 30mm のドリフトチューブを倂積みしたものをフレームに固定した構造をしている。チューブ内には、Ar(91%)/N<sub>2</sub>(4%)/CH<sub>4</sub>(5%) の組成の混合ガスが 3 気圧で封入されており、最大ドリフト時間は、500ns、位置分解能は、80 $\mu$ m である。

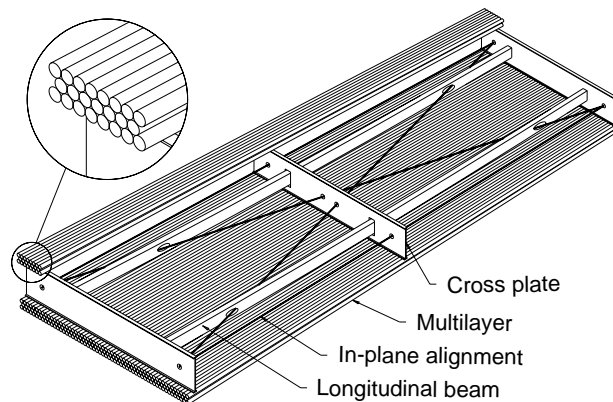


図 1.9: MDT(Monitored Drift Tube)

#### 1.3.4.2 CSC(Cathode Strip Chamber):精密検出器

CSC は、図 1.10 の構造をした、カソード読みだしの MWPC である。位置分解能は、60 $\mu$ m であり、かつ 30ns 以下のドリフト時間を持つ。



Ar/CO<sub>2</sub>/CF<sub>4</sub> の混合ガスで動作し、バックグラウンドイベントとしてミュオン測定妨げになる中性子への感度は 10<sup>-4</sup> 未満であり、 $\gamma$  線への感度も 1% 程度である。そのため、最も放射線強度の高い超前方に設置される。(図 1.8 参照)

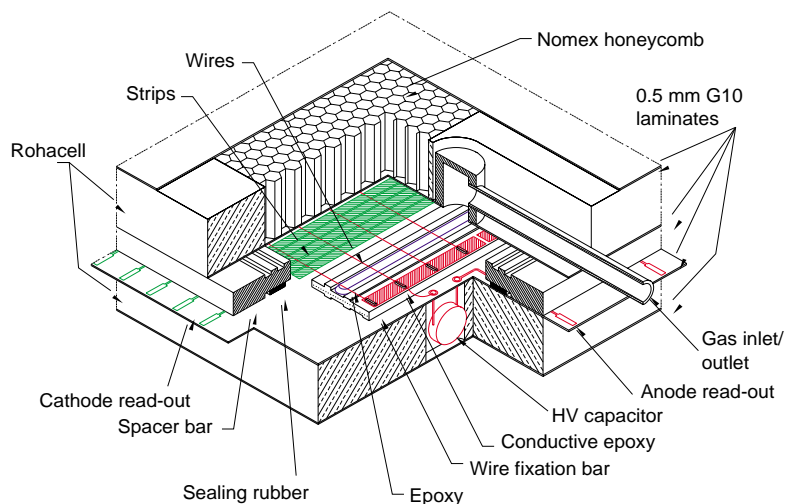


図 1.10: CSC(Cathode Strip Tube)

#### 1.3.4.3 RPC(Resistive Plate Chamber):トリガ用検出器

RPC は  $|\eta| < 1$  の Barrel 部をカバーするトリガ用検出器である。RPC の特徴は、アノードにワイヤを用いないガスチェンバーであることで、2枚の並行するベークライト板の間のガス (C<sub>2</sub>H<sub>2</sub>F<sub>4</sub>(不燃性)) には数 kV/mm の電場を印加し、信号はストリップから読み出す。RPC はストリップ方向が直交するように重ねられ、二次元読みだしが可能である。

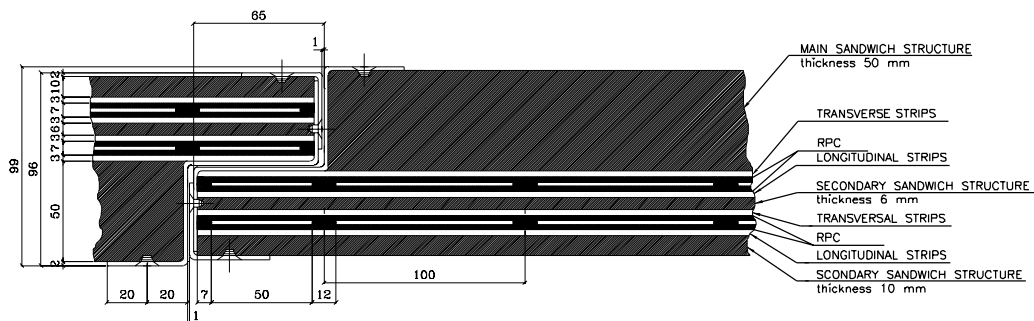


図 1.11: RPC(Resistive Plate Chamber)

#### 1.3.4.4 TGC(Thin Gap Chamber):トリガ用検出器

TGC は、 $1 < |\eta| < 2.7$  の EndCap 部をカバーするトリガ用検出器である。アノードワイヤとカソード面の間隔 (ギャップ) がワイヤ間隔 (1.8mm) よりも狭い 1.4mm と狭くなっている MWPC(Multi Wire

Proportional Chamber) 型検出器であり、そのため TGC(Thin Gap Chamber) と称される。この狭いギャップによりドリフト時間が短くなり、高い時間分解能を確保している。

TGC は、1.6mm 厚のガラスエポキシ樹脂 (G-10) の構造体の内側壁に炭素を塗布することによりカソード面を形成する。(Graphite layer) 3.1kV の HV(High Voltage) を印加するアノードワイヤは  $50\mu\text{m}$  の金メッキされたタンゲステン線であり、カソード面とは 1.4mm のギャップを持って張られている。また、カソード面の Graphite layer の面抵抗を大きくすることで、カソード面の電荷拡散を遅くしており、これにより G-10 の外側にストリップ状に貼った銅箔のカソードストリップでこの電荷の変化を読み取る事を可能にしている。カソードストリップは、アノードワイヤとは直交する方向に向いて形成されているため、2次元読みだしが可能になっている。なお、アノードワイヤは  $\phi$  方向に張られ、6-20 本毎にワイヤグループとして読み出される。また、カソードストリップは R 方向に張られており、1 本づつ読み出される。

TGC はガスゲイン  $10^6$  の Limited Streamer mode で動作し、クエンチ能力が高い  $\text{CO}_2 : n\text{-pentan} = 55 : 45$  の混合ガスを使用する。そのため、TGC は  $1\text{kHz}/\text{cm}^2$  の高入射頻度下でも動作する。これは、実際の ATLAS 実験で予想される最大の入射頻度の 10 倍に相当する。

TGC はその面積に対して極めて薄い構造をしており、そのままでは支持できないため、複数の TGC をペーパーハニカムを支持体を張り合わせて用いられる。2 枚構成のものを TGC Doublet、3 枚組のものを TGC Triplet と呼ぶ。

図 1.12 に TGC の構造を、図 1.13 に TGC Doublet と Triplet の構造を示す。

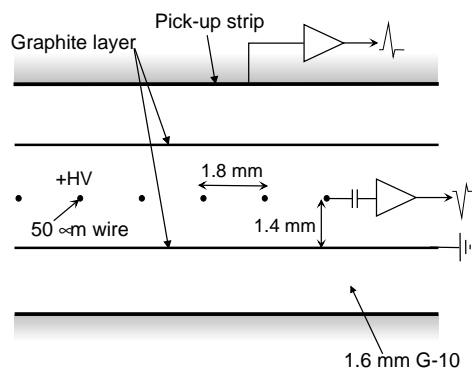


図 1.12: TGC(Thin Gap Chamber) の構造

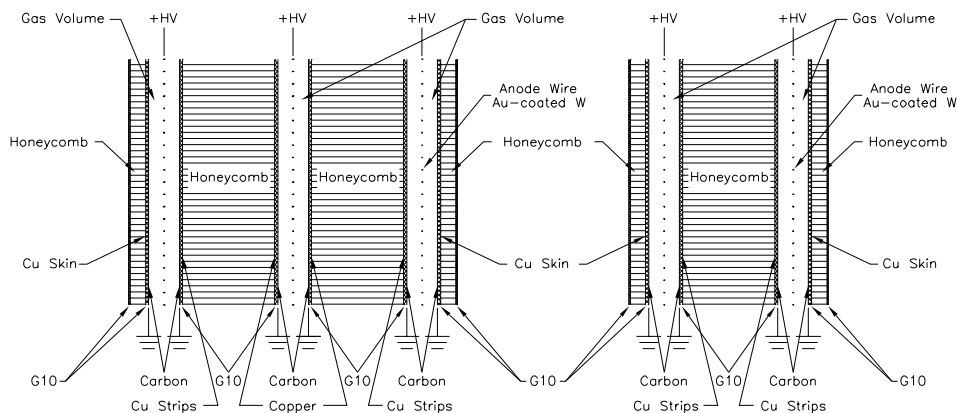


図 1.13: TGC(Thin Gap Chamber) Doublet と Triplet

## 第2章 ATLAS実験のトリガとデータ収集システム

### 2.1 トリガとデータ収集システムの構成

#### 2.1.1 トリガシステム

LHC 加速器による 40.08MHz のバンチ交差によって、ATLAS 検出器の衝突点で発生する陽子陽子の衝突率は 1GHz にも上る。これらの多量なイベントの中から目的とする物理現象のイベントを同定し、その現象由来のシグナルを捜し出す必要がある。トリガシステムは、これらの膨大なイベントの中からオンラインで、目的とする物理を含む可能性のあるイベントのみを、データ収集システム (DAQ) の対象にさせる判断を行っている。最終的に DAQ システムに送られるイベントレートは、データストレージの容量の関係で 100Hz 程度にすることが要請されている。

この目的を達成するために、ATLAS 実験のトリガシステムは 3 段階のステージに分けて順次イベントレートを落していく設計になっている。

##### 2.1.1.1 トリガシステムのスキーム

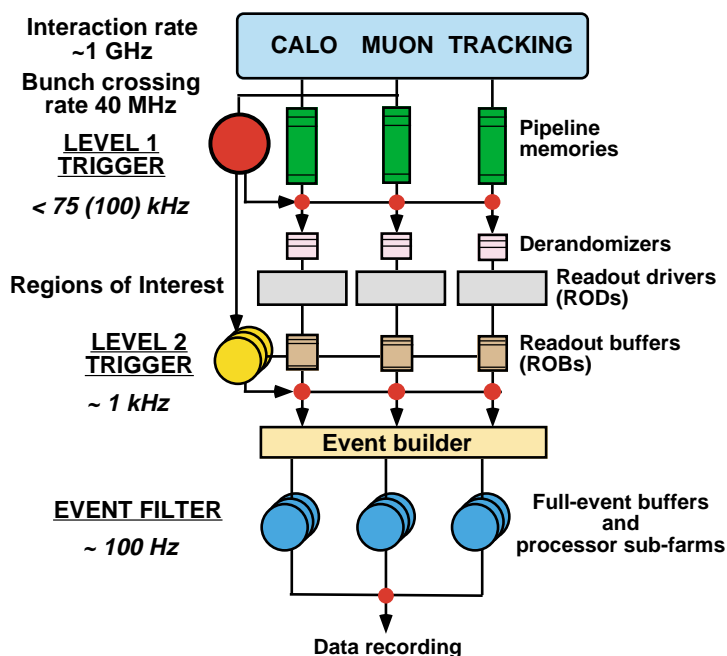


図 2.1: ATLAS 実験のトリガシステムのスキーム

ATLAS 実験のトリガスキームを図 2.1 に示す。図のように、Level-1 トリガ (LVL1)、Level-2 トリガ (LVL2)、イベントフィルタ (EF) の 3 ステージで構成されている。

LVL1 では、LHC クロック 40.08MHz の入力を 最大 75kHz のイベントレート<sup>1</sup>に落す。LVL1 でのイベントの選別に用いるのは、トリガ用ミュオン検出器 (TGC, RPC) と全てのカロリメータのエネルギー情報を用い、内部検出器 (Tracking) は用いない。LVL1 では、バンチ交差の時点から  $2.0\mu\text{s}$  以内に<sup>2</sup> L1A(Level-1 Accept) 信号を生成する判断を行う。

LVL2 トリガは全検出器領域のデータを対象としてトリガ処理を行うのではなく、LVL1 トリガで、High- $P_T$  と判断されたミュオンやジェット、電磁クラスタの存在する領域 (RoI: Region of Interest) を設定し、その領域についてトリガ処理を行っている。LVL2 トリガでは、RoI 内部について、内部検出器からのデータも含め、各検出器系からの完全な位置情報と精度を持つデータを使い、イベントレートを 約 1kHz まで落している。LVL2 トリガシステムでは並列処理が行われ、1 イベントの処理に必要なレイテンシ (処理時間) はおおむね 10ms である。

LVL2 で選出されたイベントに対しては、Event Builder でイベントの再構築がなされた後、イベントフィルタ (EF) に送られる。イベントフィルタでは、完全な位置情報と精度を持つ全データ (約 2MB/event) を用いた、オンライン最後のイベント選別が行われる。EF では通常オフラインで用いられるアルゴリズムや解析手法が適用され、コンピュータ・ファームによって並列処理が行われる。最終的に 250Hz までイベントレートを落す予定である。<sup>3</sup>したがって、データストレージには 500MB/s の記録が行われる。なお、1 イベントの判定におよそ 1s かかる。

### 2.1.2 データ収集システム (DAQ)

データ読み出し用の回路として、各検出器の on-detector electronics には FIFO 構造のメモリが 2 種類備えられている。一つは、Level-1 Buffer と呼ばれるパイプラインメモリ、もう一つは、Derandomizer である。

検出器から生成されたデータは同期化回路を経て、Level-1 Buffer に順次蓄えられ、LVL1 トリガから L1A 信号が発行されるのを待つ。L1A が各検出器に送られると、対象となるバンチのデータは Derandomizer に進み、L1A の来なかったバンチに属するデータは FIFO からそのまま捨てられる。

Readout Driver(ROD) は、Derandomizer にあるデータを一ヶ所の Readout buffer(ROB) に送る。

### 2.1.3 ディテクタコントロールシステム (DCS)

ATLAS 実験は巨大な検出器であり、全体を安定して運転するには適切な検出器の制御・保守の手段が必要になる。加えて、ATLAS 実験では、実験中は放射線レベルが高いために、実験ホールに立ち入ることが出来ない。DCS(Detector Control System) は、検出器の制御と監視を統一的行うシステムである。DCS では、検出器の運転に必要なパラメータの設定や電源システムの監視、ガスシステムの流量や圧力の監視、温度測定などを行うほか、LHC からルミノシティや放射線に関するデータを取得する。

## 2.2 Level-1(LVL1)トリガシステム

Level-1 トリガシステムでは、カロリメータとミュオントリガ検出器からの情報を元にトリガの判断を行う。LVL1 トリガシステムでは、図 2.2 のように、カロリメータとミュオントリガ検出器の DSL(Detector Specific Logic) ではそれぞれ独立にトリガ処理を行い、その結果からセントラルトリガプロセッサ (CTP) が最終的な判断を行い、L1A を発行する。これは、TTC(Timing, Trigger and Control distribution) を経由して各ディテクタに送られる。

<sup>1</sup>100kHz にアップグレード出来るよう、各検出器及び DAQ システムと LVL2 システムは設計しなければならないとされている。

<sup>2</sup>緊急時は、 $2.5\mu\text{s}$  まで許されるように  $0.5\mu\text{s}$  の予備時間が確保されている。

<sup>3</sup>図 2.1 では、Event Filter からの出力は 100Hz になっているが、現在の設計では 250Hz に変更されている。

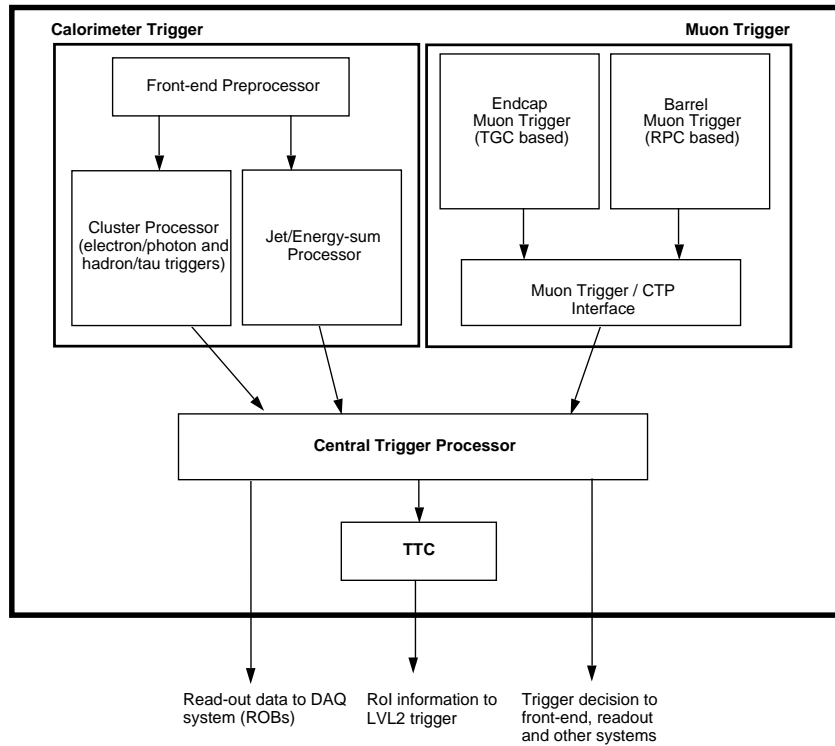


図 2.2: Level-1 トリガシステムのスキーム

以下では、LVL1 トリガシステムの中でミュオントリガシステムに関係の深い CTP と TTC に絞って説明を行う。

### 2.2.1 Cenrtal Trigger Processor(CTP)

CTP は、カロリメータとミュオントリガ検出器の DSL からのトリガ候補を総合して、LVL1 トリガの最終的な結論である L1A を発行する。

CTP は、各々のトリガ条件に合致するイベントに対しては L1A を出力する他にも、そのトラックの情報を LVL2 トリガと DAQ にも送っている。このデータは、LVL2 システムでは RoI の決定の材料に使われ、DAQ ではトリガシステムのモニタに利用される。

図 2.3 は、CTP のブロックダイアグラムである。DSL から MUCTPI(Muon CTP Interface) などのインタフェース回路を経て送られて来たトラック情報は、中央上段の Synchronisation & Alignment に入力され、ここで各 DSL からの信号の遅延時間を調節し、同一のバンチに揃えられる。引続き Trigger Formmation ブロックで最終的なトリガ判定が行われる。ここでは、最大 96 個のトリガ項目を設定可能である。さらにその結果は、Prescalers によってプリスケールすることも可能となっている。

CTP は、これらのことを 4clock のレイテンシで行う。

### 2.2.2 Muon CTP Interface(MUCTPI)

MUCTPI は、TGC と RPC の DSL からの信号を受信する。TGC 及び RPC の最終段の Sector Logic 回路からは Trigger Sector 単位で、トリガの候補を選出している。この MUCTPI では、全ての Trigger Sector からのトリガ候補について集め、境界付近でのダブルカウントを解消したあと、最終的なミュオントラックの候補を CTP に送信している。MUCTPI でも、CTP 同様にミュオンのトラックの情報

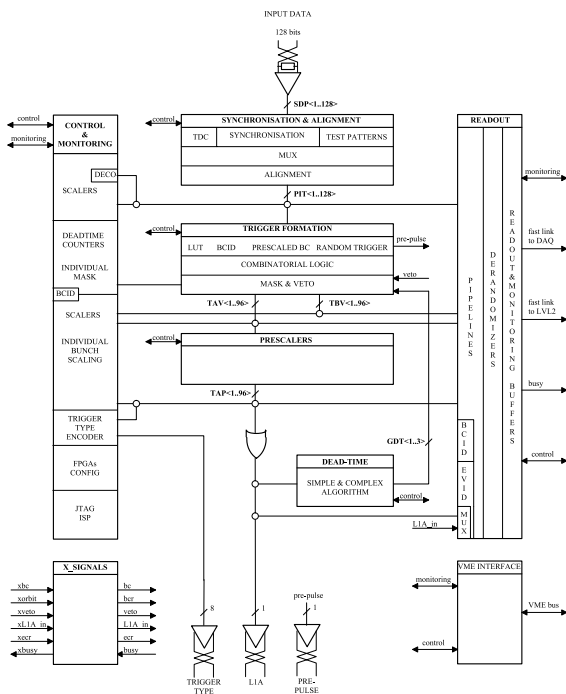


図 2.3: CTP(Central Trigger Processor) のブロックダイアグラム

を LVL2 トリガシステムと DAQ に送っている。図 2.4 は、Sector Logic から送信された信号を受信する MUCTPI のフロントエンド回路である、MIOCT の写真である。合計 16 台の MIOCT で全検出器をカバーする。



図 2.4: MUCTPI のフロントエンド (MIOCT) のプロトタイプ

### 2.2.3 Timing, Trigger and Control distribution(TTC)

TTC(Timing, Trigger and Control distribution) は、ATLAS 検出器の中で、分散されて配置されている各ディテクタに各種の制御信号を、光ファイバによって配信するシステムである。TTC で送られる制御信号には、40.08MHz の LHC クロック、Level-1 Accept(L1A)、バンチカウンタのリセット信号 (BCR)、イベントカウンタのリセット (ECR) などがある。

## 2.3 前後方ミュオントリガシステム

### 2.3.1 前後方ミュオントリガシステムの構成とレイアウト

EndCap 部 ( $1 < |\eta| < 2.7$ ) をカバーするトリガ用検出器である TGC は図 2.5 のように、配置されている。図中の EI(EndCap Inner), FI(Forward Inner), M1, M2, M3 の文字で示されているのが TGC の位置を示すステーション名である。

EI, FI のステーションは、トロイダルマグネットよりも前方に位置する小型の TGC が設置されている。これは、衝突点以外から M1, M2, M3 の TGC に入射するバックグラウンド粒子の影響を排除する目的を持っている。(図 3.9 参照)

また、M1 のステーションには Triplet の TGC、M2 と M3 のステーションには Doublet の TGC が配置されている。特に M3 の Doublet は、ミュオンの運動量測定のための基準となるため、Pivot plane と呼ばれる。(これに対して、M2 は Middle Doublet と呼ばれる)

さて、TGC のカバーする領域に入射するミュオンは、エンドキャップ部のトロイダルマグネット (図 1.8 参照) の磁場により偏向を受ける。図 2.5 中に表されているように、ミュオンの横運動量  $P_T$  の値によって偏向角が異なって来る。TGC では、この偏向角の値からミュオンの横運動量  $P_T$  を測定する。

Pivot Plane の入射点とビーム衝突点とを結ぶ直線を基準線として、Middle Doublet および Triplet の TGC で検出された点がどれだけずれているかを測定する事によって行われる。なお、Pivot Plane の入射点の座標は、subsector の座標値で与えられる。(2.3.2 節参照)

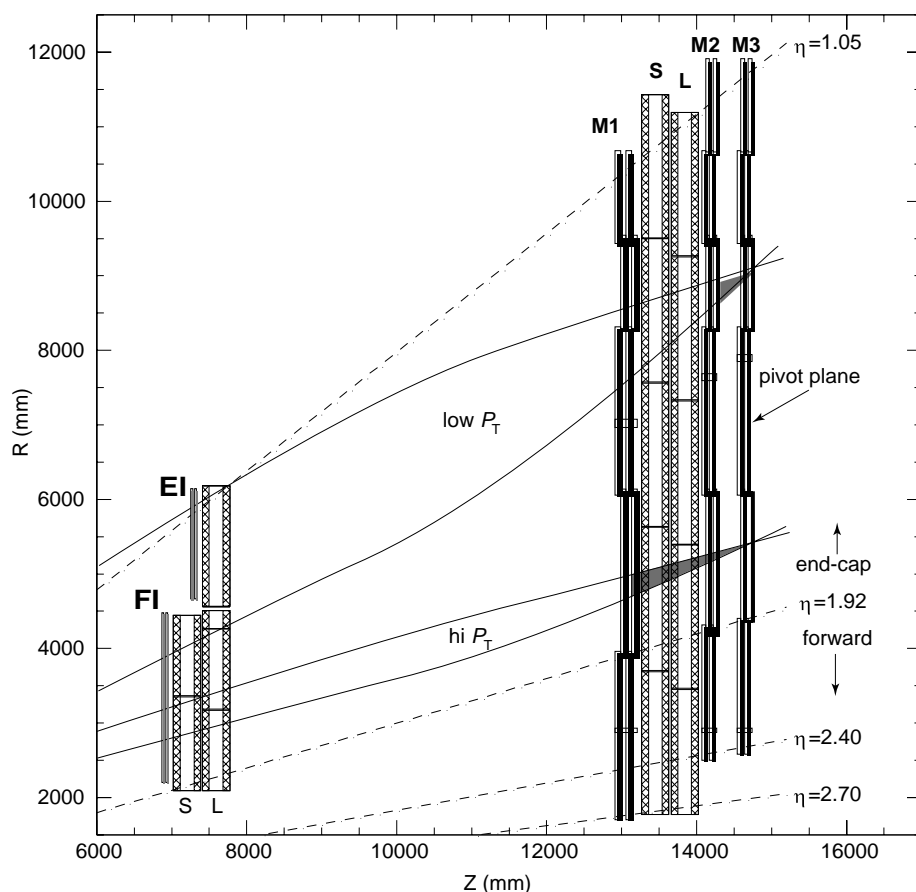


図 2.5: TGC の配置の R-z 平面図

### 2.3.2 Trigger Sector

図 2.6 は、M3 のステーションにある TGC Doublet によって構成される Pivot Plane の Octant<sup>4</sup>を表したものである。Pivot Plane は、EndCap 領域 ( $|\eta| < 1.9$ ) と Forward 領域 ( $|\eta| > 1.9$ ) に分けられる。Octant を  $\phi$  方向に、EndCap 領域 では 6 分割、Forward 領域では 3 分割した区画を Trigger Sector と呼ぶ。Trigger Sector の  $\phi$  方向の幅は、TGC の幅と等しい。また、EndCap 領域の Trigger Sector は 5 台の TGC をカバーし、Forward 領域では、1 台の TGC をカバーする。

また Trigger Sector は、subsector と呼ばれるトリガ処理の最小単位に分割される。これは、TGC からの入力 8ch 分に相当し、位置情報の最小分解能である。EndCap Trigger Sector は、R 方向に 37 分割、 $\phi$  方向に 4 分割され、Forward Trigger Sector は、R 方向に 16 分割、 $\phi$  方向に 4 分割される。

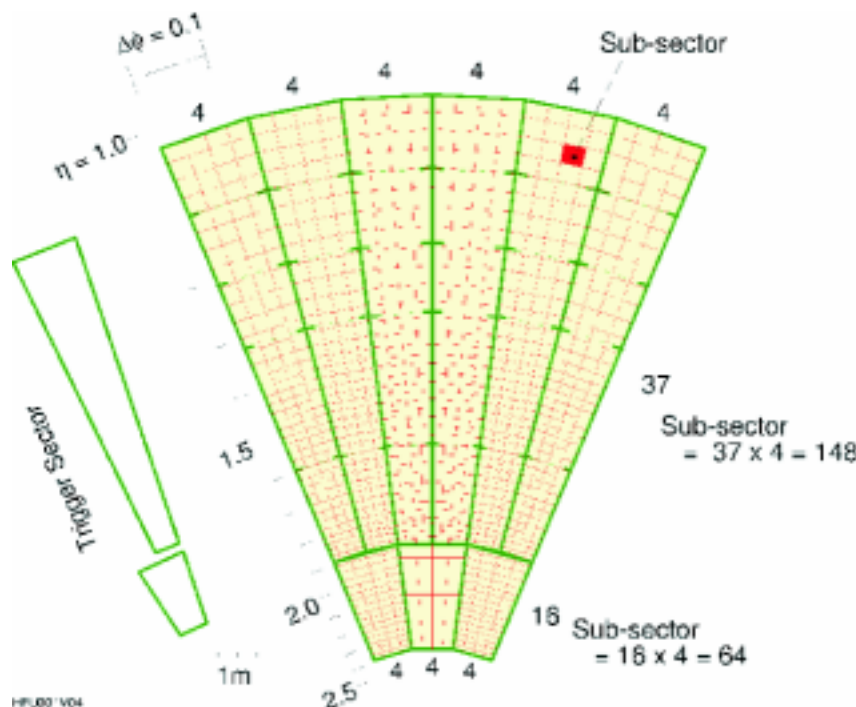


図 2.6: Trigger Sector

### 2.3.3 R- $\phi$ Coincidence Logic の特徴

Sector Logic 回路は、それぞれ R 方向 (Wire) と  $\phi$  方向 (Strip) の Hi- $P_T$  モジュールから送られて来る Pivot Plane における入射点と track の曲がりの情報を用いて、入射ミューオンの横運動量 ( $P_T$ ) を構築し、6 段階の横運動量の閾値<sup>5</sup>に分類する機能を持つ。

この Sector Logic の機能ブロックは、R 方向 (Wire) と  $\phi$  方向 (Strip) の Hi- $P_T$  モジュールからの情報を統合して入射ミューオンの横運動量を構築することから、R- $\phi$  Coincidence Logic と称される。本節では、R- $\phi$  Coincidence Logic が、直交する 2 つの信号 (R 方向と  $\phi$  方向) を用いなければならない理由、及びその特徴について考察する。

まず、R- $\phi$  Coincidence を行うに当たって考慮すべき事を挙げる。

<sup>4</sup>Pivot Plane の 1/8 の領域を Octant と呼ぶ。ATLAS 検出器が 8 回対称に作られていることから Trigger/DAQ の処理の区画として用いられる

<sup>5</sup>Hi- $P_T$ Matrix で判定される Low- $P_T$ , Hi- $P_T$  ごとに 3 段階の閾値を持つ。現在のところ、Low- $P_T$  では、最も低い閾値が 6GeV、Hi- $P_T$  では、最も低い閾値が 20GeV であることのみが決まっている。なお、Hi- $P_T$  ボードから出力される Low- $P_T$ , Hi- $P_T$  識別ビットについては、2.3.4 節を参照。



1. 横運動量 ( $P_T$ ) の 6 段階の閾値は、探索する物理により実験中に自由に変更出来なければならない。
2. Sector Logic 回路自体に厳しい latency の制約があり、複数 clock を費したくない。
3. さらに、横運動量 ( $P_T$ ) 測定用のトロイダル・マグネット群 (図 2.7 で赤色の部分) によって発生する磁場は均一磁場にはならないため、subsector の位置情報も測定のパラメータに含める必要がある。

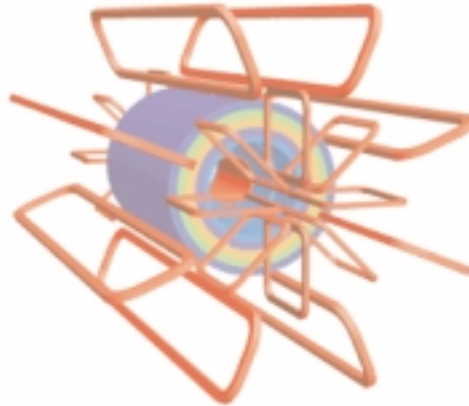


図 2.7: ミューオンの横運動量測定用トロイダル・マグネット群

図 2.8 は、ATLAS 検出器の衝突点から 6GeV のミューオンを  $\phi$  軸と  $\eta$  軸に対し均等になるように発射させた時、トロイダル・マグネットの磁場によって曲げられた後、TGC の Pivot Plane においてどの位置に入射するかを表したシミュレーションの図である。もともとは、均等な 6GeV のミューオンだったが、入射点が収束していたり、ほとんどミューオンが来ない領域があったりする。

これは、トロイダル・マグネットが 8 回対称と、かなりラフな構造であるのに加えて、前後方部のトロイダル・マグネットとバレル部のトロイダル・マグネットの磁場が干渉した結果、磁場は複雑に乱れており、ミューオンはそのなかを通過するため、座標毎に違う曲げられ方をするためである。

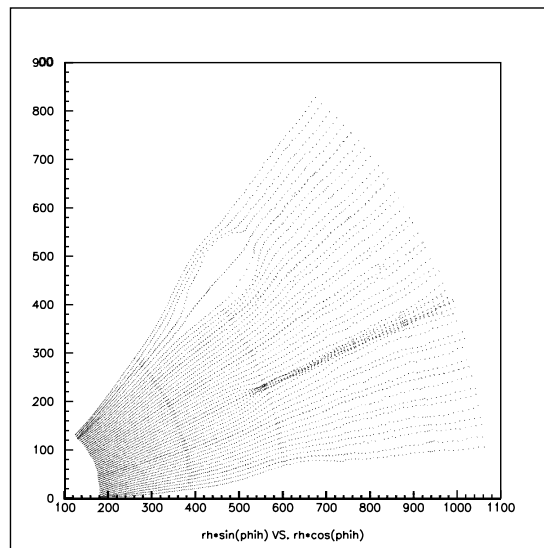


図 2.8: 均等な 6GeV のミューオンをトロイダル・マグネットの磁場中に入射した時の Pivot Plane における通過位置

このように、閾値が可変である上に、磁場によって複雑な曲げられ方をしたミューオンの track から運

動量を測定するためには、ディテクタシミュレーション及びトリガシミュレーションを行い、 $R, \phi$  それぞれの曲げられ方と hit のあった subsector の位置情報と、(6 段階の) $P_T$  値との対応マップを作成することが必要となる。この対応データは横運動量 ( $P_T$ ) の閾値の値に依存する。

これらの条件を満足するために、 $R$ - $\phi$  Coincidence は入力データをインデックスとする、完全な Look-Up Table(LUT) として実装することにした。

一方、このような方式を取ったおかげで、 $R$  方向と  $\phi$  方向の入力が異なったミュオンの横運動量の値となるような場合 - ノイズや fake hit など - については、track 候補から排除する事が可能になった。

### 2.3.4 トリガエレクトロニクス

図 2.9 に、前後方ミュオントリガシステムのエレクトロニクスのブロック図を、図 2.10 に信号線の配線図を示す。

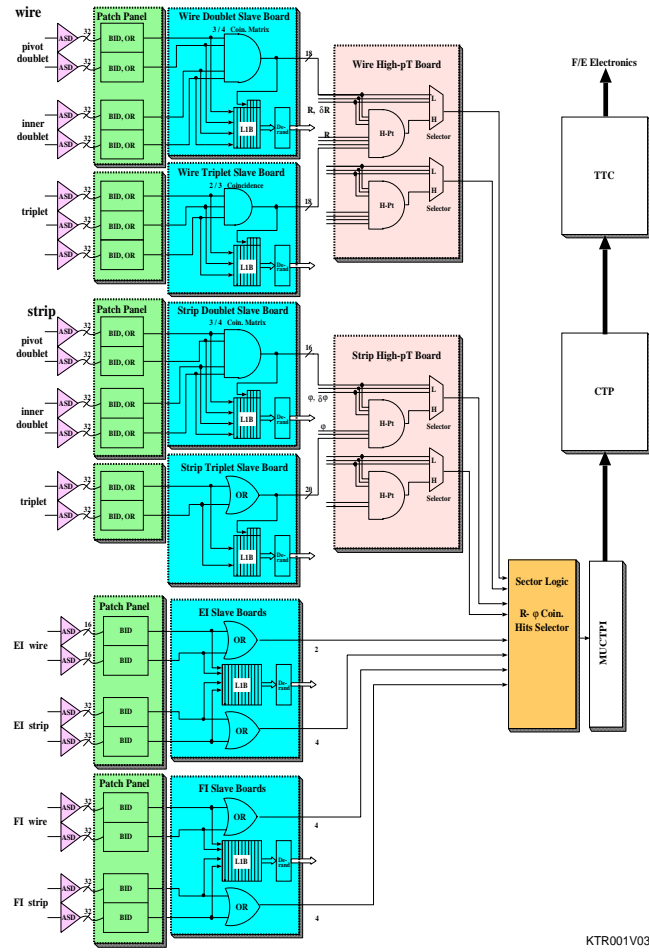
この図から分かるように、M1(Triplet), M2(inner Doublet), M3(Pivot Doublet) の各ステーションの TGC からの信号は、Hi- $P_T$  ボードまで Wire( $R$  方向) と Strip( $\phi$  方向) に分けて取り扱われる。また、EI, FI ステーションの TGC もそれぞれ個別に扱われる。これらの入力を Trigger Sector の領域を単位として統合するのが、Sector Logic 回路である。

**Wire** TGC の Wire 入力、ASD で増幅整形が行われた後 Patch Panel で、LHC クロックとの同期化 (BCID)、TGC 境界を解消するための OR が行われる。Doublet Slave Board は、Pivot Doublet(M3) と Inner Doublet(M2) の入力から、TGC のノイズやバックグラウンド粒子による誤動作による fake hit を排除する目的で、3/4 Coincidence が取られる。3/4 Coincidence とは、Pivot Doublet と Inner Doublet の計 4 枚の TGC のうち、3 枚以上の TGC が粒子を検出している場合、hit 有りともみならず Coincidence である。4 枚全てが検出していることを要求しないのは、TGC の検出効率の関係でたまたま反応しなかった TGC があつた場合でも、高いトリガ効率を確保するためである。さらに Double Slave Board 内の Low- $P_T$  Matrix により、Pivot Doublet と Inner Doublet とを用いたミュオンの曲がり具合 ( $\Delta R$ ) が  $\pm 7$  の範囲で計測が行われ、その結果は Hi- $P_T$  Board に送られる。なお、Low- $P_T$  というのは、比較的低い  $P_T$  のミュオンを対象にしたトリガ項目に対応する閾値を越えた、ミュオンの事を指す用語であり、トロイダル磁場により比較的大きく曲げられるため、M1 のステーションに位置する Triplet は使わない。(図 2.5 を参照) 一方 Triplet からの入力は、Triplet Slave Board で 2/3 Coincidence (3 枚の Triplet TGC のうち、2 枚以上の TGC の粒子検出により hit 有りとする) が取られ、そのまま Hi- $P_T$  Board に送られる。

Hi- $P_T$  Board では、Doublet Slave Board で Low- $P_T$  とされた hit 情報は、Low- $P_T$  という識別ビットを付加して、Sector Logic に送っている。それ以外の場合、2 つの Slave Board からの情報を用い、Hi- $P_T$  Matrix により、Pivot Doublet と Triplet によるミュオンの曲がり具合 ( $\Delta R$ ) の測定を  $\pm 15$  の範囲内で計測する。(図 2.5 を参照) Hi- $P_T$  Matrix からの hit 情報は、Hi- $P_T$  という識別ビットが付加されて、Sector Logic に送られる。

**Strip** TGC の Strip からの入力は、Wire の場合と基本的に同様の過程を経て処理される。しかし、Triplet からの Strip 読みだしが 3 層ではなく、2 層の読みだしになっているほか、ミュオンの曲がり具合 ( $\Delta \phi$ ) を表現するビット数が異なっている。ここでは、異なっている点を挙げる。

Doublet Slave Board で計測される Low- $P_T$  のミュオンの曲がり具合 ( $\Delta \phi$ ) は、 $\pm 3$  の範囲で計測され、Hi- $P_T$  Board での Hi- $P_T$  の  $\Delta \phi$  は、 $\pm 8$  の範囲で計測される。また Triplet Slave Board では、2 層の読みだしであるため、1/2 Coincidence 即ち OR 処理を行っている。



KTR001V03

図 2.9: 前後方ミュオントリガシステムの構成

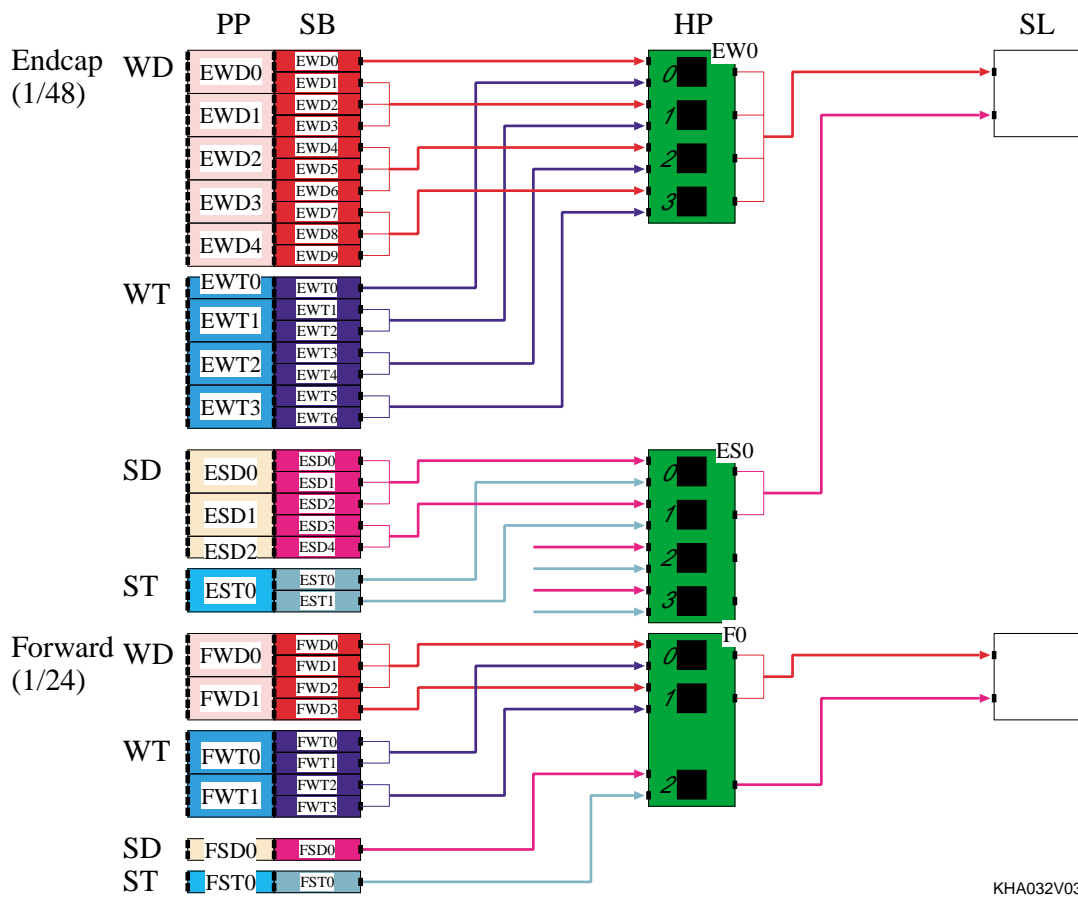


図 2.10: Patch Panel(PP), Slave Board(SB), Hi-PT(HP), Sector Logic(SL) との間の信号の接続関係

### 2.3.5 モジュールの配置

各々のモジュールは、実験ホール内 (UXA15) で検出器のそばに置かれるものと、実験ホールの外にあるエレクトロニクス室 (USA15) に設置されるものの2種類がある。実験中の実験ホール内の放射線強度は高いため、保守などのために立ち入る事は出来ず、DCS 経由での制御のみ可能となる。これに対し、USA15は放射線の影響を受けないため、実験期間中でも自由にアクセスすることが可能である。

図 2.11 は、前後方ミュオントリガシステムの各モジュールの配置を示している。ASD は TGC 側面のコネクタに直接接合しており、ここから伸びた配線は TGC ホイールの背面に設置される PS-Pack に届いている。PS-Pack には、Patch-Panel と Slave Board が搭載されている。そして Hi-PT モジュールおよび、読みだし系の回路である Star Switch(2.3.7.3 節を参照) は、TGC ホイールの再外周部に設置される。一方、Sector Logic と Star Switch の送信先である Local DAQ Master は、USA15 に設置される。Hi-PT モジュールや Star Switch の設置されている TGC ホイールの再外周部と Sector Logic や Local DAQ Master の設置される USA15 の間には、約 90m の距離がある。この部分の通信には、光ファイバケーブルを用いる。

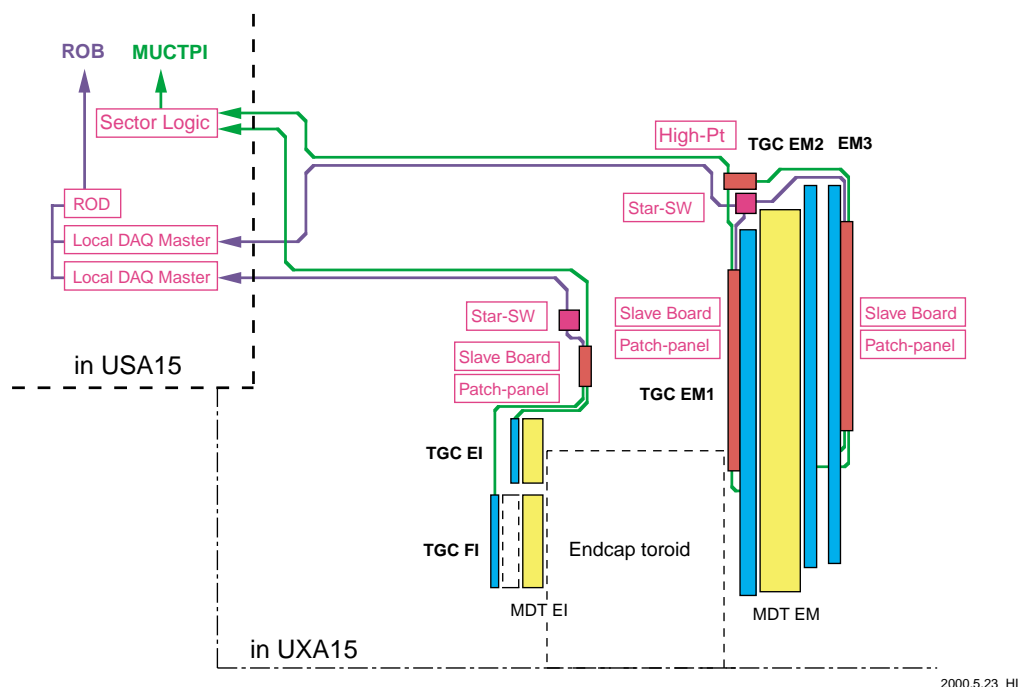


図 2.11: 各モジュールの設置場所

### 2.3.6 Latency

Latency は、ATLAS LVL1 トリガにとって極めて重要な値である。2.1.1.1 節で述べたように、LVL1 トリガ全体の Latency は標準で  $2.0\mu\text{s}$  である。これは、衝突点で陽子陽子衝突が起こった瞬間から、L1A が各ディテクタに分配されるまでの全時間である。この中には、トリガ回路の処理時間だけでなく、反応点から検出器までの粒子の飛翔時間 (TOF: Time Of Flight) や、トリガモジュールを結ぶ配線による伝搬遅延も全て含まれている。

以下の表は、この  $2.0\mu\text{s}$  の全 Latency のうち、TGC を扱う前後方ミュオントリガシステムの出力部まで消費される Latency をまとめたものである。

遅延要素	Latency	Total
TGC までの TOF(最大 63ns)	3	3
TGC の反応時間 (25ns)	1	4
ASD ( <10ns)	1	5
Patch Panel までの配線 (10m)	2	7
BunchID 及び OR 処理 (Patch Panel)	2	9
Slave Board までの配線	0	9
Delay 調整 (Slave Board)	1	10
3/4 Coincidence および 2/3 Coincidence	2	12
LVDS Tx	1.5	13.5
Delay 調整 (Hi-P <sub>T</sub> )	1	19
Hi-P <sub>T</sub> マトリクス	2	21
G-Link Tx	1	22
USA15 までの光ファイバ (90m)	18	40
G-Link Rx	2	42
Sector Logic	7	49
MUCTPI への配線 (5m)	1	50

### 2.3.7 構成要素の補足的説明

本節では、2.3.4 節でおこなった、前後方ミュオントリガシステムを構成する各モジュールについての説明で、不足した補足的情報を与える。

#### 2.3.7.1 Amp-Shaper-Discriminator(ASD)

ASD(Amp-Shaper-Discriminator) Board は、TGC 側面に直接取り付けられる。一枚の ASD Board には、4 入力の ASD ASIC が 4 個実装されており、16 入力を持つ。ASD は、TGC からの信号を、プリアンプで増幅し整形した後、指定された閾値で弁別し、LVDS 信号で Patch Panel に送る。ASD の動作電源、閾値、テスト入力は Patch Panel から供給される。

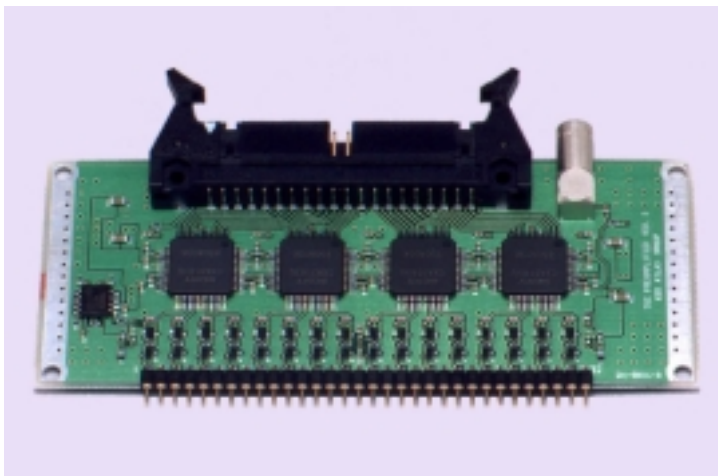


図 2.12: ASD Board

### 2.3.7.2 Hi-PT

Hi-PT ASIC から Sector Logic におくられる hit 情報は、トリガマトリクスの構造上の理由により、隣接する 2 subsector から 1hit しか出力されない。さらに、出力に G-link という規格の光ファイバを用いており、配線にかかるコストの問題から、1つの Hi-PT ASIC からは、最大 2hit に制限する Track Selector が装備されている。図 2.13 は、それぞれの Hi-PT ASIC における Track Selection について示している。

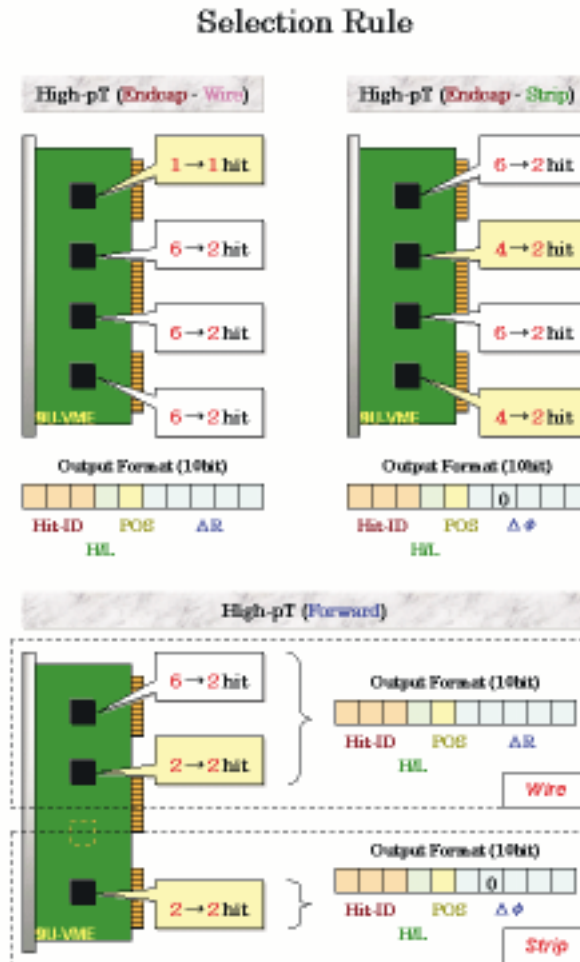


図 2.13: Hi-PT における Track Selection

### 2.3.7.3 Star Switch

Star Switch は、TGC の読み出し系に用いられる、Star 型のトポロジーを持った Switch である。TGC は ATLAS 検出器の両サイド (A, C) の EndCap 部に分かれて設置されており、それぞれの Slave Board は分散されて配置されている。これらから、故障耐久性を確保しながら効率良く読み出し、USA15 の区画にある ROD(図 2.11 参照) まで伝送する<sup>6</sup>方法として、Star 型のトポロジーが採用された。

<sup>6</sup>正確には、Local DAQ Master(LDM) が Star Switch からの入力を受け取り、ROD が LDM からデータを読み出して ROB に送信する。

## 第3章 Sector Logic

第3章は、本論文の核心部である。これまでに前後方ミュオントリガシステムとそこでの Sector Logic の役割を見て来た。まず、第3.1節では Sector Logic の基礎設計をまとめる。ここで、各機能ブロックである Track Selection Logic と  $R\text{-}\phi$  Coincidence Logic の詳細が述べられる。次の第3.2節と第3.3節では、これらの機能ブロックについて機能検証を行うために製作したプロトタイプ製作とその評価をまとめた。第3.4節では、Sector Logic のほぼ全機能を実装したプロトタイプモジュール Prototype-0 の設計について述べる。

### 3.1 Sector Logic の 基本設計

先に述べたように、Sector Logic は1つの Trigger Sector を対象に、入射ミュオンの横運動量 ( $P_T$ ) 測定 ( $R\text{-}\phi$  Coincidence) と、その値をもとにした hit の選別 (Track Selection) を行う回路である。Sector Logic の設計にあたっての制約条件は次のようにまとめられる。

1. LHC の運転周波数 40.08MHz(LHC クロック) に同期して、デッドタイムレスで動作しなければならない。つまり、連続したバンチの入力に対しても正しい結果を出力しなければならない。
2. Hi- $P_T$  モジュールからの信号を受信してから、MUCTPI に結果を出力するまでに許される latency は最大  $8\text{clock}(=200\text{ns})$  であり、一定 clock でなければならない。[latency 数固定]
3. 入射ミュオンの運動量測定で用いられる 6 段階の閾値は、探索する物理により実験中に自由に変更が出来なければならない。

1. の条件は、必要とされるスループットを規定する。処理の洩れが無いようにするには、40.08 M/s 以上の処理能力が必要となる。

2. の条件は、短時間に処理を終了させることが要求される LVL1 トリガシステムの CTP(Central Trigger Processor) において、この Sector Logic からの信号と TGC 以外のディテクタからの信号とを統合して各バンチごとのトリガ信号を生成するためには、レイテンシが一定値(かつ短時間)でなければならないからである。

このようなスループットや latency の制約を持った回路を設計する場合、次に示すような 2 通りの設計方針がある。

- CPU や DSP などを並べた並列処理
- 高速な論理回路によるパイプライン構造

前者の CPU 等を用いる方法は、プログラム処理のため柔軟性が高く、複数個で並列処理することにより大きなスループットを扱う事が出来る。そのため、ATLAS 実験でも Event Builder のようなシステムには、この方法が採用されている。しかし、複雑な処理になるとステップ数が増大するため、今回のような latency の制約があった場合、並列処理では対応出来ない。クロックの高速化には限界がある。

後者のパイプライン構造は、システムクロックによって動作するフリップフロップ (FF) と、その間に位置する組み合わせ回路によって構成される。全体を LHC クロック同期させて動作させるため、スループットの条件は自動的に満たさせる。問題は、組み合わせ回路部分の遅延時間が LHC クロックの周期以



内に出来るかである。パイプラインの段数を増やし、1段あたりの処理内容を減らせば、遅延時間を減らす事が出来る。しかし、latency の制約があるため無闇に増やすことは出来ない。

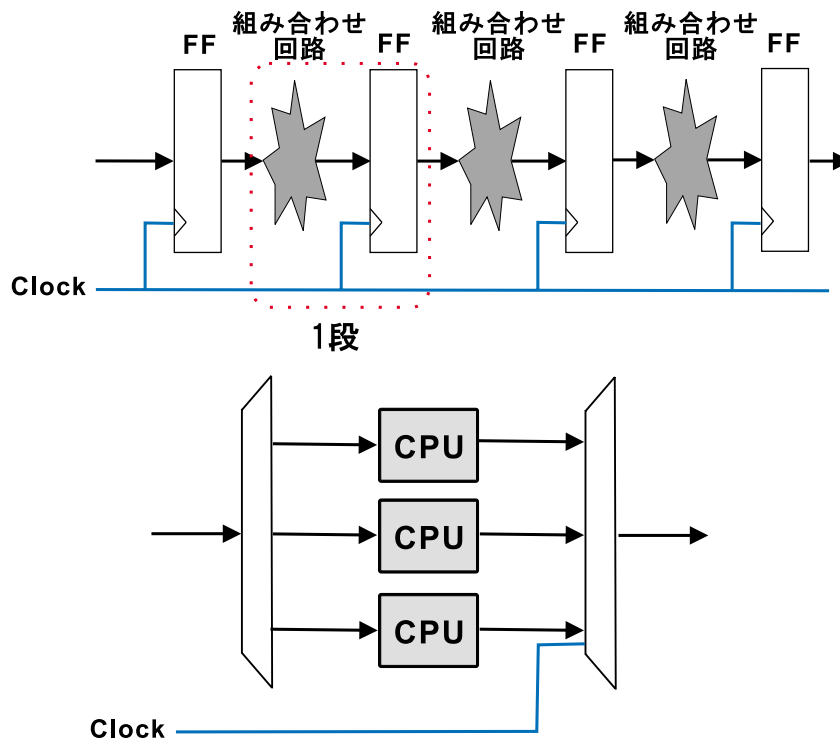


図 3.1: パイプライン構造 (上) と CPU 等の並列処理

また、Trigger Sector 内にあるミュオン track が発生する単位の subsector は Forward 部で 64 個、Endcap 部で 148 個と相当の数であり、これらを対象にして track の選別を行う Track Selection Logic はかなり規模の大きい処理になる。したがって、パイプライン方式を採用する場合、1段あたりに掛かる遅延時間をクロック周期内に押え込むことが鍵となる。他方、並列化した CPU 等を用いる場合、相当のステップ数が必要となる。200ns という latency 制約は一般の CPU 等にとってはかなり厳しい条件である。

ところで、Sector Logic を特長付ける R- $\phi$  Coincidence は、2.3.3 節で考察したように LUT(Look-Up Table) を用いた設計にする必要がある。

LUT とは、入力信号の取り得る全てのパターンについて、予め処理内容を計算したデータをメモリに格納し、入力信号をインデックスとしてメモリを参照することによって結果を得る方法である。LUT では、処理時間は入力信号の内容に依存せず一定のため、パイプライン構造の要素に適している。また、入力信号数が多く LUT のサイズは大きくなるため、全体を並列化して個別に LUT を持った場合、高コストになり得る。

これらの条件を考慮した結果、Sector Logic 回路はパイプライン構造を用いて設計することに決定した。

### 3.1.1 Sector Logic の構成

単一のパイプライン構造を持つことにしたので、前段の回路である Hi- $P_T$  モジュールからの入力を受け取ってから 2 個の track 候補を出力するまで、順次連続して処理を行う。そのことから、図 3.2 のような構成になる。

ここで R- $\phi$  Coincidence Logic と Track Selection Logic は、それぞれ複数段のパイプラインとなっている。

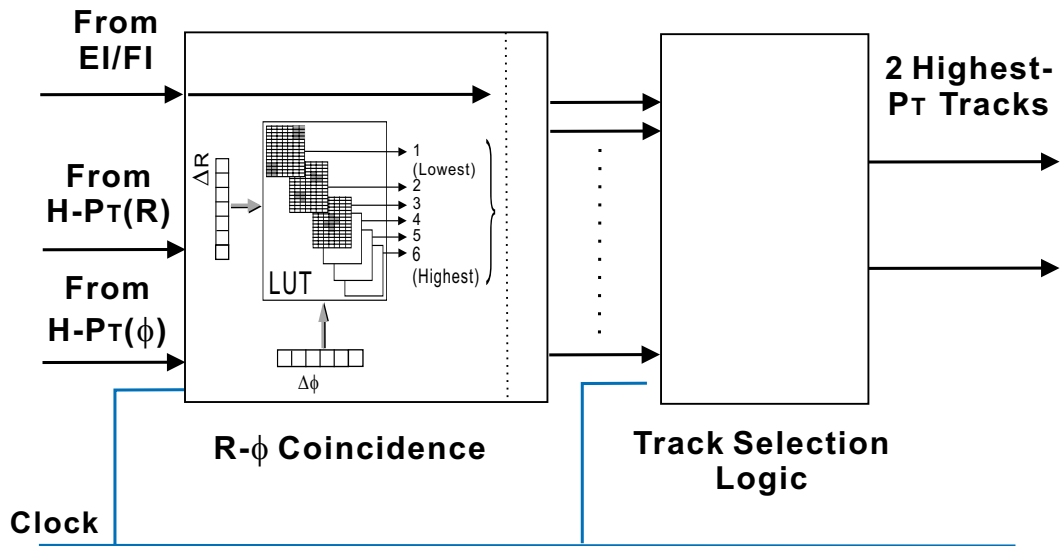


図 3.2: Sector Logic のブロックダイアグラム (概念図)

前段の R- $\phi$  Coincidence Logic では、R, $\phi$  各々の Hi-P<sub>T</sub> モジュールからの入力から 6 段階の運動量の閾値別の hit を LUT(Look-Up Table) を用いて生成する。EI/FI チェンバからの信号は、この LUT の出力をさらに選別する。

後段の Track Selection Logic は R- $\phi$  Coincidence Logic で生成された複数の hit から、運動量の値の大きいものから順に 2 個の候補を選出する Priority logic である。

これらの実装の詳細については、3.1.4 で述べる。

### 3.1.2 Sub Sector Cluster(SSC)

2.3.2 節で述べたように、Trigger Sector は subsector<sup>1</sup>と呼ばれる領域に分割されており、R、 $\phi$  各々の Hi-P<sub>T</sub> モジュールからは hit の subsector 単位の座標値 (R,  $\phi$ ) と hit の軌跡の曲がり ( $\Delta R$ ,  $\Delta\phi$ ) が送られて来る。

R- $\phi$  Coincidence Logic では、これらの入力を用いて subsector 単位で運動量の測定を行わなければならない。しかしながら、Trigger Sector 内の subsector 数は Forward 部で 64 個 (=16(R)×4( $\phi$ )), EndCap 部では 148 個 (=37(R)×4( $\phi$ )) とかなりの数になる。そのため、全ての subsector で独立に運動量測定を行なった場合、後段の Track Selection Logic の入力数も subsector 数必要となり、その結果相当大きな複雑な回路となる。そうなると設計仕様を満たす速度で動作させることは困難になる。したがって何らかの方法で、運動量測定を行う数を減らさなければならない。

Hi-P<sub>T</sub> モジュールでは、R 方向、 $\phi$  方向ともに隣接する 2 つの subsector から 1 つの hit に絞られて出力される。そのため、隣接する 2 つの subsector をまとめて扱い、その中から 1 つのミュオン track の運動量測定を行わせる事が出来る。

これらのことから、subsector を R 方向に 2subsector,  $\phi$  方向に 4subsector の小さなブロックにまとめ、この単位で運動量測定を行う事にした。以下、このブロックを SSC(SubSector Cluster) と呼ぶことにする。(図 3.3 参照) このような大きさにした結果、SSC は R 方向の Hi-P<sub>T</sub> からは 1hit、 $\phi$  方向の Hi-P<sub>T</sub> からは、2hit 分の入力を持つことになる。そのため、 $\phi$  方向の Hi-P<sub>T</sub> からの 2hit の入力が同時に存在した場合が問題になる。しかし、これが実際に 2 つのミュオンが入射したことによるケースは稀であり、大半は  $\phi$  の片側を生じさせたミュオンは別の R hit に属しており、fake hit である事がシミュレーショ

<sup>1</sup>R(wire group),  $\phi$  (strip), それぞれ 8ch の入力に相当する。

ンにより判っている。

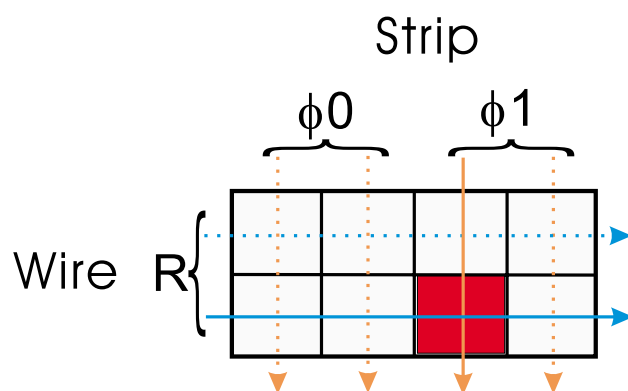


図 3.3: SSC の構造

そのため、1つの SSC からは 1個の track 候補しか出力しない。したがって、次のような手順で SSC からの track 候補が選出される。

1. R と  $\phi_0$ 、R と  $\phi_1$  それぞれで運動量を求める。ここで、ほとんどの fake track が排除される。
2. 2つの track 候補が残った場合、運動量の大きな track を SSC の出力とする。
3. もし運動量の値が同じ場合、どちらか一方の track 候補 を SSC の出力とする。

さて、EndCap 部 及び Forward 部の Trigger Sector は、それぞれ、19個、8個の SSC に分割されている。SSC と R 方向の Hi-P<sub>T</sub> 及び TGC との対応を図 3.4 に示す。

この図で分かるように、Hi-P<sub>T</sub> に送られるまでに R 方向の入力は、一定の入力チャンネル数毎に分けられて (Patch Panel, Slave Board を経由して) 別々の Hi-P<sub>T</sub> モジュールの ASIC chip に送られる。Endcap 領域では、4個の Hi-P<sub>T</sub>ASIC で、Forward 領域では、2個の Hi-P<sub>T</sub>ASIC で処理される。2.3.7.2 で述べたように、Hi-P<sub>T</sub>ASIC では隣同士の subsector から 1つの hit を選び出した後、track selector でカバーする範囲内の hit を 2個まで減らして Sector Logic に送っている。図中の Hi-P<sub>T</sub>ASIC に付いている吹き出しは、track selector への入力数と出力数を表したものである。

例えば、 $\eta=1.05-1.37$  について考えよう。この領域をカバーする R 方向の Hi-P<sub>T</sub> の ASIC は、4枚の TGC からの入力がある。図 3.4 によると、この領域に対応する SSC は、SSC1 から SSC6 までの 6個である。SSC の R 方向の大きさは、Hi-P<sub>T</sub> が 1hit を出力する大きさに取ったので、この Hi-P<sub>T</sub>ASIC では最大 6 hit を発生する。Hi-P<sub>T</sub>ASIC の track selector は、これから 2hit を選出する。そして、その結果が Sector Logic に送られる。一方、 $\phi$  方向には、この領域は 4個の TGC から構成されている。したがって、各 TGC から 2hit 発生するため、合計 8hit が発生する。この領域は 2個の  $\phi$  方向の Hi-P<sub>T</sub>ASIC によりカバーされており、それぞれで出力を 2hit に減らして Sector Logic に送っている。(3.1.4.1 節の 1.-(b) を参照)

### 3.1.3 TGC 境界の問題

Forward 部の Trigger Sector は Pivot Plane において 1枚の TGC Doublet であるのに対し、Endcap 部の Trigger Sector は 5枚の TGC Doublet で構成される。

残念な事に、R 方向に 4つある TGC の境界と subsector の境界はどれも一致していない。(  $\phi$  方向は、TGC の境界と Trigger Sector の境界が一致している。) したがって、図 3.5 のように TGC の境界を内部にもつ SSC が存在する。<sup>2</sup>

<sup>2</sup>この図で、TGC 境界の線は、overlap 部分を示している。overlap は SSC6 のみ 1ch で、残りは 2ch である。

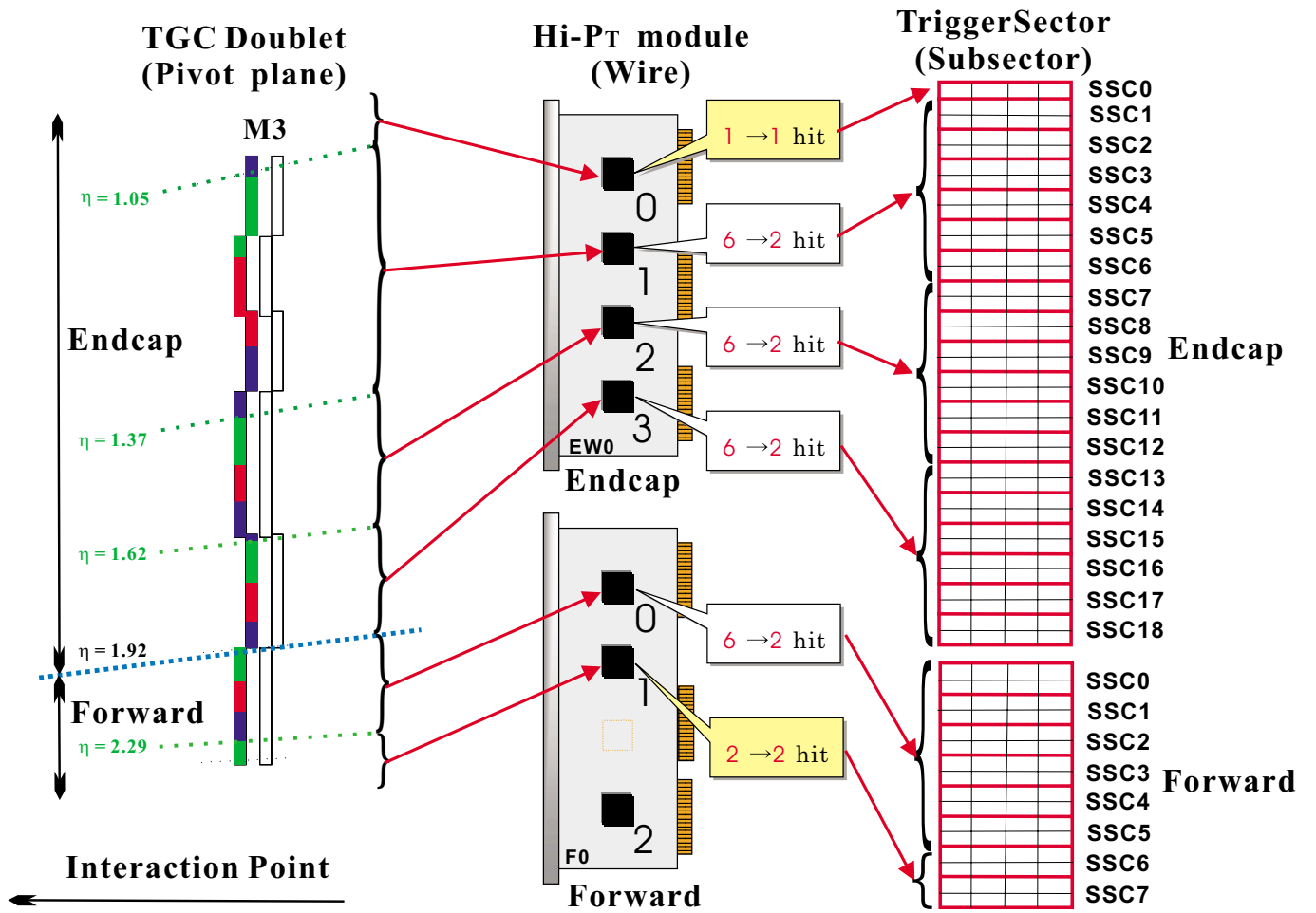


図 3.4: 入力の構造と Trigger Sector の SSC への分割

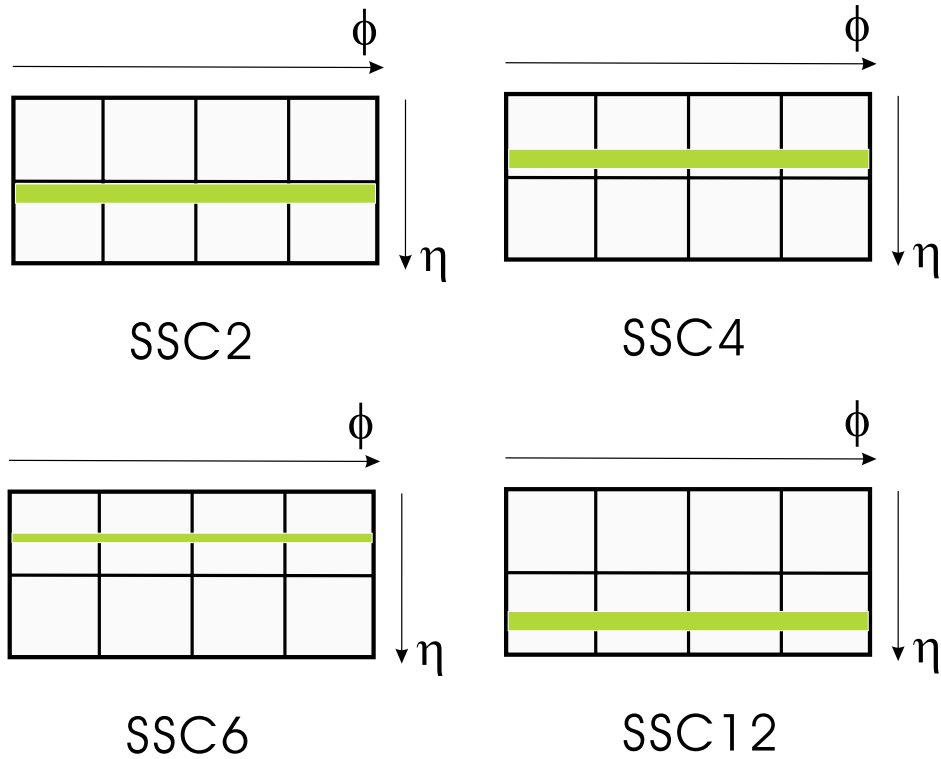


図 3.5: TGC 境界と、それを含む SSC

TGC の境界では、Patch Panel で overlap 部の OR を取ることによって double count を防止するとともに、全体が 1 つの大きな TGC として扱えるように配線されている。そのため、これより下流の Slave Board, Hi- $P_T$  では、これらの領域を特別扱いせず処理が行われている。したがって、この TGC 境界を内部にもつ subsector に R 方向の hit が存在した場合、どちらの TGC 由来の hit か (あるいは両方か) という事は分からない。

このことは、R- $\phi$  Coincidence Logic で  $\phi$  方向の情報と合わせて運動量測定を行う時に問題となる。 $\phi$  方向の信号は TGC ごとに読み出されるため、その subsector を含む SSC には同じ座標に対して  $\phi$  方向の Hi- $P_T$  からの入力が 2 個存在する。そのため、1 つの subsector で 2 通りの運動量測定を行い、その結果から 1 つのミューオンの track 候補を選ぶ必要がある。また、その SSC において、R 方向の入力が TGC 境界を含まない subsector 側にあった時、これらの subsector に属さない側の TGC から  $\phi$  方向の入力があっても、両者からは 1 つのミューオンの track を構成出来ないため、Track 候補から外さなければならない。

一方、SSC 単位で R- $\phi$  Coincidence Logic を回路に実装することを考慮すると、このような複雑な処理を行う TGC 境界を含んだ SSC をそのまま設計すると、他の SSC に比べて大きく複雑な回路になり、速度面でボトルネックになってしまい、好ましくない。そのため、何らかの方法でこれらの SSC の持つ複雑さを解消して、TGC 境界を持たない SSC と同等に扱えるようにすることが望まれる。

そこで、図 3.6 に示すように SSC をそれぞれの TGC からの  $\phi$  方向の入力ごとに独立した 2 つの SSC に分割した。

- SSC2→SSC2a + SSC2b
- SSC4→SSC4a + SSC4b
- SSC6→SSC6a + SSC6b

- SSC12→SSC12a + SSC12b

こうすることで、1つの subsector で 2通りの運動量測定を行う必要が無くなる。また、これにより後段の Track Selection Logic に入力される track 候補は 19 から 23 に増加した。

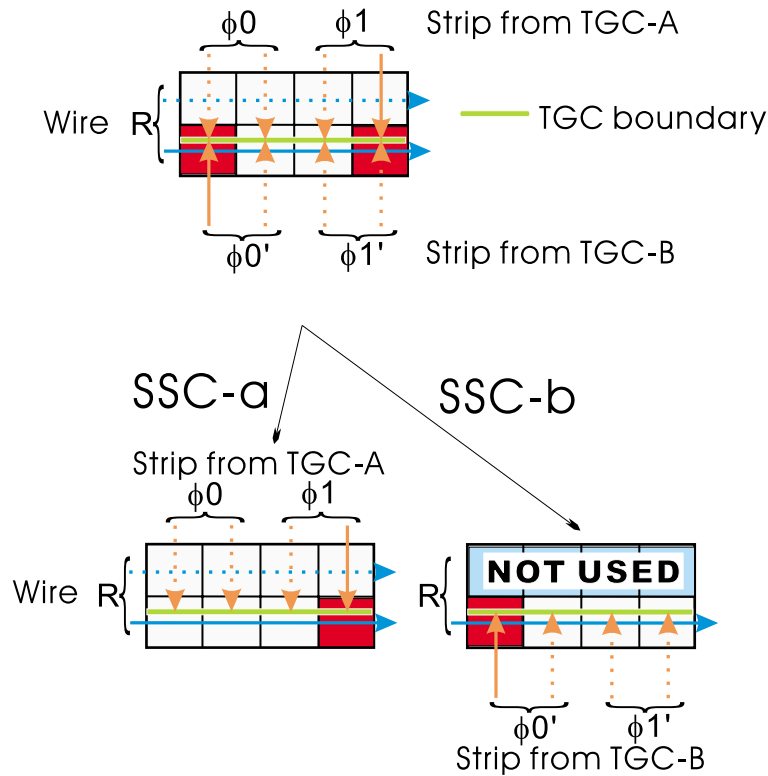


図 3.6: TGC 境界を含む SSC の分割

この図では、R方向の  $Hi-P_T$  からの入力下段の subsector、TGC-A からの  $\phi 1$  は右端列の subsector、TGC-B からの  $\phi 0'$  は左端列の subsector に入力されている。図で、赤く示した subsector は、これらの入力から考えられるミューオンの track 位置である。

これを分割した SSC-a 及び SSC-b では R 方向の入力は共通であるが、 $\phi$  方向の入力をそれぞれ TGC-A からの  $\phi$  入力を SSC-a、TGC-B からの  $\phi$  入力を SSC-b に振り分けた。

SSC-a では、全ての subsector から track 候補を作る事が出来るのに対し、SSC-b では R 方向の入力が上段の subsector だった場合、その位置に strip は存在しないため、track 候補を出力してはならない。従って、SSC-b では、上段の subsector を全ての入力パターンに対し track 候補を出力しないように設定される。

SSC-a と SSC-b とで、2個出て来た track 候補は、そのまま Track Selection Logic に送られ、そこで他の SSC と全く同等に選別が行われる。これらの track 候補の  $P_T$  値が等しい場合、Track Selection Logic で、SSC-a と SSC-b のどちらを優先させるかは、まだ確定していない。<sup>3</sup>

### 3.1.4 Sector Logic の機能ブロック

図 3.7 は、Sector Logic での Trigger 処理について、具体的な各機能ブロックにブレイクダウンして描いた図である。

<sup>3</sup>SSC 内での TGC 境界の位置が問題になってくる。最終的には、Detector/Trigger simulation を行って決定することになっている。

データ収集システムへの readout ブロックや、Clock と Trigger 信号の分配を行っている TTC(Trigger, Time and Control) ブロックなどの周辺回路については、Sector Logic 固有の機能ではないため、ここでは触れない。これらのブロックについては、Prototype-0 の設計に関連して 3.4.3.3 で概要を説明してある。

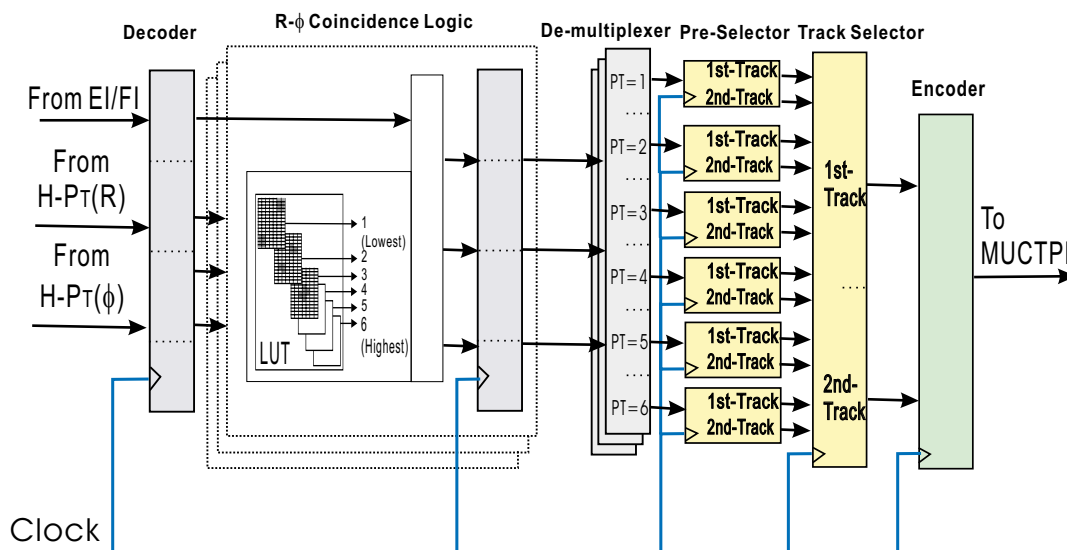


図 3.7: Sector Logic の機能ブロック別の構成

さて、Hi- $P_T$  モジュールから送られて来た入力、それぞれの機能ブロックで順に処理を受ける。De-multiplexer を除いては機能ブロックの出力部分には flip flop(FF) が配置され、全体としてパイプライン構造となっている。

### 3.1.4.1 Decoder

ここでは、Hi- $P_T$  からの hit 入力を各 SSC 単位に組み換えを行なっている。2.3.7.2 節で述べたように、Hi- $P_T$  モジュールからは、Hi- $P_T$ ASIC 1 個につき 1 本の optical cable で出力される。この optical cable は、1 本当たり 20bit のデータ幅を持ち、最大 2 hit を送ることが出来る。R 方向に付いては、これらの関係が 図 3.4 で表されている。

本論文では、図 2.10 において Hi- $P_T$  モジュール (EW0, ES0, F0) に載っている Hi- $P_T$ ASIC に対し、 $\eta$  の小さい方から順に chip-0, chip-1, chip-2, chip-3 と番号を付けて扱う。それ以外の各モジュールの名前に付いては、標準スキーム [4] に従っている。

以下は、各 Hi- $P_T$  モジュールからの入力をどのようにして、SSC に振り分けるかを入力別にまとめたものである。

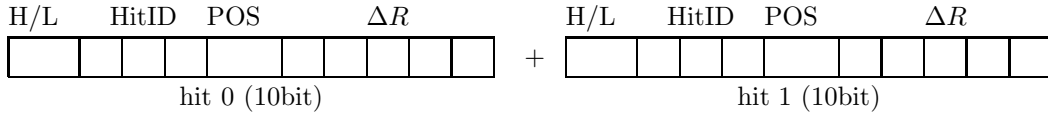
#### 1. EndCap 領域

EndCap 領域をカバーする Hi- $P_T$  モジュールは、R 方向 (Wire) が EW0、 $\phi$  方向 (Strip) が ES0 である。 $\phi$  (Strip) 方向は、2 個の Hi- $P_T$ ASIC で EndCap 部の Trigger Sector をカバー出来る。従って、4 個の Hi- $P_T$ ASIC が載っている ES0 は、2 つの Trigger Sector で共用している。

##### (a) R 方向 (Wire) … EW0

Hi- $P_T$  モジュール EW0 は、4 個の Hi- $P_T$ ASIC を持っている。従って Sector Logic には、4 optical cable の入力がある。

各 optical cable(20bit) への Output Format は、



となる。SSC 1 個のみ (SSC0) をカバーしているため、高々1 個しか hit が発生しない EW0-Chip0 については、hit 1 の部分をゼロでパディングする。(図 3.4 参照)

また、ここで HitID は、 $\eta$  方向に Sub Sector 2 個ずつのペアに対して順に付けた番号であり、POS はこの 2 個のどちらかを定めるビット (若い側が 0) である。

Sector Logic では、これらの情報から、 $R$ - $\phi$  Coincidence を取る Sub Sector Cluster 番号 (SSC#: [0-18]) を生成する。

よって、この領域 (EndCap region) では

$$\begin{aligned}
 SSC\# &= HitID && (EW0 - Chip0) \\
 SSC\# &= HitID + 1 && (EW0 - Chip1) \\
 SSC\# &= HitID + 7 && (EW0 - Chip2) \\
 SSC\# &= HitID + 13 && (EW0 - Chip3)
 \end{aligned}$$

である。

以下の図は、Hi-Pt board と、そのカバーする範囲を示してある。

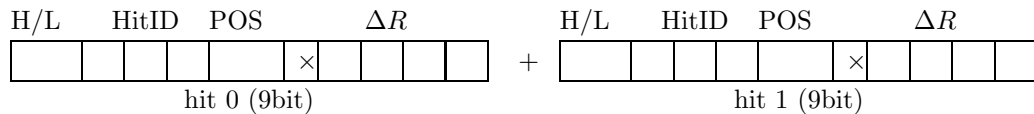
Hi-Pt	$\eta$	Slave Board	SSC	HitID	Max Hit
EW0-Chip0	- 1.05	EWD0 EWT0	0	0	1 → 1
EW0-Chip1	1.05 - 1.37	EWD1 EWT1	1 2(a,b)	0 1	6 → 2
		EWD2	3 4(a,b)	2 3	
		EWD3 EWT2	5 6(a,b)	4 5	
EW0-Chip2	1.37 - 1.62	EWD4 EWT3	7 8	0 1	6 → 2
		EWD5	9 10	2 3	
		EWD6 EWT4	11 12(a,b)	4 5	
EW0-Chip3	1.62 - 1.92	EWD7 EWT5	13 14	0 1	6 → 2
		EWD8	15 16	2 3	
		EWD9 EWT6	17 18	4 5	

なお、SSC(a,b) と表されているのは、3.5 節で説明した、分割された SSC である。

(b)  $\phi$  方向 (Strip) ... ES0

前述したように Hi-P<sub>T</sub> モジュール ES0 に搭載された 4 個の Hi-P<sub>T</sub>ASIC のうち、2 個でこの領域がカバーされる。従って Sector Logic には、2 optical cable の入力がある。

各 optical cable(20bit) への Output Format は、



ここで、× はパディング (0)。

HitID については、次のようになる。

ES0-Chip0



HitID	Pivot Chamber	$\phi$
0	M3-E5 (T9)	0,1
1	M3-E5 (T9)	2,3
2	M3-E4 (T8)	0,1
3	M3-E4 (T8)	2,3
ES0-Chip1		
HitID	Pivot Chamber	$\phi$
0	M3-E3 (T7)	0,1
1	M3-E3 (T7)	2,3
2	M3-E2 (T6)	0,1
3	M3-E2 (T6)	2,3
4	M3-E1 (T5)	0,1
5	M3-E1 (T5)	2,3

$\phi$  が 0 or 1、あるいは 2 or 3 を決めるのが POS であり、 $\phi=0/2$  のとき POS=0、 $\phi=1/3$  のとき POS=1 である。

よって、まとめると

Hi-Pt	Slave Board	Pivot Chamber	HitID	Sub Sector	SSC	Max Hit
ES0-Chip0	ESD0 EST0	M3-E5	0 1	00-05	00-02	4 → 2
	ESD1	M3-E4	2 3	05-08	02-04	
ES0-Chip1	ESD2	M3-E3	0 1	08-12	04-06	6 → 2
	ESD3 EST1	M3-E2	2 3	12-25	06-12	
	ESD4	M3-E1	4 5	25-37	12-18	

となる。

Strip の場合、HitID に対応する SSC は複数となる。例えば、Hi-Pt の ES0-Chip0 から HitID=0 の hit が来れば、SSC=00 から SSC=02 までの 3 つの SSC に同時に入力される。ここで、SSC=02, 08, 04, 06, 12 が重複しているが、これは SSC が Chamber 境界を跨いでいる (Sub Sector も同じ) ために、両方の Chamber からの信号を受けることを意味している。

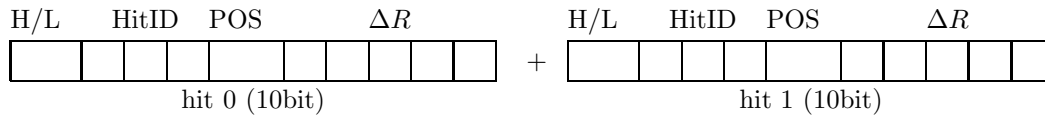
## 2. Forward 領域

Forward 領域をカバーする Hi-P<sub>T</sub> モジュールは、R 方向 (Wire)、 $\phi$  方向 (Strip) とともに F0 である。F0 には 3 個の Hi-P<sub>T</sub>ASIC があるが、2 個が R 方向 (Wire) 用、1 個が  $\phi$  (Strip) 用である。

(a) R 方向 (Wire)・・・F0

この領域は F0 の 2 個の Hi-P<sub>T</sub>ASIC でカバーされるため、Sector Logic には、2 optical cable の入力がある。

各 optical cable(20bit) への Output Format は、



ここで、HitID は、 $\eta$  方向に Sub Sector 2 個ずつのペアに対して順に付けた番号であり、POS はこの 2 個のどちらかを決めるビット (若い側が 0) である。

Sector Logic では、これらの情報から、R- $\phi$  Coincidence を取る Sub Sector Cluster 番号 (SSC#: [0-7]) を生成する。よって、この領域 (Forward region) では

$$SSC\# = HitID \quad (F0 - Chip0)$$

$$SSC\# = HitID + 6 \quad (F0 - \text{Chip1})$$

である。

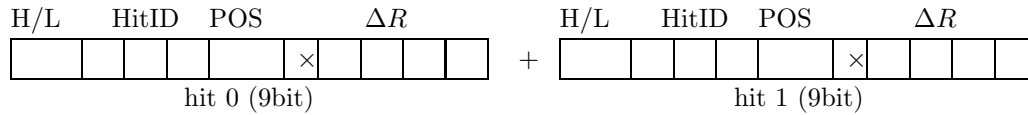
以下の図は、Hi-Pt board と、そのカバーする範囲を示してある。

Hi-Pt	$\eta$	Slave Board	SSC	HitID	Max Hit
F0-Chip0	1.92 – 2.29	FWD0 FWT0	0 1	0 1	6 → 2
		FWD1	2 3	2 3	
		FWD2 FWT1	4 5	4 5	
F0-Chip1	2.29 –	FWD3 FWT2,3	6 7	0 1	2 → 2

(b)  $\phi$  方向 (Strip) … F0

この領域は F0 の 1 個の Hi-PT ASIC でカバーされるため、Sector Logic には、1 optical cable の入力がある。

各 optical cable(20bit) への Output Format は、



ここで、× はパディング (0)。

HitID については、次のようになる。

F0-Chip3		
HitID	Pivot Chamber	$\phi$
0	M3-F (T2)	0,1
1	M3-F (T2)	2,3

$\phi$  が 0 or 1、あるいは 2 or 3 を決めるのが POS であり、 $\phi=0/2$  のとき POS=0、 $\phi=1/3$  のとき POS=1 である。

よって、まとめると

Hi-Pt	Slave Board	Pivot Chamber	HitID	Sub Sector	SSC	Max Hit
F0-Chip3	FSD0 FST0	M3-F	0 1	00-15	00-07	2 → 2

となる。

Strip の場合、HitID に対応する SSC は複数となる。特に、Forward region は 1 chamber で構成されているので、全ての SSC に入力される。HitID は、POS と併せて  $\phi$  を決めることにのみ使用される。

### 3.1.4.2 R- $\phi$ Coincidence Logic

R- $\phi$  Coincidence Logic の役割は、R 方向 (Wire) と  $\phi$  方向 (Strip) の両方の入力を用いて、LUT により入射ミューオンの横運動量を測定し、6 段階の横運動量の閾値に分類して出力することである。直交する 2 つの信号を用いることにより、それぞれの入力と同じミューオンの横運動量の値となるような hit で無い場合 – ノイズや fake hit など – については、track 候補から排除することが出来る。

図 3.8 は、SSC への入力信号を示している。Hi-PT から送られて来た HitID は Trigger Sector 内での SSC の位置情報なので、SSC への入力にならない。よって、R 方向 (Wire) が 7bit × 1、 $\phi$  方向 (Strip) が 6bit × 2 で、LUT への入力は、合計 19bit となる。これは、2M のアドレス空間に相当し、出力は 3bit で表現した 6 段階別の横運動量値 ( $P_T$ ) である。

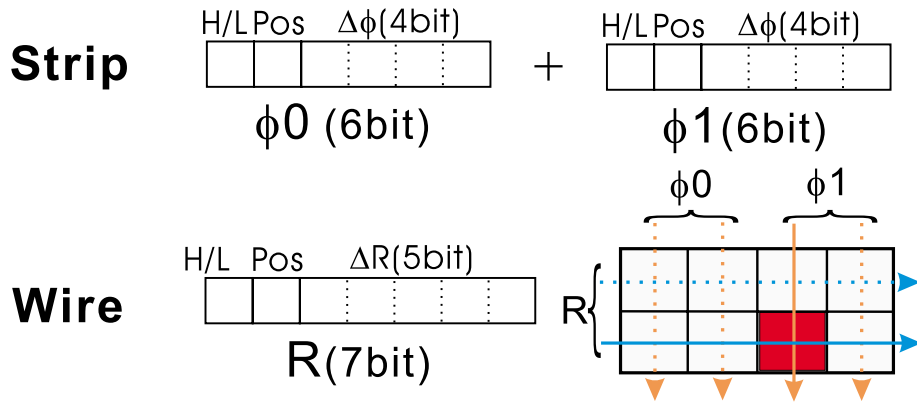


図 3.8: SSC への入力信号

このようにして SSC 毎に  $P_T$  が求まった後、 $R$ - $\phi$  Coincidence Logic では、この結果を EI/FI チェンバ (小型の TGC) での hit 情報を用いることによって、Interaction Point 以外の方向から入射した fake hit を排除する。EI/FI については LUT ではなく、該当する領域で hit が無かったときには入力をマスクするような、シンプルな組み合わせ回路で実装する予定である。

EI/FI チェンバは、図 3.9 のように、穴の空いている空間が有る上に TGC に対し  $\phi$  方向に傾いて設置されているため、実装に当たって考慮すべき条件は多く、実装方法はまだ確定していない。

EI/FI によるマスクが終わると、 $R$ - $\phi$  Coincidence Logic は、3bit の  $P_T$  の値と共に SSC 内における ミューオン track の位置 ( $R$ :1bit,  $\phi$ :2bit) を次の De-multiplexer に送る。

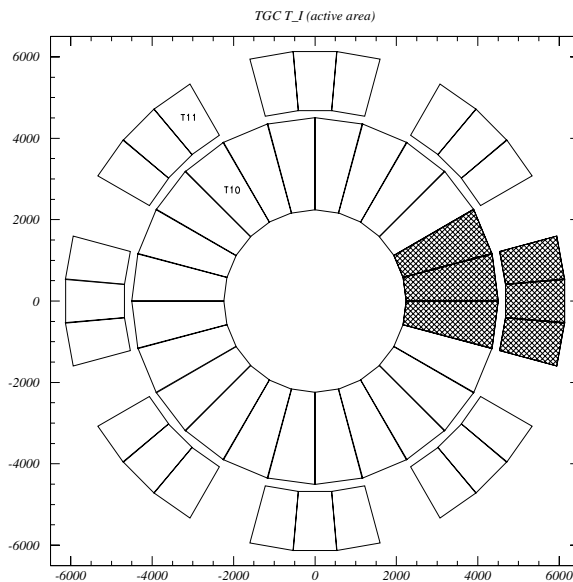


図 3.9: EI(EndCap Inner), FI(Forward Inner) TGC の  $R$ - $\phi$  平面図

### 3.1.4.3 De-multiplexer

De-multiplexer は、 $R$ - $\phi$  Coincidence Logic で SSC 毎に測定されたミューオン track(EndCap 領域は 23 個, Forward 領域は 8 個) を、各  $P_T$  の値に対応する Track Selection Logic の 6 個の Track Pre-Selector

に分配する。論理的には De-multiplexer は、入力する SSC の数だけ用意される。

De-multiplexer は、出力に FF を持たない組み合わせ回路であり、その伝搬遅延 ( $t_{PD}$ ) は、Track Pre-Selector に加算される。

図 3.10 に 1 つの SSC あたりの Demultiplexer の入力と出力の関係を示す。各  $P_T$  の値別の Track Pre-Selector に送るのは、SSC 内におけるミュオン track の位置 (R:1bit,  $\phi$ :2bit) と、その SSC でその  $P_T$  の値のミュオン track が見付かった事を示す hit 信号 (1bit) である。

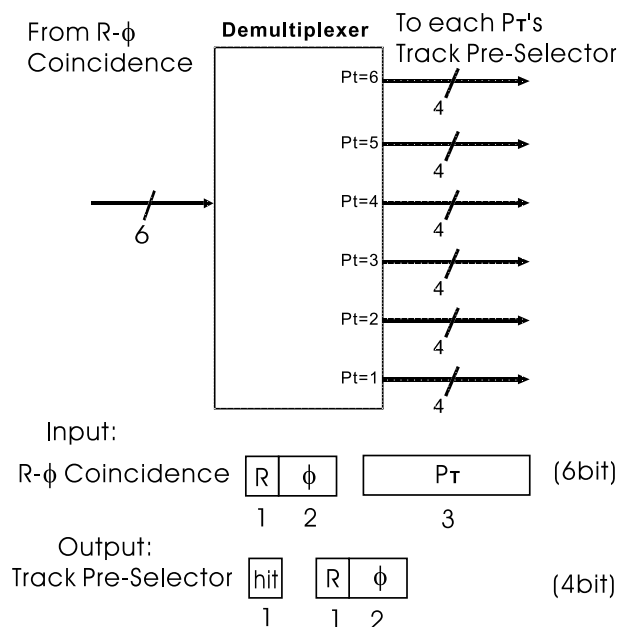


図 3.10: De-multiplexer の入力信号と出力信号

### 3.1.4.4 Track Selection Logic(Track Pre-Selector, Track Selector)

Track Selection Logic は R- $\phi$  Coincidence Logic で生成された複数の hit から、運動量の値の大きいものから順に 2 個の候補を選出する Priority logic である。パイプライン構造の中で動作させるため、処理の並列度を高くすることにより一段当りの遅延時間を抑えている。そのため、Track Selection Logic を Track Pre-Selector と Track Selector の 2 つのブロックに分けた。それぞれの段は出力に FF を持ち、ラッチされた出力が後段に送られる。

#### Track Pre-Selector

Track Pre-Selector は、各  $P_T$  の値別に 6 個あり、それぞれ De-multiplexer から 23 個 (EndCap 領域) あるいは、8 個 (Forward 領域) の入力を持つ。それぞれの Track Pre-Selector は同じ  $P_T$  の値の入力から Primary/Secondary Selector によって、最大 2 個の track を選出する。Primary/Secondary Selector とは入力端子に優先順位のついたセレクトラ (組み合わせ回路) であり、同時に複数の入力があった場合、優先順位の高い入力端子への入力から順に 2 つまでを選ぶセレクトラである。Primary/Secondary Selector の詳細については後述する。

入力 track は、 $\eta$  の小さい方 (EndCap 面の外周側、SSC0 が最も優先) が優先して priority logic によって選出され、その SSC の位置情報が付加されて hit 信号とともに出力される。6 個の Track Pre-Selector からの合計 12 個の track は Track Selector に送られる。

図 3.11 は、1 個の Track Pre-Selector の構造と入出力信号を描いた図である。入力は、hit 信号と 3bit の位置情報 (R:1bit,  $\phi$ :2bit) に分けて扱われる。それぞれの SSC から出力された位置情報は、1st 候補用

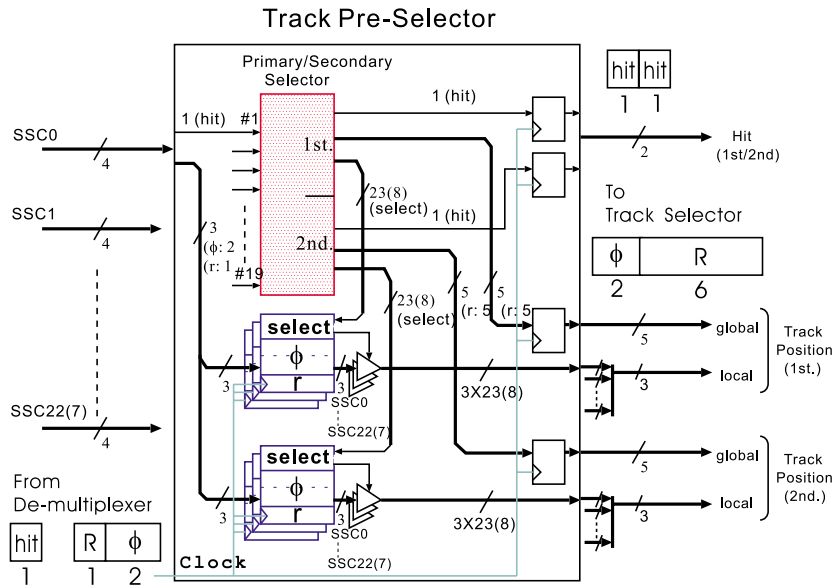


図 3.11: Track Pre-Selector の構造と入出力信号

及び 2nd 候補用の 2 つのレジスタに分けて、clock の立上りのタイミングで格納される。これと並行して hit 信号から、Primary/Secondary Selector で 1st 候補及び 2nd 候補が選出され、選ばれたラインからは Select 信号が出力される。この Select 信号は、(1st 候補、2nd 候補それぞれ) その SSC のデータと同じレジスタに格納される。また、Primary/Secondary Selector では、1st 候補や 2nd 候補を選出すると、同時に hit 信号を出力し、出力 FF を経て Track Selector に送られる。clock の立上りのタイミングで、各レジスタからはデータが出力される。レジスタの位置情報 (3bit) の出力は 3-state バッファに接続されており、select 信号が active のデータだけが、3-state バッファの先にあるバスに出力され、SSC の座標情報 (図 3.11 では global と表されている、R 座標の 5bit) とともに Track Selector に送られる。

まとめると、Track Pre-Selector から出力されるデータは、その Track Pre-Selector で 1st/2nd 候補が選出された事を示す hit 信号 (2bit) と 1st/2nd 候補それぞれについて、SSC の座標情報 (R: 5bit) を含めた、ミューオン Track の位置情報 (R:6bit,  $\phi$ :2bit) となる。

### Track Selector

Track Selector では、6 個の Track Pre-Selector からの 12 個の入力から最終的な 2 個の track を選出する。Track Selector は、Sector Logic 中に 1 個のみ存在する。図 3.12 は Track Pre-Selector の構造と入出力信号を描いた図である。

基本的な構造は Track Pre-Selector と同じであるため、異なっている部分のみを説明する。まず、Track Selector の Primary/Secondary Selector の優先順位は、 $P_T=6$ (最大) の Track Pre-Selector で 1st 候補、同 2nd 候補、 $\dots$ 、 $P_T=1$ (最小) の Track Pre-Selector で 1st 候補、同 2nd 候補 の順である。

また、track 位置情報を格納するためのレジスタが 4 個有るが、6 個の Pre-Selector で選ばれた 1st/2nd 候補それぞれについて、Track Selector の 1nd/2nd 候補として別々に選択することを可能にするため、二重化したことによる。

最終的に、Track Selector で 1st 候補と 2nd 候補が選ばれ、それぞれ  $P_T$  値 (3bit) と位置情報 (R:6bit,  $\phi$ :2bit) が Encoder に送られる。

### Primary/Secondary Selector について

Primary/Secondary Selector とは入力端子に優先順位のついたセクタ (組み合わせ回路) であり、同時に複数の入力があった場合、優先順位の高い入力端子への入力から順に 2 つまでを選ぶセクタである。

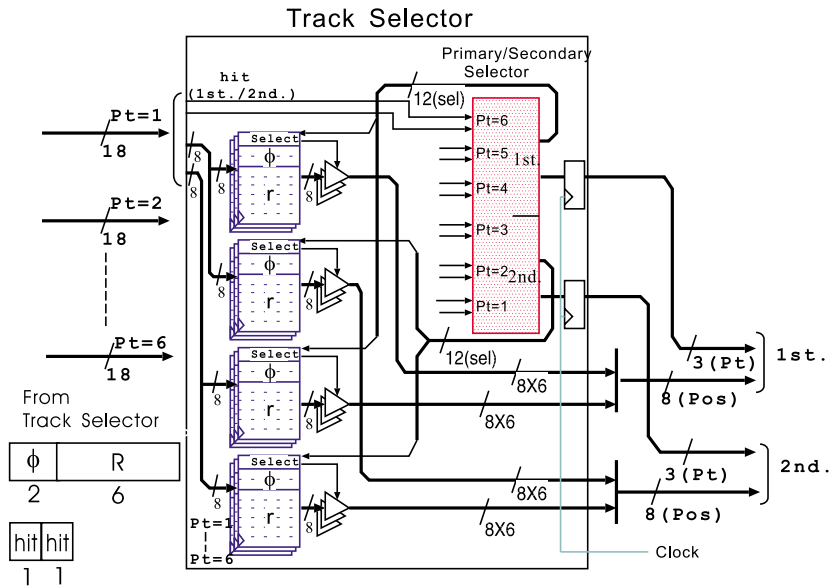


図 3.12: Track Selector の構造と入出力信号

このような論理回路は、大きく分けて 2 通りの設計方法がある。

- priority logic を使用する。
- 全入力に優先順位の値を付加し、全ての組み合わせについて入力線の優先順位の値をマグニチュード・コンパレータで比較する。 $({}_N C_2$  通りの比較を行う)  $N-1$  本の入力 (残りの入力全て) に対し勝っている入力を優先順位 1 とし、順に  $N-2$  本の入力に勝っている入力を優先順位 2、... とする。

今回は、シンプルでオーバーヘッドが少ない、priority logic を用いる実装を採用した。組み合わせ回路の基本回路である priority logic は、 $N$  本の入力線に順に優先順位を付けておき、複数の入力線から同時に入力が有った場合、最も優先順位の高い入力線を有効な入力として選択する回路である。特に、有効になった入力線の番号の値を (二進数などに) エンコードして出力する回路を Priority Encoder と言う。

今回は出力をエンコードせず、有効になった入力線のみをそのまま出力する  $N$  入力  $\times$   $N$  出力の priority selector を用い、図 3.13 のように設計した。

優先順位第 1 位の出力は、Priority Selector の出力をそのまま使えば良い。問題の優先順位第 2 位は、優先順位第 1 位の Priority Selector (上側) の出力の有るチャンネルに対し、入力をマスクした Priority Selector (下側) を用いることにより得られる。

この方法は、設計がシンプルでオーバーヘッドが小さいものの、優先順位 2 位の出力を得るまでに 2 個の Priority Selector の伝搬遅延を受ける欠点がある。しかしながら、次節で述べるプロトタイプによるテストでは、遅延時間は対応可能な範囲であることが確認されている。

### 3.1.4.5 Encoder

Encoder では Track Selector からの入力を受けて、CTP (Central Trigger Processor) のフロントエンドである、MUCTPI (Muon CTP Interface) に信号を送り出す出力回路である。MUCTPI へは、Track Selector から送られて来た、1st/2nd のミュオン track 候補の  $P_T$  値 (3bit) と位置情報 (R:6bit,  $\phi$ :2bit) の値を、BCID カウンタの値やその他のフラグとともに 32bit の LVDS 信号で送られる。

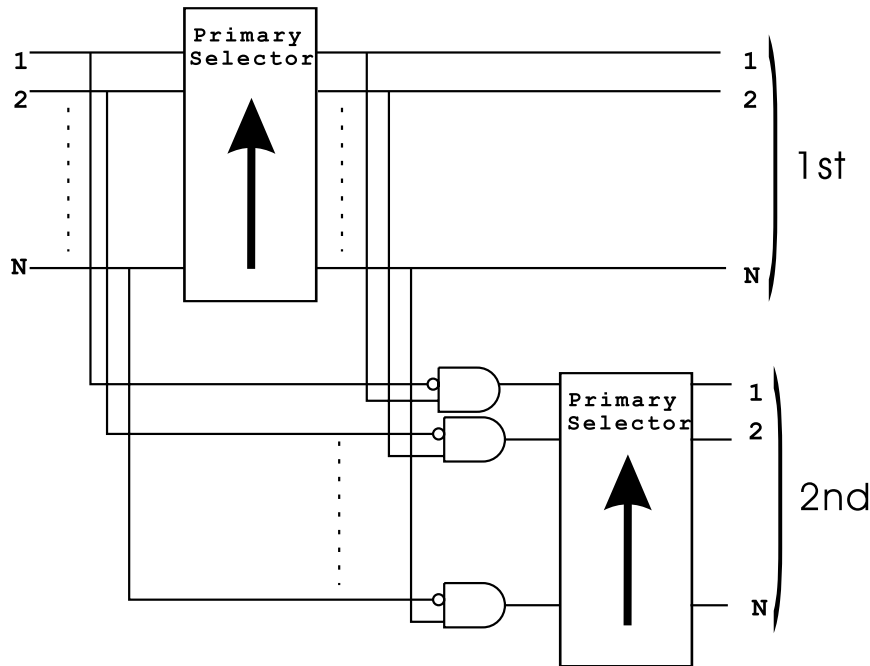


図 3.13: Primary/Secondary Selector の実装

## 3.2 Track Selection Logic のプロトタイプとテスト

Track Selection Logic の仕様は、3.1.4.4 で述べたように決定された。この基本設計に基づいた各機能ブロックの記述には、HDL(Hardware Description Language)である Verilog-HDL を用いた。Verilog-HDL で記述された Track Selection Logic は、HDL シミュレータである Verilog-XL(Cadence 社製)を用いて論理シミュレーションを行い、設計意図が正しく実装されているかを検証した。

そして、実際に FPGA に書き込んで動作の検証を行った。このプロトタイピングには、pt3 モジュールを用いた。pt3 モジュールは、ATLAS 実験の読みだしシステムの検証のために製作されたプロトタイプモジュールであるが、汎用性を重視して設計されているために、Track Selection Logic のプロトタイピングに用いることが出来る。(西田 [6] を参照)

このプロトタイプの目的は、基本設計に基づいて設計した機能を、実際にプロトタイプ上で動作させてみることにより、基本設計に問題は無いが、必要な性能を満たす事が出来るかを評価し、基本設計にフィードバックする事が目的である。

### 3.2.1 Track Selection Logic の Verilog-HDL による記述と論理シミュレーション

今回の検証のため Verilog-HDL で記述した回路は、図 3.7 における Track Pre-Selector から Track Selector までの区間の回路である。HDL シミュレータでの論理シミュレーションは回路の伝搬遅延を受けることなく行われるため、どれだけ大きな回路であってもシミュレータ上では動作させることが可能である。<sup>4</sup> しかしながら、今回は次のような理由により SSC 数を 4 個に減らして論理シミュレーションを行った。

- シミュレーションする回路規模が大きくなればなるほど、シミュレーションを実行するのに長時間必要になり、計算機資源を大きく消費する。

<sup>4</sup>HDL シミュレータでは、論理合成後に遅延情報を用いたゲートレベルのシミュレーションを行うことも可能である。この場合、本物の回路と全く同じ挙動をする。

### Data format for barrel, end-cap and forward

Bit #	Meaning	Barrel	End-cap	Forwar
0	>2 candidates in a sector			
1	Sub-sector1 with overlap flags	ROI1<0>	ROI1<0>	ROI1<0>
2		ROI1<1>	ROI1<1>	ROI1<1>
3		ROI1<2>	ROI1<2>	ROI1<2>
4		ROI1<3>	ROI1<3>	ROI1<3>
5		ROI1<4>	ROI1<4>	ROI1<4>
6		res	ROI1<5>	ROI1<5>
7		res	ROI1<6>	res
8		OVL1<0> [note 1]	ROI1<7>	res
9		OVL1<1> [note 1]	OVL1 [note 2]	res
10	Sub-sector2 with overlap flags	ROI2<0>	ROI2<0>	ROI2<0>
11		ROI2<1>	ROI2<1>	ROI2<1>
12		ROI2<2>	ROI2<2>	ROI2<2>
13		ROI2<3>	ROI2<3>	ROI2<3>
14		ROI2<4>	ROI2<4>	ROI2<4>
15		res	ROI2<5>	ROI2<5>
16		res	ROI2<6>	res
17		OVL2<0> [note 1]	ROI2<7>	res
18		OVL2<1> [note 1]	OVL2 [note 2]	res
19	Pt1<0>	[note 3]	[note 3]	
20	Pt1<1>	[note 3]	[note 3]	
21	Pt1<2>	[note 3]	[note 3]	
22	Pt2<0>	[note 3]	[note 3]	
23	Pt2<1>	[note 3]	[note 3]	
24	Pt2<2>	[note 3]	[note 3]	
25	>1 candidate in ROI1		res	res
26	>1 candidate in ROI2		res	res
27	BCID<0>	[note 4]		
28	BCID<1>	[note 4]		
29	BCID<2>	[note 4]		
30		res	res	res
31		res	res	res

**Note 1.** OVLi<1..0>

- 00 No overlap
- 01 Overlap in phi
- 10 Overlap in eta
- 11 Overlap in phi and eta

**Note 2.** OVLi

- 0 No overlap
- 1 Overlap barrel - endcap

**Note 3.** Pti<2..0>

- 001 Low Pt 1
- 010 Low Pt 2
- 011 Low Pt 3
- 100 High Pt 1
- 101 High Pt 2
- 110 High Pt 3
- 111 No candidate
- 000 Reserved

**Note 4.** Three bits for the BCID should be enough for checking purposes as it is very unlikely to have a misalignment greater than 8 BC.

They are the low order bits of the BCID provided by the TTC system (TTCrx receiver chips or equivalent).

図 3.14: MUCTPI へ送るデータのフォーマット



- 入力する SSC の数を増やすと、パラメタライズ出来ない配線の記述量が増え、本質的でない部分でバグの混入の可能性が増える。
- テストベクタ作成の手数が増え、同様にバグ混入の可能性が増える。

なお  $P_T$  の値は本番用と同じく 6 通りある。

### 3.2.1.1 Primary/Secondary Selector の実装

Track Selection Logic の中心となる Primary/Secondary Selector の Verilog-HDL による実装<sup>5</sup>を以下に示す。

```

module precore( hitin, presel_1, presel_2, prepos_1, prepos_2, prehit );
    parameter      N_SSC = 4;
    input   [N_SSC:1]    hitin;
    output  [N_SSC:1]    presel_1;      // Pre selector(1st) select
    output  [N_SSC:1]    presel_2;      // Pre selector(2nd) select
    output  [4:0]        prepos_1;      // Pre selector(1st) position output
    output  [4:0]        prepos_2;      // Pre selector(2nd) position output
    output  [1:0]        prehit;        // Pre selector      hit      output

    assign {prehit[1], prepos_1, presel_1} = primary( hitin );
    assign {prehit[0], prepos_2, presel_2} = primary( ~presel_1 & hitin );

    function [N_SSC+6:1] primary;
    input   [N_SSC:1]    hitin;
    begin : loop
        integer i;
        primary = 0;
        for (i = N_SSC; i >= 1; i = i - 1)
            if ( hitin[i] ) begin
                primary[i] = hitin[i];          // set primary bit as sel
                primary[N_SSC+5:N_SSC+1] = i;  // set prepos (position bits)
                primary[N_SSC+6] = hitin[i];    // set prehit bit
            end
        end
    end
endfunction

endmodule

```

HDL 設計では、論理回路を具体的な論理ゲートに書き下す必要は無く、このような回路の動作のみを記述した RTL(Register Transfer Level) の記述をすることで、論理シミュレーションや実際のデバイスへの論理合成を行う事が出来る。さらに、入力チャンネル数を変更したい場合、特に Priority Logic の入力チャンネル数が変化した場合、論理ゲートレベルでは回路は大きく変わり、最初から設計をやり直さなければ

<sup>5</sup>これは Track Pre-Selector 用であるが、Track Selector 用も入力数が異なるだけで全く同じ実装である。

ならなくなるが、このような RTL の記述では、パラメータ N\_SSC を定義しなおすだけで済む。このメリットは大きいと言える。

### 3.2.1.2 結果

Track Selection Logic の論理シミュレーションの結果を図 3.15 に示す。Track Selection Logic は、Track Pre-Selector と Track Selector の 2 つの機能ブロックを含み、それぞれが出力に FF を持っているため、入力した時点から 2 clock 遅れて出力されている事が分かる。

さて、Pt=6 から Pt=1 までの上の 6 行は Track Pre-Selector への入力であり、Pos\_1st と Pt\_1st は、Track Selector からの 1st 候補の SSC 位置とその P<sub>T</sub> 値であり、Pos\_2nd と Pt\_2nd は、Track Selector からの 2nd 候補の SSC 位置とその P<sub>T</sub> 値である。

ここでシミュレーションした回路は、P<sub>T</sub>=1 が最も優先度の高く<sup>6</sup>、同じ運動量の入力が複数ある場合、SSC 座標の大きいものほど優先するようにしてある。

図 3.15 において、(1) の時点では、P<sub>T</sub>=6 の Track Pre-Selector に 1010 という入力が入っている。これは、P<sub>T</sub>=6 の入力で SSC 位置が 4 のものと 2 のものが有ることを示している。

そして、(1) の入力があったから 2clock 後に Track Selector からの出力が (2) に現れている。

入力はともに P<sub>T</sub>=6 であるため、SSC 座標が大きい 4 のものが 1st 候補として選ばれ、(Pos\_1st=4, Pt\_1st=6) そして、次に SSC 座標が大きい 2 のものが 2nd 候補として選ばれている。(Pos\_2nd=2, Pt\_2nd=6)

このテストでは、このパターンの他に、

- 入力が 1 個の場合
- 4 個の異なる P<sub>T</sub> の値をもつ入力
- 2 個の異なる P<sub>T</sub> の値をもつ入力

のパターンについても検証し、その結果設計意図通り正しく動作させられることが確認された。

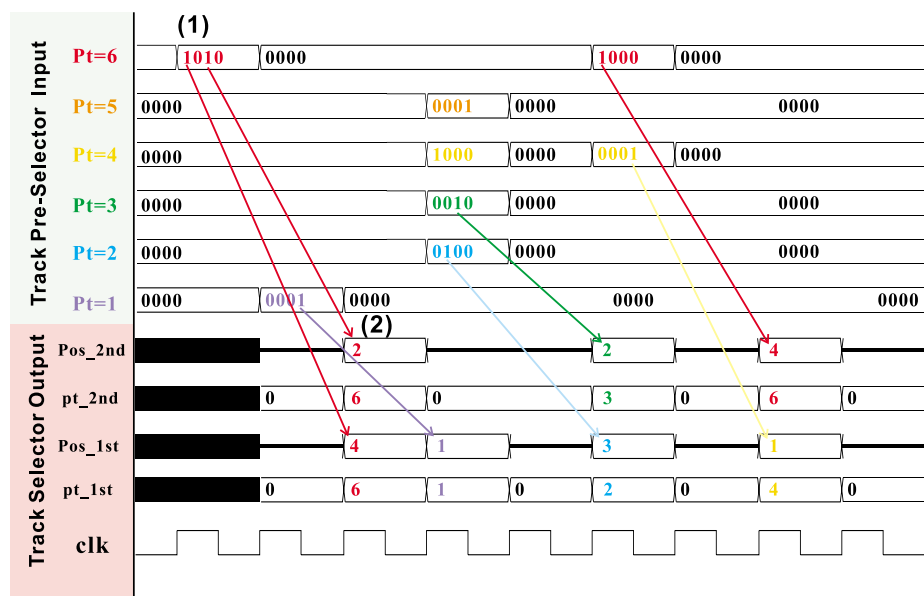


図 3.15: Track Selection Logic の論理シミュレーションの結果

<sup>6</sup>現在の設計では P<sub>T</sub>=6 を最も優先度の高い入力 (すなわち横運動量が最大) としている。

### 3.2.2 pt3 モジュールの仕様

pt3 モジュールは、Xilinx 社製の FPGA を 4 個搭載した 6U(Double Height) の VME ボードである。これらの FPGA は Spartan シリーズの XCS40-3PQ208C(QFP208 ピン (ユーザ I/O ピンは 169 ピン)) であり、ユーザーが自由に configuration(論理合成・配置配線した回路をデバイスに書き込む操作)して使用することが出来る。この FPGA は、内部に 784 個の CLB (Configurable Logic Block) と呼ばれる論理ブロックを持ち、40k システムゲート相当の規模のデバイスである。このデバイスは廉価版の製品として開発されており、動作速度や容量の点ではそれほど優れている訳ではない。したがって、実装した回路の動作速度は、実際に Sector Logic で使用するデバイスを用いた場合よりも遅くなる事を考慮しなければならない。また、今回の Track Selection Logic のような大規模な回路の場合、デバイスの容量と配線リソースの関係で機能を縮小した機能ブロックのみの実装しか出来ない。

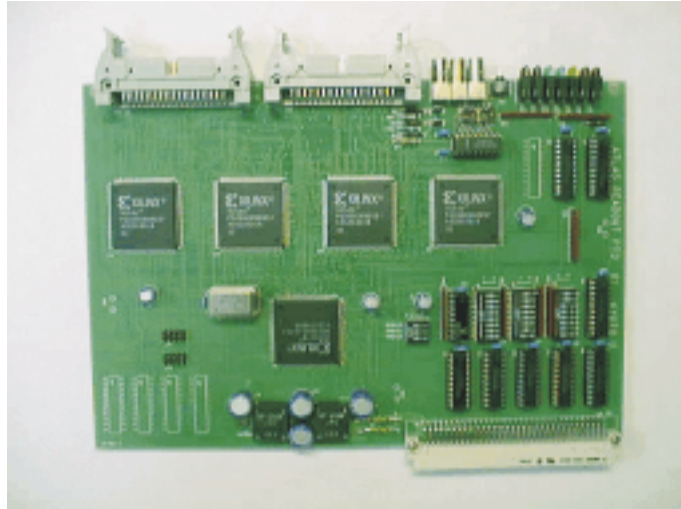


図 3.16: pt3 モジュールの写真

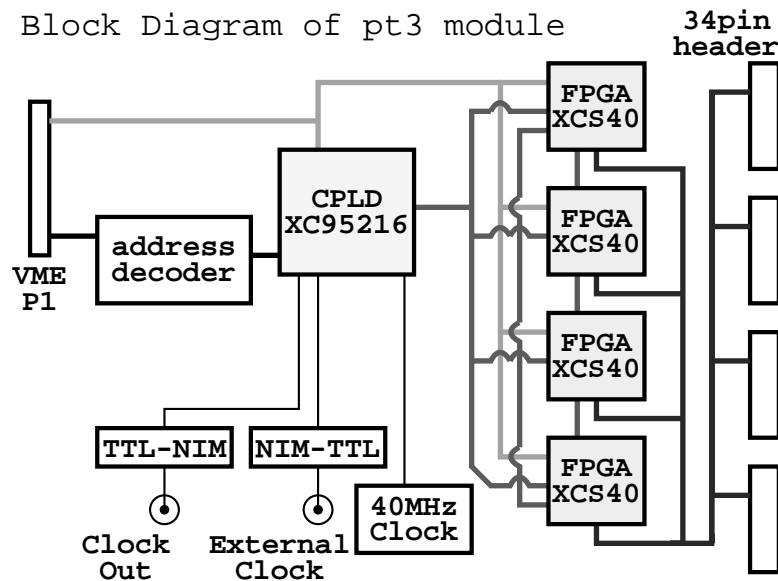


図 3.17: pt3 モジュールのブロック図

pt3 の写真を図 3.16 に、そのブロック図を図 3.17 に示す。4 個の FPGA は 5 本の 16bit 幅のバス (BUSA, BUSB, BUCS, BUSD, BUSE) で相互接続されており、そのうち 4 本 (BUSA, BUSB, BUCS, BUSD) はフロントパネルのピンヘッダをにも接続されており、これらの信号をロジックアナライザ等で観測出来るようになっている。さらに、隣接する FPGA 同士を接続する 16bit の配線が 4 組用意されている。(LineWX, LineXY, LineYZ, LineZW) また、VME 周りのコントロールは XC95216-10HQ208C という CPLD が受け持っており、VME バス側から各 FPGA にアクセスしたり FPGA を configuration することが出来る。

クロックは、pt3 モジュールに実装されている 40MHz のクロックの他、フロントパネルに LEMO コネクタから 2 系統のクロック入力があるが、CPLD でこれらを切替える事が出来る。

### 3.2.3 プロトタイプの仕様と設計

pt3 モジュールに Track Selection Logic を実装するに際して、最も制約となったのは FPGA を相互接続する配線リソースの数であった。Priority Logic という回路の性質上から、入力信号の個数を現実的な数に近づけなければ動作速度の検証にならない。そのため、Forward 領域と同じ 8 個の SSC をカバーさせた。

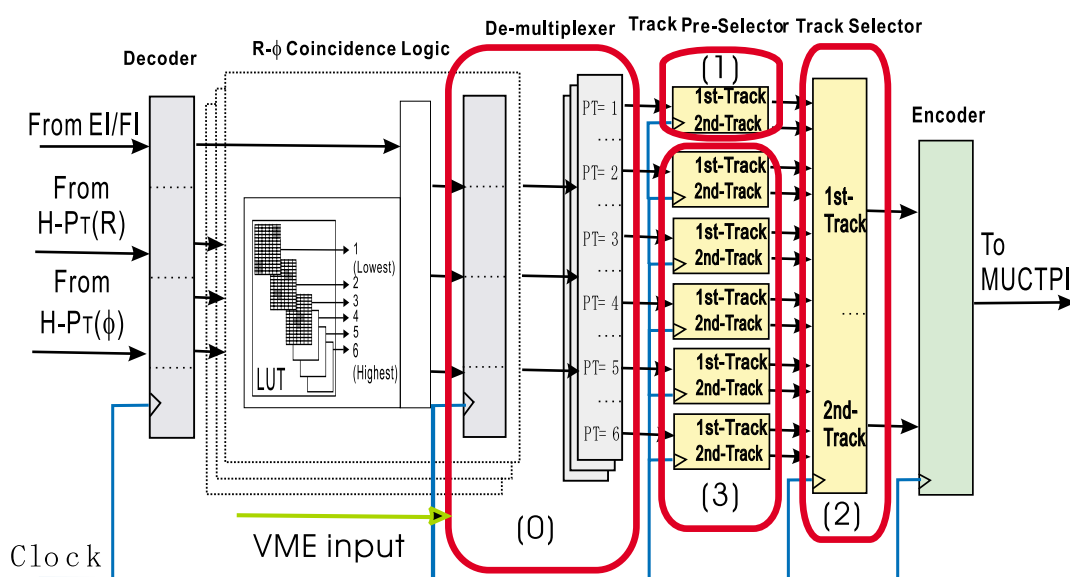


図 3.18: Track Selection Logic の 4 個の FPGA への振り分け

図 3.18 に、pt3 モジュールの 4 個の FPGA への各機能ブロックの割り振りを示す。

**FPGA0 パターンジェネレータ** pt3 モジュールにおいて、De-multiplexer の出力をエミュレートする、パターンジェネレータとなる。データや制御は VME バスからコントロールする。

**FPGA1 Track Pre-Selector** pt3 モジュールでは、Track Pre-Selector のうち、この  $P_T=1$  のものだけが、正しく機能を実装したものである。前述したように入力 SSC 数は 8。

**FPGA2 Dummy Latch** pt3 モジュールの配線リソースの関係で、 $P_T=2$  から  $P_T=6$  までの Track Pre-Selector については実装しない。代わりに、Track Pre-Selector の hit 出力のみを (0) のパターンジェネレータで生成し、それをそのまま FF でラッチして Track Selector に送出している。

**FPGA3 Track Selector** Track Selector は、規模の縮小等をしていないフルスペックである。12 入力。 $P_T=2$  から  $P_T=6$  までの入力は、FPGA2 Dummy Latch から hit 信号のみを受け付ける。これらの track の位置情報は、内部で設定した固定値を用いる。

これらの FPGA 相互のデータの配線を図 3.19 に示す。

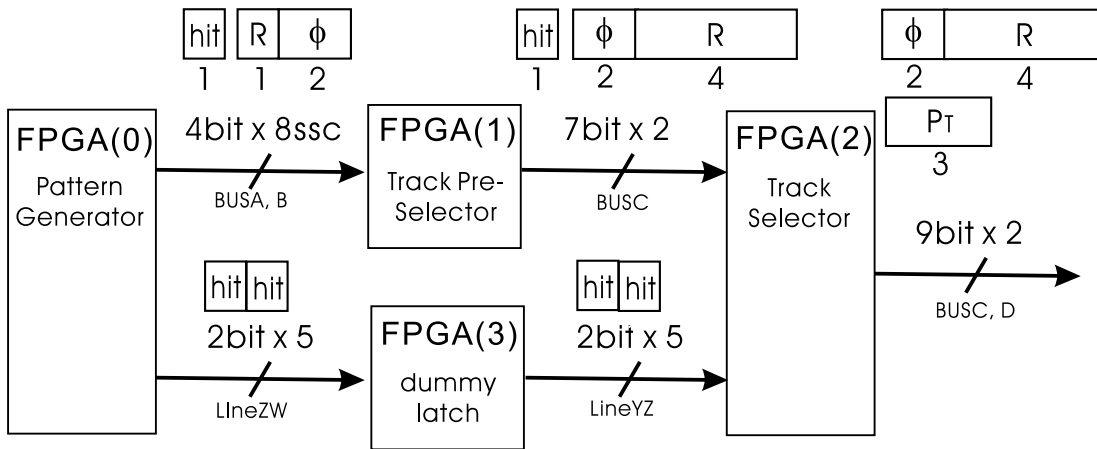


図 3.19: Track Slection Logic の 4 個の FPGA 間のデータの配線

### 3.2.4 プロトタイプの動作検証

3.2.3 節で設計した各 FPGA 毎の Verilog-HDL 記述を論理合成・配置配線を行い、それぞれの FPGA を configuration した。論理合成・配置配線は、FPGA ベンダである Xilinx 社が供給している Foundation シリーズを用いて行った。FPGA の配置配線を行うフィッター・プログラムは、FF 間の最大遅延パスを計算して最大動作周波数を推算する機能がある。4 個の FPGA のうち、FPGA3 の Track Selector が最も遅く、24.876MHz と計算された。これは、最悪の条件下 (電源電圧も規格内の最低値など) での最悪値であり、実際の回路は、これよりも速く動作するのが普通である。

全ての FPGA の configuration の終了後、pt3 の外部クロック入力をパルスジェネレータから与え、回路の挙動を観測するために、ロジック・アナライザをピン・ヘッドに接続した。最初に与えたクロックの周波数は、フィッターの推定値に基づき 20MHz とした。図 3.20 は、このときの波形である。設計意図通り、正常に動作している。

この図において HIT(demulti) と POS1,2,3(demulti) は、パターンジェネレータである FPGA0 から  $P_T=1$  の Track Pre-Selector である FPGA 1 に出力される hit 信号及び track の (SSC 内の) 位置であり、数字は SSC 番号である。(POS4 から POS7 はロジック・アナライザのチャンネル数の都合で観測していない)

さて、中央付近の FPGA0 と書かれている縦の破線は、FPGA0 のパターンジェネレータからデータが出力されるクロック (最下段) の立上りを指している。このクロックの立上りを受けて、FPGA0 からは  $P_T=1$  で、SSC 番号が 3、track の位置が 4 となる track を出力している。またこの波形には現れていないが、FPGA0 からは  $P_T=6$  の hit 信号も 1hit 分 FPGA3 に送っている。(  $P_T=1$  が 1track、  $P_T=6$  が 1track の合計 2track ある。)

図中に書き入れた矢印は、  $P_T=1$  の track が FPGA1(Track Pre-Selector) で 1st 候補となり、更に FPGA2(Track Selector) でも 1st 候補として出力されていく過程を示している。

さて、このようなテストパターンを何通りか用意して実験した結果、入力クロックが 20MHz では、pt3 の FPGA Track に実装した Selection Logic は設計通りに動作することが確かめられた。そこで、pt3 のクロック入力に接続したパルスジェネレータの出力周波数を少しずつ上げていったところ、入力クロック周波数が 34.0MHz を越えると、出力データの値が期待値と異なったり、出力値が確定するのに 1clock 余分にかかるなどの誤動作をするようになった。

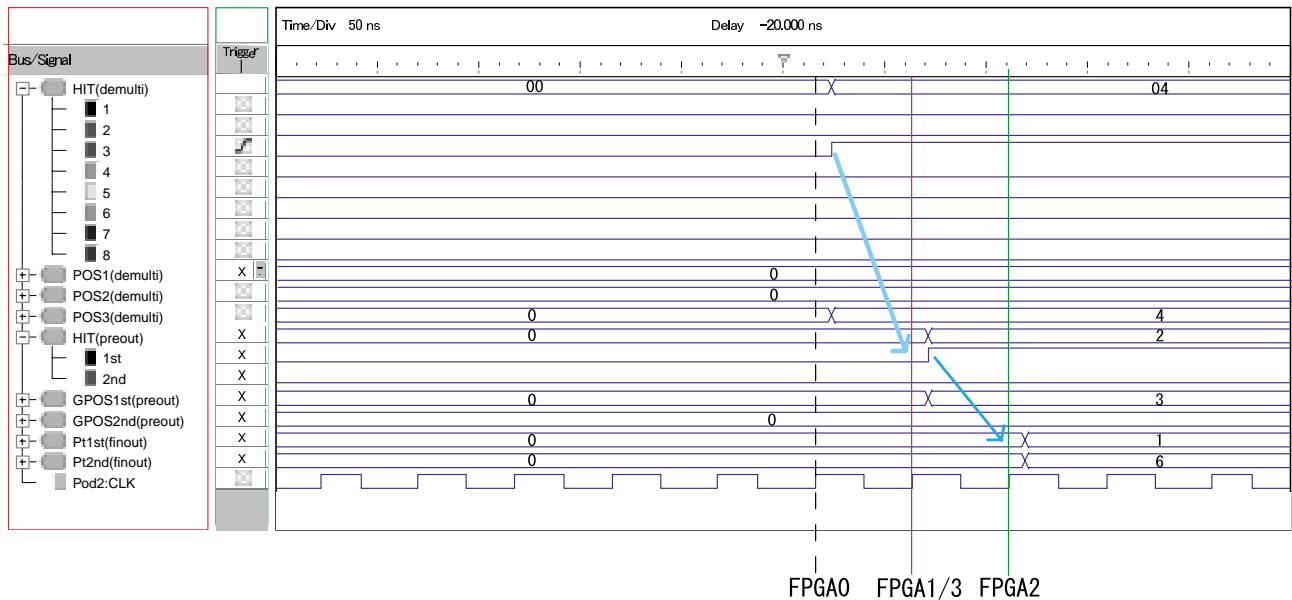


図 3.20: pt3 による Track Selection Logic の動作波形 (20MHz)

図 3.21 に、ATLAS 実験で用いられる 40MHz のクロックを入力した時の波形を示す。この場合、FPGA0 から 2clock 目は、Track Selector の 1st 候補の  $P_T$  値が、”3” と間違っ値が出力されている。3clock 目には、正しい値に戻っている。

このような誤動作が起こる原因は、入力クロックの周期が短くなる事により、最も伝搬遅延が長い信号が次段の FF のセットアップ (時間) 制約を破り、その結果 FF がメタステーブル状態になるためである。従って、より高い周波数で動作させるためには、回路全体の動作速度を律速しているパス (信号経路) の伝搬遅延時間を減らす事が必要となる。FPGA のようなプログラマブルロジックの場合、HDL での論理記述を工夫する以外にも論理合成や配置配線を行うときにタイミング制約条件を付加したり、エフォートレベルを上げて処理を行わせることにより、最大遅延時間のより少ない回路を合成することが出来る。

さて、pt3 に実装した Track Selection Logic の最高安定動作周波数が 34.0MHz と目標の 40MHz に及ばなかったものの、Spartan という廉価版のデバイスでそれなりの動作速度が達成出来たことは、一応の成功と考えられる。Spartan よりも高性能な FPGA (例えば、Virtex シリーズなど) を用いることで 40MHz 以上の動作速度を得ることは可能であろうと思われる。<sup>7</sup>

さらに、FPGA 業界では、激しい開発競争が繰り広げられており、その結果年々高性能なデバイスがリリースされている。したがって、本番用の回路を製作する頃には、現在よりも格段に高性能なデバイスが利用可能となり、十分な設計マージンが確保可能であろうと考えられる。

### 3.3 R- $\phi$ Coincidence Logic の設計とテスト

R- $\phi$  Coincidence Logic は、すでに述べたように LUT(Look-Up Table) を用いて実装される。1 つの SSC に入力される信号線は 19bit ある。これを実装するには、次の 2 通りの方法が考えられる。

- 512K 以上のリニアなメモリ空間を持つ大きな 1 個のメモリに直接 19bit を入力する。
- SSC への入力を左右の半分の SSC(halfSSC) に分けて ( R:7bit +  $\phi$  0:6bit と R:7bit +  $\phi$  1:6bit)、2

<sup>7</sup>Virtex で配置配線を行った時、FPGA のフィッター・プログラムは、すべての FPGA について 40MHz 以上との結果が結果が出た。ただし、これには FPGA 外部の配線 (基板上の配線など) は含まれておらず、複数の FPGA によるシステムに対してはあくまで参考値とすべきである。

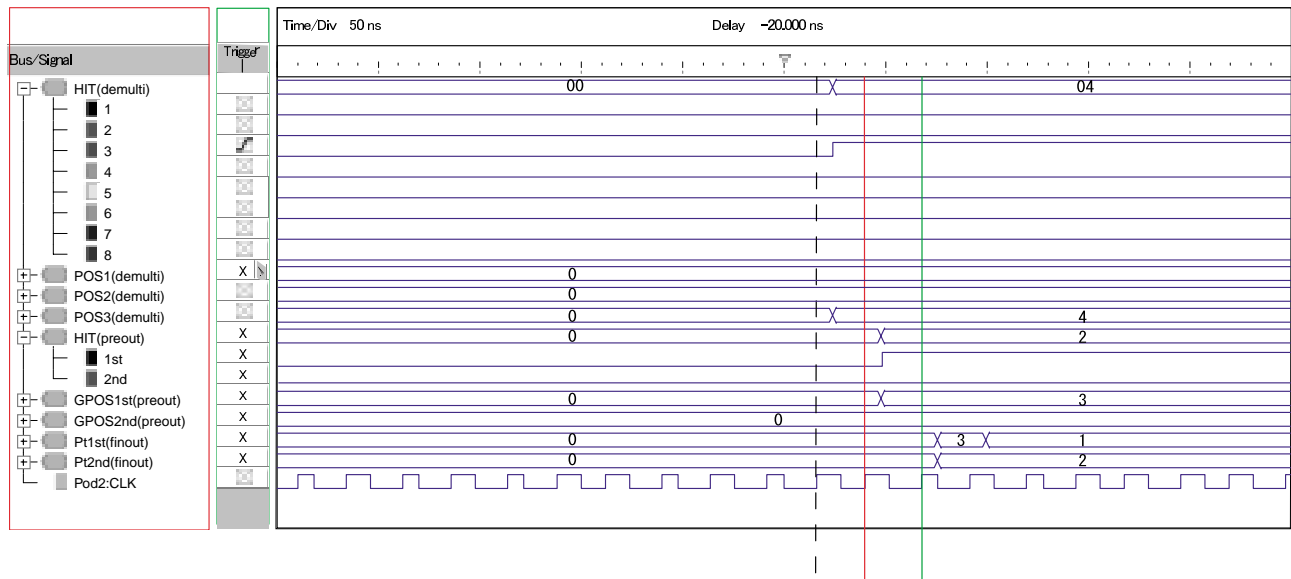


図 3.21: pt3 による Track Selection Logic の動作波形 (40MHz)

個のメモリを用いる。(8K × 2 のメモリ空間)

どちらを選択するかは、使用するメモリデバイスに依存する。

さて、SSC は最大 1track を生成する  $R$ - $\phi$  Coincidence の単位である。 $\phi$  方向の入力が同時に 2hit 入力された時、2 track の候補が生じ得るが、3.1.2 で考察したように、SSC ではどちらか一方に選別を行う必要がある。前者の 512K のメモリ空間を持つメモリの場合は、LUT のデータ中にどちらを選出するかが含まれることになる。

問題は、後者である。2つの  $\phi$  入力毎に LUT を用いる方法は、論理的には図 3.22 で表されるように、半分の SSC (halfSSC) を単位とした LUT 処理を行うことになる。

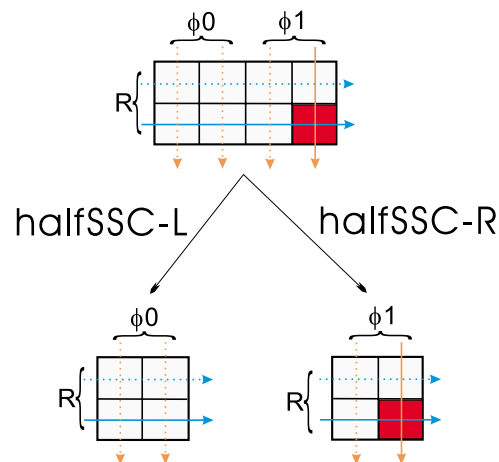


図 3.22: SSC の halfSSC への分解

左右の halfSSC からの 2track から候補を 1つにするために、次のような論理回路を用いることにした。

1. 両者の  $P_T$  の値をマグニチュード・コンパレータで比較して大きい方を採用する。
2. もし等しいときは、予め決めておいたどちらか一方を採用する。

これらのテストを行うために必要となるメモリデバイスを pt3 モジュールは持たないため、専用のテストモジュールを設計・製作した。

### 3.3.1 プロトタイプの仕様と設計

R- $\phi$  Coincidence Logic のテストモジュール slm1(Sector Logic Prototype minus1) は、pt3 と同じく、6U の VME モジュールであり、2 個の Virtex シリーズの FPGA である、XCV300-4PQ240C(QFP240 ピン(ユーザ I/O ピンは 166 ピン)) と VME コントロール用の CPLD を持ち、高速非同期 SRAM(4bit  $\times$  1M, アクセスタイム  $T_{AA}=15\text{ns}$ ) を 4 個搭載している。また、後述するようにこの FPGA には、BlockRAM というメモリブロックが内蔵されている。フロントパネルには信号を観測するためにピンヘッダがあり、32bit の信号を観測出来る。クロックは、モジュールに実装されている 40MHz のクロックの他、フロントパネルに LEMO コネクタのクロック入力が装備され、これらを切替える事が出来る。

プロトタイプ slm1 の写真を図 3.23 に、ブロック図を 図 slm1-block に示す。

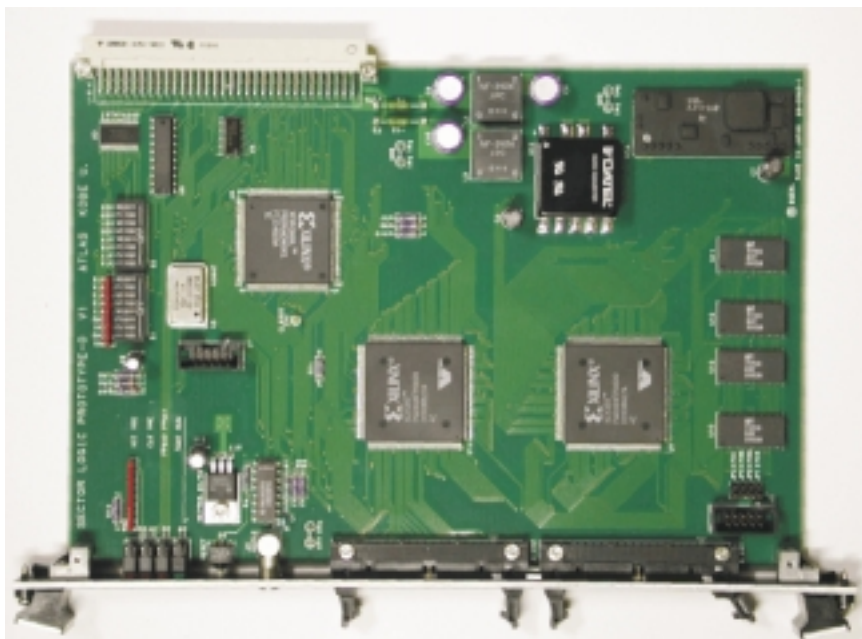


図 3.23: R- $\phi$  Coincidence Logic の実装評価用プロトタイプ (slm1)

#### 3.3.1.1 低電圧化への対応

半導体業界ではデバイスの高集積化・高速化に伴い、年々電源電圧が低電圧化している。従来の標準的な信号レベルであった 5V TTL に比べ電圧振幅が少ない分、より高速動作が可能となる。<sup>8</sup>そのため、システム全体の信号電圧も、それに合わせ 3.3V などの低電圧を用いた方が性能が向上する。さらに、今回用いた SRAM のように 3.3V の信号電圧専用のデバイスも増えて来たため、システムの低電圧化は必須事項である。また、デバイス同士の相互接続をする場合、信号電圧を統一しておかないと途中にレベルコンバータなどが必要となり、速度面からも不利になる。現在のところ、単一電圧のみの低電圧デバイス(メモリなど)は 3.3V の製品が多いことを考えるとシステム全体の信号電圧を 3.3V に統一するのが適切であ

<sup>8</sup>今日、デバイス内部の低電圧化が急速に進んでいるために、デバイスごとによって電源電圧がバラバラになっている。(3.3V, 2.5V, 1.8V,...) また、信号電圧の規格も多く存在するようになったため、デバイスの電源電圧(コア電圧)と信号電圧(IO 電圧)が別々に設定できるようになってきた



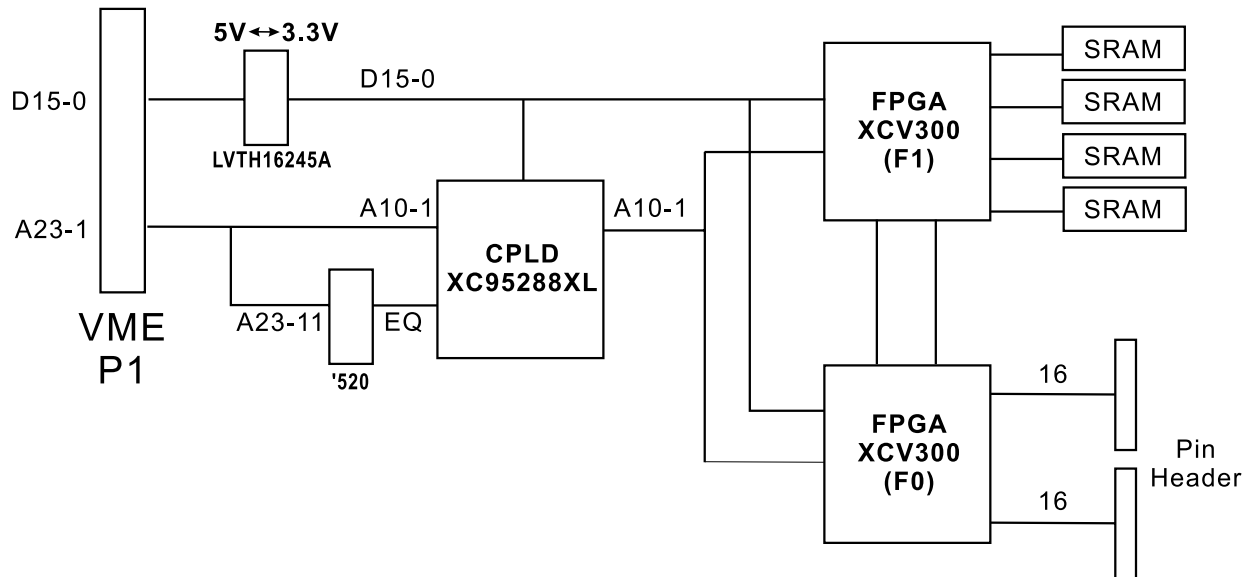


図 3.24: slm1 モジュールのブロック図

ると考えられる。

そのため、slm1 モジュールのシグナルレベルは 3.3V-TTL とし、5V-TTL の VME Bus とのインタフェースは、データバスはトランシーバ IC(SN74LVTH16245A) を用いた。また、アドレスバスと VME コントロールラインは 5V 信号への直接入出力も可能 (5V Tolerant) な CPLD を採用し、これに行わせることにした。<sup>9</sup>

### 3.3.1.2 VME コントロール CPLD と VME

VME コントロール用の CPLD には、Xilinx 社の 5V Tolerant な 3.3V CPLD である XC95288XL-7PQ208C を用いた。<sup>10</sup>

この CPLD の行う役割は次の通りである。

- VME の制御ライン (LWORD\*, BERR\*, DTACK\*, DS1\*, DS0\*, AS\*, AM0-AM5, IACK\*, SYS-RESET\*) のコントロール
- VME バスからの FPGA Configuration のコントロール
- VME バスからの FPGA アクセスのインターフェース (ストロブ信号の生成など)
- クロック信号の切替えと分配
- アドレスバスの電圧変換 (VME バス側 (5V) と内部バス (3.3V))

### 3.3.1.3 IO Emulator ブロック FPGA(F0)

FPGA(F0) は、IO Emulator ブロックを司る。つまり、実際に R-φ Coincidence を行う FPGA(F1) に対し、Hi-P<sub>T</sub> からの信号をエミュレートするためのテストパターンを生成するパターンジェネレータの働

<sup>9</sup>5V Tolerant でないデバイスに 5V の信号を与えると、最悪の場合ラッチアップを起こして過電流がながれ、デバイスや周辺回路を破壊することがある。

<sup>10</sup>pt3 では、同じ 9500 シリーズの 5V CPLD の XC95216-10HQ208C を使用している。

きをする。また、F1 から出力された結果をフロントパネルの 32 ピンのピンヘッダに (必要に応じて FF でラッチして) 出力している。このように入出力をエミュレートする回路を別の FPGA に分けたのは、観測用回路の挿入による特性変化が発生する事を防ぎ、出来るだけ本番と同じ条件で回路を評価するためである。

この slm1 モジュールでは、4 個の SRAM を用いて最大 4 SSC を扱う事が出来る。これらの入力に対応する、1 個の R 方向 (Wire) Hi- $P_T$ ASIC(最大 2hit) と 1 個の  $\phi$  方向 Hi- $P_T$ ASIC(最大 2hit) 分の入力信号が F0 から、R- $\phi$  Coincidence を行う FPGA(F1) に対し送られる。FPGA のユーザ IO ピン数の制約のため、SSC 数との関係で使われることのない HitID の上位ビットを削るなどの変更は加えてあるが、それ以外は全く Hi- $P_T$ ASIC からの信号と同じである。

便宜上、F0(IO Emulator) から F1(R- $\phi$  Coincidence Logic) に送られる信号を

- R 方向 (Wire) の Hi- $P_T$  の hit0  $\rightarrow$  WIRE0
- R 方向 (Wire) の Hi- $P_T$  の hit1  $\rightarrow$  WIRE1
- $\phi$  方向 (Strip) の Hi- $P_T$  の hit0  $\rightarrow$  STRIP0
- $\phi$  方向 (Strip) の Hi- $P_T$  の hit1  $\rightarrow$  STRIP1

とする。

F0 と F1 間の信号線についてまとめると、

信号線	方向	bit	内容
WIRE0/1	F0 ⇒ F1	0	$\Delta R[0]$
WIRE0/1	F0 ⇒ F1	1	$\Delta R[1]$
WIRE0/1	F0 ⇒ F1	2	$\Delta R[2]$
WIRE0/1	F0 ⇒ F1	3	$\Delta R[3]$
WIRE0/1	F0 ⇒ F1	4	$\Delta R[4]$
WIRE0/1	F0 ⇒ F1	5	POS
WIRE0/1	F0 ⇒ F1	6	hitID[0]
WIRE0/1	F0 ⇒ F1	7	hitID[1]
WIRE0/1	F0 ⇒ F1	8	H/L
STRIP0/1	F0 ⇒ F1	0	$\Delta \phi[0]$
STRIP0/1	F0 ⇒ F1	1	$\Delta \phi[1]$
STRIP0/1	F0 ⇒ F1	2	$\Delta \phi[2]$
STRIP0/1	F0 ⇒ F1	3	$\Delta \phi[3]$
STRIP0/1	F0 ⇒ F1	4	POS
STRIP0/1	F0 ⇒ F1	5	hitID[0]
STRIP0/1	F0 ⇒ F1	6	H/L
SSC0/1/2/3	F0 ⇐ F1	0	$P_T[0]$
SSC0/1/2/3	F0 ⇐ F1	1	$P_T[1]$
SSC0/1/2/3	F0 ⇐ F1	2	$P_T[2]$
SSC0/1/2/3	F0 ⇐ F1	3	$\phi [0]$
SSC0/1/2/3	F0 ⇐ F1	4	$\phi [1]$
SSC0/1/2/3	F0 ⇐ F1	5	R
SSC0/1/2/3	F0 ⇐ F1	6	hit

となる。

### 3.3.1.4 R- $\phi$ Coincidence Logic ブロック (F1)

R- $\phi$  Coincidence Logic ブロック (F1) は、Decoder と R- $\phi$  Coincidence Logic から構成される。

Decoder では、F0 から入力された信号を SSC 毎に振り分ける。(詳細は 3.1.4.1 節を参照) また、 $\phi$  方向 (Strip) の Hi- $P_T$  からの信号は、全ての SSC に分配される。SSC に入力される信号は次の通りである。<sup>11</sup>

<sup>11</sup>表の中で、 $\phi$  (Right) や  $\phi$  (Left) は、それぞれ SSC の左半分に入力される hit と右半分に入力される hit である。このような扱いは Hi- $P_T$  の構造上、hit0(hit1) が  $\phi$  (Right) になることもあれば、 $\phi$  (Left) になることもあるためである。

bit 範囲	信号名	内容
0..4	R	$\Delta R$ (5bit)
5	R	POS
6	R	H/L
7..10	$\phi$ (Right)	$\Delta\phi$ (4bit)
11	$\phi$ (Right)	POS
12	$\phi$ (Right)	H/L
13..16	$\phi$ (Left)	$\Delta\phi$ (4bit)
17	$\phi$ (Left)	POS
18	$\phi$ (Left)	H/L

R- $\phi$  Coincidence Logic では、SSC に振り分けられた 19bit の入力を LUT にアドレス入力として送り、4bit のデータ出力を SSC の出力に用いる。

R- $\phi$  Coincidence Logic の LUT 周りの回路は、図 3.25 のようなブロック図<sup>12</sup>で表される。

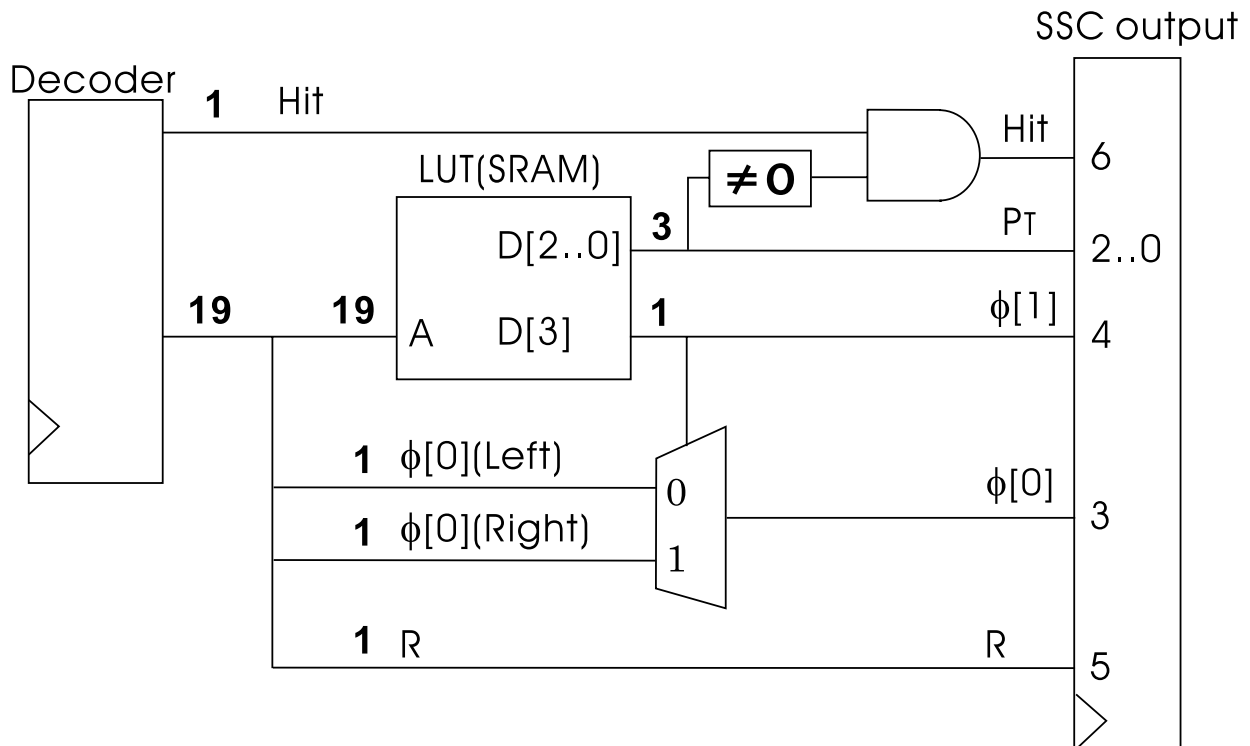


図 3.25: R- $\phi$  Coincidence Logic の LUT 周りの回路

LUT のデータ出力の 3bit 目は、LUT から出力される  $P_T$  の Track が  $\phi$  (Left) と  $\phi$  (Right) のどちらからの入力に依るものなのかを示す bit であり、 $\phi$  [1] となる他、 $\phi$  [0] の切替えにも使われる。また、Decoder からの Hit 信号は、SSC に hit 入力が全く無いときに SSC からの hit 出力をマスクするものであり、LUT のデータが何らかの原因で間違っていた場合へのリメディとして用意してある。

この図で SSC Output に FF が付けてあるのは、これは R- $\phi$  Coincidence の処理以外の伝搬遅延を出来るだけ受けない、より良い条件で測定を行うためのものである。この FF を F0 に移設すると、2 個の

<sup>12</sup>この図は、LUT に 19bit 入力を持つ SRAM を用いたものであるが、halfSSC を用いた実装の場合、2 個の LUT ブロックと マグニチュード・コンパレータで置き換えられる。

FPGA の IO 遅延と基板上の電氣的遅延を受け、動作速度は低下する。<sup>13</sup>

### 3.3.2 プロトタイプの動作検証

slm1 では LUT に用いるメモリデバイスとして、4 個搭載している SRAM の他に FPGA に内蔵されている BlockRAM とよばれる メモリブロックがある。この節では、それぞれのデバイスを用いた LUT の実装とその評価結果について述べる。

#### 3.3.2.1 SRAM を用いた設計

現在生産されている 高速非同期 SRAM は、容量が 1Mbit 以上の製品が主流である。小容量の製品は設計が古く、ほとんど低速なデバイスしか存在しない。そのため、SSC を halfSSC に分割して扱う方法は使えない。そこで、512K 以上のメモリ空間を持つ、4bit 幅の 高速非同期 SRAM を用いることにした。

slm1 モジュールに用いた SRAM は、IDT 社製の IDT71V428S15Y という 4bit×1M、アクセスタイムが (やや遅い) 15ns の製品である。<sup>14</sup>メモリ空間は 4M あるが、最上位 bit(MSB) を使わないことで、半分の容量として使用している。

LUT の動作中は、SRAM の  $\overline{OE}$ ,  $\overline{CS}$  を Low(Active) に、 $\overline{WE}$  を High(Inactive) に固定する。これにより、SRAM はアドレス入力の確定後、アクセスタイム時間後にデータが出力される組み合わせ回路として動作を開始する。

また、slm1 モジュールには 4 個の SRAM が搭載されているため、4SSC をカバーしている。

#### 3.3.2.2 SRAM 版の結果

図 3.26 は、40MHz のシステムクロックで SRAM 版の R- $\phi$  Coincidence 回路を動かした時の正常<sup>15</sup>に動作している波形である。

この波形で、 $P_T$  の値の 7 は、nohit を意味する。(図 3.14 参照)

まず、(1) のタイミングで F0 から F1 にデータが出力される。(WIRE0/1, STRIP0/1) このデータは (2) のタイミングで F1 の Decoder の FF でラッチされ、その出力が SRAM にアドレス入力として送られる。SRAM では、アドレス入力が確定した時点からメモリアクセス時間 (=15ns) 後に対応するデータを F1 に戻し、残りの組み合わせ回路 (図 3.25) を経て、SSC output となる。この結果は、(3) のタイミングで F1 内部の FF でラッチされ、F0 を経由してピン・ヘッダに出力される。(SSC0/3 の hit, R, PHI[1], PHI[0])

この図では、(3) のタイミングから 3/4 周期ほど経った時点でデータが出力されているが、これは F1 の FF からの出力が一旦基板上に出力され、それが F0 内部の配線を経由して、再び基板上に出て観測用のピンヘッダに至るまでの遅延時間である。したがって、ここの遅延は回路動作に影響がないため 1 周期を越えても問題ない。むしろ、観測すべきことは (3) のタイミングで F1 内の FF に正常<sup>16</sup>にデータがラッチされるかである。

このように観測しながらシステムクロックを上げていった所、46.0MHz まで安定して動作することが確認された。

<sup>13</sup>後述の BlockRAM を用いた実装の場合、クロック周期にして約 5ns 程度の増大が観測された。

<sup>14</sup>実際、LUT として必要なアドレス空間は、19bit 即ち 512K で足りる。しかしながら、高速非同期 SRAM は限られた市場の中で取り引きされているためにスポットでの購入が極めて困難で、最適な製品が選べる状況では無かった。そこで、使える可能性のある製品を全て手配して待ち続けた結果、この製品のみが入手出来たのである。データシートにある (量産中の) 最も速い製品は、アクセスタイムが 8ns である。

<sup>15</sup>クロックの duty cycle が時間的に変動しているが、これはロジック・アナライザ (Agilent Technology 社製 E9340A) のサンプリング周波数が 250MHz (=4ns) と測定対象にたいして、十分に速くないために起こる見かけの現象である。

<sup>16</sup>メタステーブル状態にならないということ

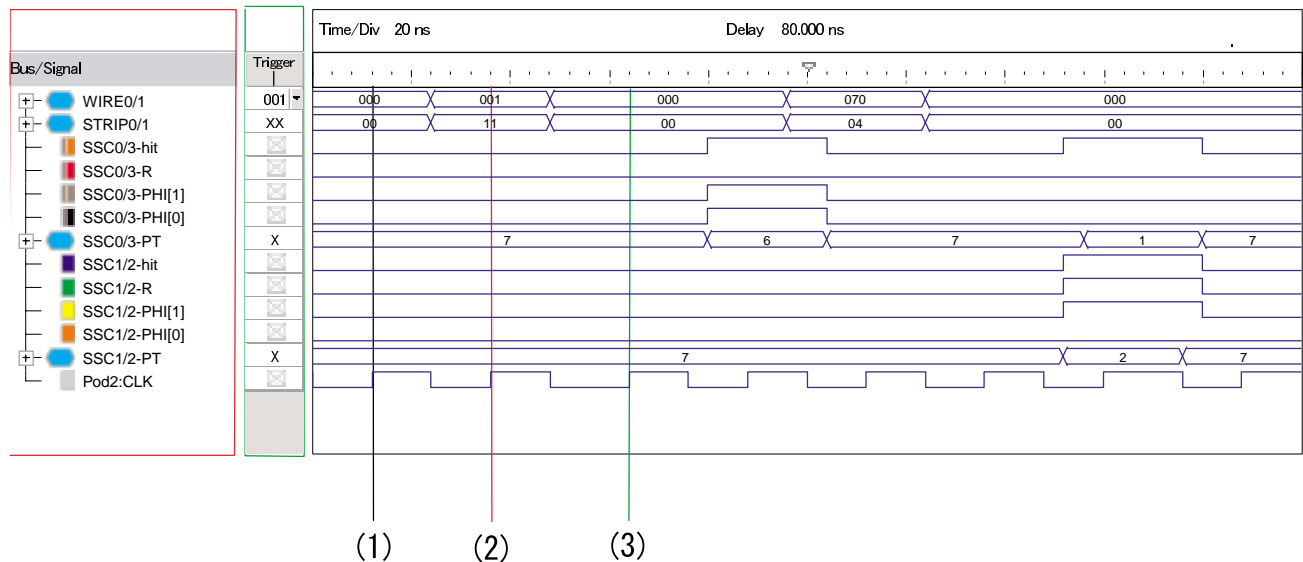


図 3.26: R- $\phi$  Coincidence SRAM 版 (40MHz)

### 3.3.2.3 BlockRAM を用いた設計

BlockRAM は、Xilinx 社の Virtex シリーズの FPGA に on-chip で備えられている 1 個当たり 4096 bit の同期 RAM ブロックの名称であり、正式には Block SelectRAM+ と呼ばれる。デュアルポートあるいはシングルポートメモリとして使用することが可能であり、アスペクト比 (アドレスの深さとデータ幅の比) を 1bit×4K から 16bit×256 までの範囲で設定することが出来る。

slm1 に用いた XCV300-4PQ240C の場合 16 個の BlockRAM を持っており、64Kbit のメモリとなる。また、2000 年第 3 四半期に出荷が開始されたばかりの Virtex-E Extended Memory(Virtex-EM) シリーズの FPGA は、1 個当たり 560Kbit(XCV405E)、1120Kbit(XCV812E) と、かなり大規模に BlockRAM を持った FPGA も登場して来ている。

BlockRAM を R- $\phi$  Coincidence の LUT に用いる場合の特長と問題点は次のようになる。

クロックの立上りからデータが出るまでの遅延時間 ( $T_{BCKO}$ ) は、Virtex の場合、最も遅いスピードグレードが -4 の製品でも 4.3ns であり、さらに内部配線のみでロジック回路と接続されるので、一般の SRAM よりも高速動作が期待できる。

FPGA の Configuration データで BlockRAM のデータの初期値を与えたり、Readback 機能によって実験中のデータの照合をする事ができるため、実験中のメンテナンスが容易になる。

- × Read Through 式の同期式 RAM であるため、非同期 SRAM には不要だった読み出しのための 1clock が必要である。<sup>17</sup>
- × 1 個のデバイスに搭載されている RAM 容量は、単体の SRAM よりも一桁少ない。
- × 一社 (Xilinx 社) の特定のアーキテクチャに依存した設計であり、将来に渡って製品が供給され続ける保証がない。

このため、BlockRAM を用いた R- $\phi$  Coincidence を作るには、SSC を halfSSC に分ける方法を探る必要がある。

BlockRAM を用いて halfSSC の LUT を作るには、

<sup>17</sup>多くの単体の同期式 RAM は、Read Pipelined 方式を採用しており、この場合読み出しには 2clock が必要となる。

1. 1bit×4K の構成にした BlockRAM を 6 個組み合わせる。(1bit 版)
2. 4bit×1K の構成にした BlockRAM を 8 個組み合わせる。(4bit 版)<sup>18</sup>

の 2 通りが考えられ、1 個の SSC を作るのに、前者の方法 (1bit 版) では 12 個の BlockRAM が、後者の方法 (4bit 版) では 16 個の BlockRAM が、それぞれ必要になって来る。

幸いなことに、slm1 モジュールで使用している XCV300-4PQ240C は BlockRAM をちょうど 16 個持っているため、扱う SSC 数を 1 個に限定し、両方の方法を試すことが出来た。

### 3.3.2.4 BlockRAM 版の結果

BlockRAM を用いて LUT を実装した時の波形を、図 3.27 に示す。

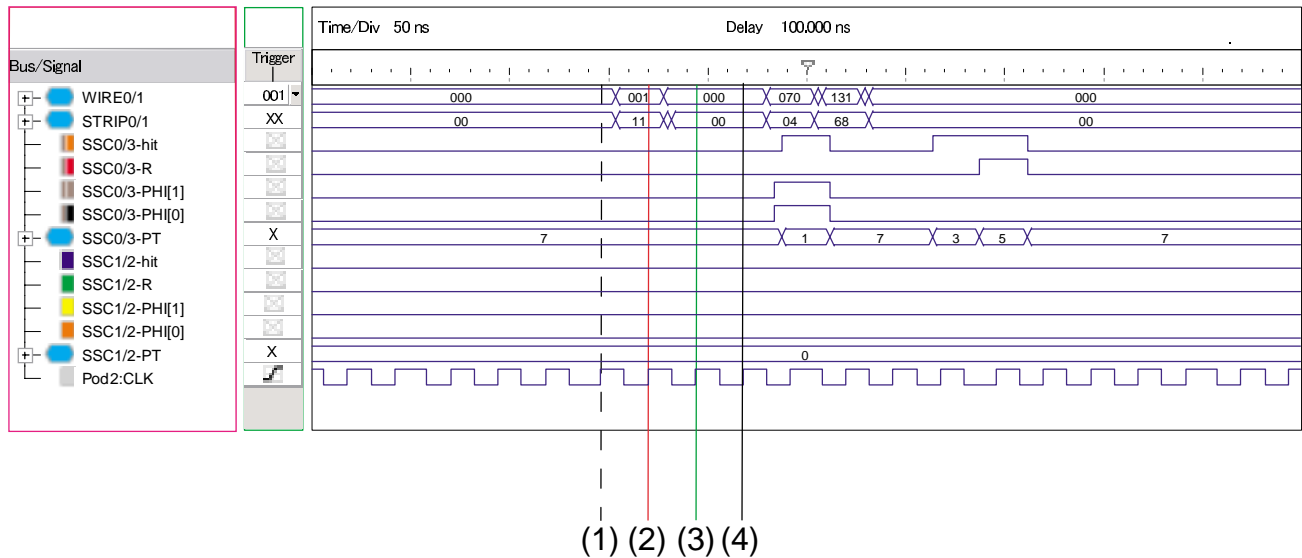


図 3.27: R- $\phi$  Coincidence BlockRAM 4bit 版 (40MHz)

これを見ると、SRAM 版では、(3) のクロックの立上りの後に結果が出力されたのに対して、BlockRAM 版では、(4) のクロックの立上りの後に結果が出力されており、全体として 1clock 余分に掛かっている。これは BlockRAM が同期式のメモリであるためであり、Decoder からの出力 (2) の後、すぐに非同期式の SRAM はメモリアクセスを行うのに対し、BlockRAM は次のクロックの立上り (3) でメモリアクセスを行っているために生じる差である。

なお、これは 4bit 版の波形であるが、この 1clock の遅れを除いては SRAM 版と同様に正しく動作している。また、1bit 版も 40MHz で問題なく動作した。そこで、1bit 版と 4bit 版それぞれについて、入力するクロックの周波数を少しずつ上げて行き、最大動作周波数を測定した。すると、

1bit 版 55.8MHz

4bit 版 57.4MHz

と、どちらも SRAM 版の最高動作周波数を大きく上回った。また、若干の差であるが 4bit 版の方が高速であったため、4bit 版について、さらに論理合成・配置配線時にタイミング制約を付加し、最適化レベルを高めてみたところ、最高動作周波数は 60.6MHz の最高値を達成した。このときの、(限界ぎりぎりの) 動作波形を図 3.28 に示す。

<sup>18</sup>残る 1bit は使わない。

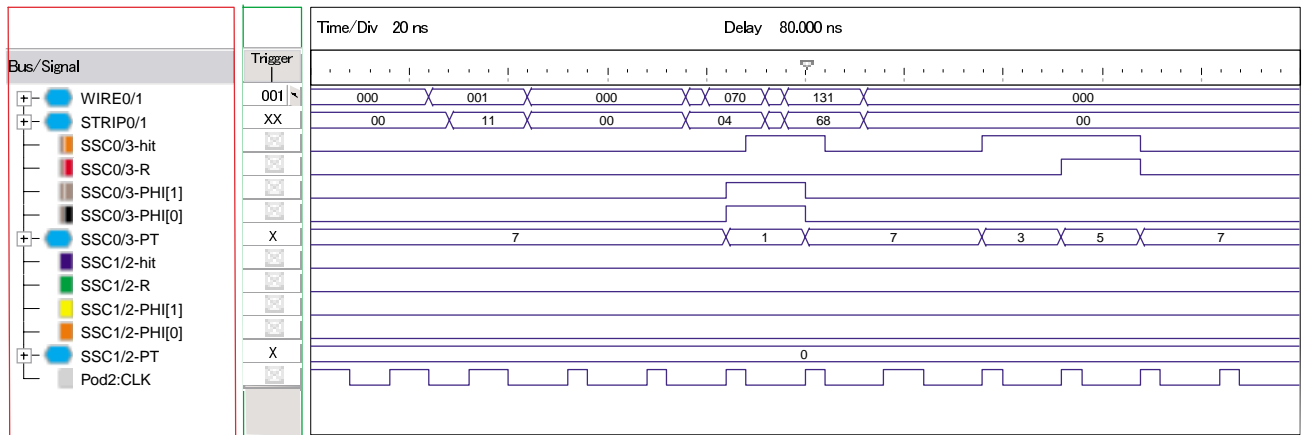


図 3.28: R- $\phi$  Coincidence BlockRAM 4bit 版 [最適化](60.6MHz)

### 3.3.2.5 どちらを使うか?

この結果および高速非同期 SRAM の市場での入手性を考えると、現在のところ、BlockRAM を用いた方が得策であると考えられる。そのため、次のプロトタイプ Prototype-0 では BlockRAM を用いることにした。

しかしながら、SRAM 版の結果も SRAM のアクセスタイムが 15ns と比較的遅い製品だった事を考慮すれば、全く遜色のない結果であったと言える。そのため、SRAM 版の実装は、将来 BlockRAM を用いるに際して何らかの問題が発生した場合に備えた保険として取っておくことにする。

## 3.4 Prototype-0 の設計

Prototype-0 は、ほぼ全機能を実装した Forward Trigger Sector 用の Sector Logic モジュールである。このモジュールの製作目的は、基礎設計から開始し各機能ブロックの検証を終わらせた Sector Logic 回路について、全体を統合させた状態で設計目的通りの機能を果たしているかを検証し、本番用の Sector Logic に最適な実装方法を探求することである。そのため、可能な限り実際の ATLAS 実験と同じ条件で動作させられるように設計する必要がある。

また、この Prototype-0 は、2001 年 9 月に予定されている、前後方ミュオントリガシステムの第 1 回フルシステムテストに用いられることになっており、Hi- $P_T$  モジュールのプロトタイプと実際に接続し、TTC から 40.08MHz の LHC クロックを与えてテストされることになっている。

### 3.4.1 要求される仕様と構成

本番では Sector Logic は、9U の VME モジュールに実装される。これが収納される Sector Logic クレートには、2 オクタント分 18 枚の Sector Logic モジュールの他に、VME コントローラモジュールと readout のための Star Switch モジュールが各 1 枚スロットに収納されるほか、Sector Logic モジュールに 1 対 1 に対応し、Optical Interface を受け持つ 18 枚のトランジション・ボードがクレートの裏側から収納される予定である。Optical Interface 部分をトランジション・ボードに分けたのは、実験期間中のメンテナンスの利便性を考慮したためである。また Sector Logic モジュールとトランジション・モジュール



ルの間は1対1の通信であることから、VME クレートの J3 の位置の一部を切抜きにして、両者が直接高密度コネクタで結合させる予定である。

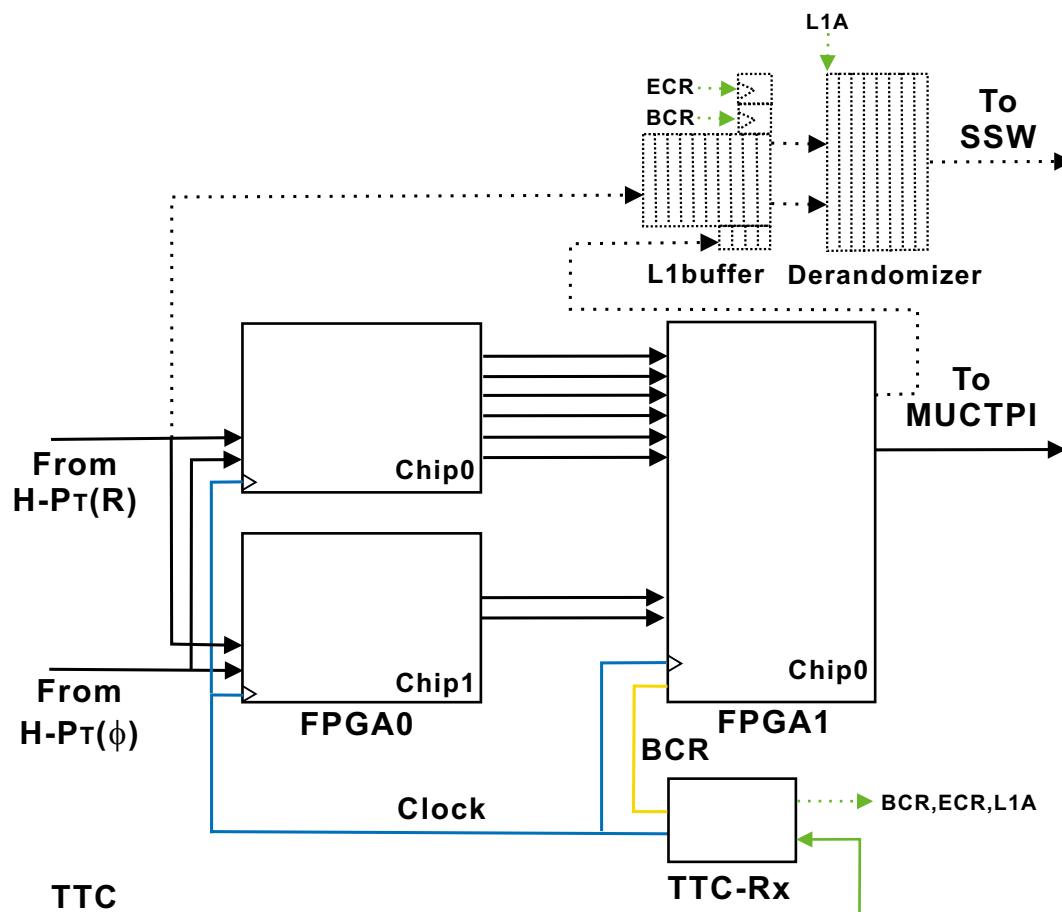


図 3.29: Prototype-0 の主要部のブロックダイアグラム

図 3.29 は、Prototype-0 の主要部のブロックダイアグラムである。今回は実装しない Readout(3.4.3.3 節) のための L1buffer、Derandomizer およびそのための配線などは図中では点線で描かれている。

一方、この Prototype-0 は Forward Sector Logic であるため、EI/FI による選別は行わない。<sup>19</sup>

Prototype-0 は、本番用と同じ 9U の VME モジュールとして実装される。しかし、本番用と違い、専用のバックプレーンやトランジション・ボードは用いず、標準的なバックプレーンでテスト出来るようにする。<sup>20</sup>

### 3.4.2 VME コントロール CPLD

VME コントロール用の CPLD には、slm1 と同じく、Xilinx 社の 3.3V CPLD である XC95288XL-7PQ208C を採用した。この CPLD の行う役割は次の通りである。

- VME の制御ライン (LWORD\*, BERR\*, DTACK\*, DS1\*, DS0\*, AS\*, AM0-AM5, IACK\*, SYS-RESET\*) のコントロール
- VME バスからの FPGA Configuration のコントロール

<sup>19</sup>EI/FI による選別は、EndCap Sector Logic のみである。

<sup>20</sup>今回のようなプロトタイプテストでは本番と違って、隣同士のモジュール間の物理的干渉を考慮する必要がないため、基板表面に出入力コネクタを実装してもかまわない。

- VME バスからの FPGA アクセスのインターフェース (ストロブ信号の生成など)
- クロック信号の切替えと分配
- アドレスバスの電圧変換 (VME バス側 (5V) と内部バス (3.3V))

### 3.4.3 入出力ブロック

この Prototype-0 の持つ入出力信号は、Hi-P<sub>T</sub> からの Optical 入力 (G-link)、MUCTPI への LVDS 出力、TTC 信号などである。Optical 入力に関しては、デバックやテストの容易さを考慮して、Optical 入力と 電気信号の両方を切替えて入力できるようになっている。

#### 3.4.3.1 Hi-P<sub>T</sub> からの入力

実験ホール内にある Hi-P<sub>T</sub> からの信号は、G-link という Optical Interface によって Sector Logic のある USA15 に送られてくる。3.1.4.1 節で述べたように、Forward 領域には、F0-Chip0(R), F0-Chip1(R), F0-Chip2( $\phi$ ) の 3 個の Hi-P<sub>T</sub>ASIC があり、それぞれ 1 本の G-link Optical Cable を持つ。なお、G-link Optical Cable は、1 本あたり 20bit のデータ幅である。

#### 3.4.3.2 MUCTPI への出力部

MUCTPI へは、LVDS で 図 3.14 の 32bit フォーマットで出力する。使用するコネクタは、ハロゲンフリーのツイストペアケーブル用の 0.635mm ピッチ、68 ピンの製品であり、Robinson 社の P25LE-068S-TGF が KEL 社の 8925E/R が指定されている。詳しくは、Farthouat [5] を参照のこと。

#### 3.4.3.3 TTCrx と Readout Buffer

Prototype-0 は、クロックを始めとする TTC の信号を受信する TTC Rx を装備する。しかし、今回は Readout Buffer を持たないため、TTC のうち使用する信号は、クロックと BCR(Bunch Counter Reset) だけである。なお、BCR は MUCTPI に送るデータを生成する Encoder が使用する。

### 3.4.4 FPGA-0(XCV405E-6BG560C × 2)

この FPGA ブロックでは、R- $\phi$  Coincidence を行う。ここで使用する XCV405E-6BG560C という FPGA は、Virtex-E Extended Memory(Virtex-EM) シリーズの FPGA であり、560Kbit の BlockRAM を持つ。この FPGA ブロックの 2 個の FPGA は、それぞれ次のような領域を受け持つ。

chip#	MaxHit	Subsector	SSC	Hi-P <sub>T</sub> (R)	Hi-P <sub>T</sub> ( $\phi$ )
0	2	0-11	00, 01, 02, 03, 04, 05	F0-chip0	F0-chip2
1	2	12-15	06, 07	F0-chip1	F0-chip2

一方、これらの FPGA の各入出力は、1 信号当り次のようなビット・アサインメントになっている。

		9	8	7	6	5	4	3	2	1	0
Input	Hi-P <sub>T</sub> (R) (10bit)	H/L	HitID	HitID	HitID	Pos	$\Delta R$	$\Delta R$	$\Delta R$	$\Delta R$	$\Delta R$
	Hi-P <sub>T</sub> ( $\phi$ ) (9bit)	H/L	HitID	HitID	HitID	Pos	$\Delta \phi$	(NA)	$\Delta \phi$	$\Delta \phi$	$\Delta \phi$
Output	(8bit)			Hit	sign	R	$\phi$	$\phi$	Pt	Pt	Pt

ここで、各入出力のそれぞれの総 bit 数は、

$$\text{Input} = 10\text{bit} \times \text{MaxHit} + 9\text{bit} \times 2$$

$$\text{Output} = 8\text{bit} \times \#\text{SSC}$$

と表されるので、トータルの IO 数は次のようにまとめられる。

Chip#	MaxHit	#SSC	Input	Output	VME	Others	Total
0	2	6	38	48	32	16	134
1	2	2	38	16	32	16	102

### 3.4.5 FPGA-0(XCV400E-6BG432C × 1)

このFPGAブロックでは、Track Selection Logic(Track Pre-Selector, Track Selector) および、Encoderの役割を行う。この Virtex-E シリーズの FPGA は、直接 LVDS 信号の扱う事ができるため、外付けバッファは必要ない。

さてこの FPGA の入出力信号は、

入力: FPGA-0 からの入力 7bit × 8(total# of SSC)

出力: Encorder 出力 32bit

となり、各入出力の 1 信号当りのビット・アサイメントは次のようになる。

		7	6	5	4	3	2	1	0
Input	FPGA-0(8bit)	Hit	sign	R	$\phi$	$\phi$	Pt	Pt	Pt
Output	To MUCTPI(32bit)	res	res	BCID	BCID	BCID	res	res	Pt2
		Pt2	Pt2	Pt1	PT1	PT1	res	res	res
		ROI2	ROI2	ROI2	ROI2	ROI2	ROI2	res	res
		res	ROI1	ROI1	ROI1	ROI1	ROI1	ROI1	[note1]

res: 予約ビットを意味し、常時 "1" を出力することになっている。

note1: 2 候補より多くのが 1 つの Sector 内から発見された。

結局、FPGA-1 のトータルの IO 数は次のようにまとめられる。

Chip#	#SSC	Input	Output	Readout	VME	Others	Total
0	8	64	32	18	32	16	162

## 第4章 まとめ

Sector Logic 回路に与えられた仕様に基づいて基礎設計から始め、主要な機能ブロックについてプロトタイプを試作して実現可能性を検証し、その結果を用いてほぼ全機能をもつ Prototype-0 の設計を行った。

また、大規模な論理回路である Sector Logic 回路を開発するにあたって、ハードウェア記述言語 (HDL) を用いた設計技法を採用し、使用するデバイスに基盤上に実装したまま自由に設計変更の出来る FPGA(Field Programmable Gate Array) を選択した。この結果、トリガとしての柔軟性を維持しつつ、要求される高速な回路動作を達成する事が可能となった。

まず、始めに検討したのは、入射ミューオンの横運動量 ( $P_T$ ) を構築する R- $\phi$  Coincidence である。横運動量 ( $P_T$ ) 測定用のトロイダル・マグネット群の発生する磁場が均一磁場にはならないため、R 方向、 $\phi$  方向それぞれのトラックの曲がり情報と hit のあった subsector 位置の情報を用いてミューオンの横運動量 ( $P_T$ ) の構築を行う必要があった。さらに、6 段階ある  $P_T$  の閾値は可変でなければならないため、入力データと  $P_T$  値の対応はディテクタシミュレーション及びトリガシミュレーションを行って決定される対応マップを作成することが必要となる。そのため R- $\phi$  Coincidence には、Look-Up Table(LUT) 方式を採用することにした。そして、このような方式を取ったために、R 方向と  $\phi$  方向の入力が異なったミューオンの横運動量の値となるような場合 - ノイズや fake hit など - については、track 候補から排除する事が可能になった。

Sector Logic はまた、短時間での処理が要求される Level-1 トリガシステム (LVL1) の一部であり、7clock(=175ns) という短時間 (かつ一定時間) で全処理を終えなければならない。この制約条件を確実に満たす回路形式として、R- $\phi$  Coincidence に LUT を採用することを考慮に入れた結果、基本構造として、パイプライン構造を採用することにした。

残る主要な機能ブロックの Track Selection Logic については、40.08MHz という比較的高いクロックに同期して動作させるために、Track Pre-Selector と Track Selector の 2 段に回路を分け、さらに各々で処理の並列度を高めることにより高速化を計った。Track Pre-Selector と Track Selector は、入力する信号数が異なるだけで基本となる構造がほとんど同一である。そのため、同一の HDL による回路記述をそれぞれの回路に対応させることにより、効率的な設計が可能となった。

この基本設計に基づいて、各機能ブロック (R- $\phi$  Coincidence, Track Selection) を実装したプロトタイプモジュールを製作・評価した。R- $\phi$  Coincidence 回路のプロトタイプでは、高速の非同期 SRAM を用いたものと FPGA 内部のメモリブロックである BlockRAM を用いたものの両者について実装・評価を行い、ともに設計仕様を満足させることが可能である事が確認出来た。また、Track Selection Logic 回路については、汎用ロジックボードである、pt3 モジュールを使った実装・評価を行った。このプロトタイプリングの動作速度は 34.0MHz と、目的とする 40.08MHz に若干及ばなかった。しかし、より高速なデバイスを使用した専用の回路を用いれば、目標の動作速度を達成することは十分に可能であると考えている。

Prototype-0 は、Forward Trigger Sector 用の Sector Logic のほぼ全機能を実装した初のプロトタイプモジュールである。これは、2001 年 9 月に予定されている、前後方ミューオントリガシステムの第 1 回フルシステムテストに用いられ、本番用と同じ 9U の VME ボードとして実装される。フルシステムテストでは、Hi- $P_T$  モジュールのプロトタイプと実際に接続し、TTC から 40.08MHz の LHC クロックを与えてテストを行う。これに含まれない、readout の機能および、専用バックプレーンについては、その次のプロトタイプ (Prototype-1) で実装される予定である。

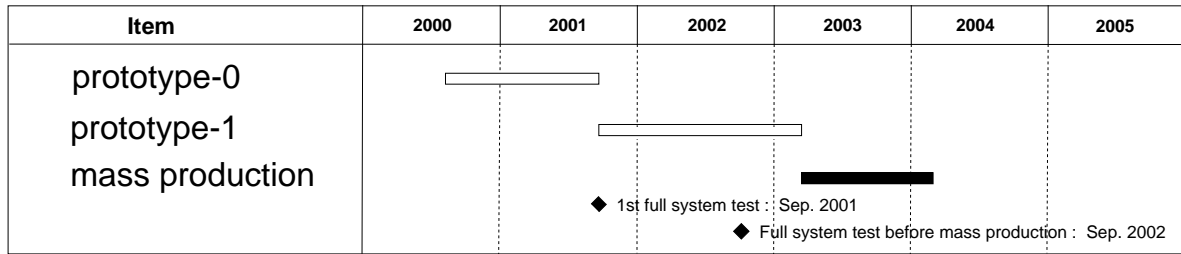


図 4.1: Sector Logic の開発・量産スケジュール

今後おこなうこととしてハード面では、フルシステムテストの結果を活かしてより本番用に近づけた Prototype-1 の設計を行うことである。この作業には、専用バックプレーンの設計、オプティカル・インターフェース用のトランジションボードの設計、Sector Logic クレートに入れた Sector Logic ボードをコントロールするシステム(クレートコントローラ及びサーバシステムなど)の開発などが含まれる。

また、ソフト面では、 $R\text{-}\phi$  Coincidence の Coincidence map を作成するためのディテクタ・トリガシミュレーションを行なう必要がある。この Coincidence map の正確さによって、Sector Logic ひいてはトリガシステムの性能(トリガ効率)が決まることになるので、トリガ効率の良い Coincidence map の生成方法を確立しなければならない。

一方、素粒子反応の断面積を高精度で測定するためには、トリガ効率の正確な値が不可欠である。この値は、ディテクタ・トリガシミュレーションによって求める以外に方法はない。したがってこのシミュレーションは、ATLAS 実験の測定を行う上で必要となる基礎データとなるため、高精度な結果が要求される。

また、Sector Logic を含む Level-1(LVL1) トリガシステムの出力(L1A)は、全ての測定器からのデータを用いて判断する Level-2(LVL2) トリガシステムにも判断材料を与える有効な情報として送られる。したがって、Higgs 粒子生成や SUSY 粒子生成などの物理現象に關与する様々なプロセスに対して、LVL2 トリガシステムまで含めて、それぞれのトリガに対するトリガ条件を決める必要がある。Sector Logic で行う muon の  $P_T$  の閾値もその一つであり、この値を決定することを目標のひとつとして、トリガシステムの総合的な研究を行うことが今後の課題である。

# 謝辞

本研究を行う機会を与えて頂き、研究を進めるにあたり、議論を通じて適切な助言を頂いた藏重 久弥助教授に深く感謝いたします。同助教授の私の研究への深い理解と忍耐強い支援なしには、この研究は成し遂げられなかったことでしょう。また、研究上の細かい点にいたるまで丁寧に御指導いただきました、武田 廣教授には、講義やコロキウムを始めとする日常の研究生活の中で素粒子物理学の指導を賜りました。また、同教授は ATLAS 日本グループの代表の一人として、全体の維持運営に携わっておられます。同じく神戸大学の野崎 光昭氏、川越 清以氏、原 俊雄氏、本間 康浩氏、石井 恒次氏、鈴木 州氏には、日常の研究生活の中で、様々な指導・助言を頂きました。ここに、心より感謝申し上げます。

KEK(高エネルギー加速器研究機構)の佐々木 修氏には、研究上の方針から細部に至るまで、常に丁寧に説明して下さい、快く議論をして下さいました。ここに深く感謝申し上げます。また、研究を進める上でエレクトロニクスのノウハウや実装方法などを親身になって説明して下さいました KEK 回路室の池野 正弘氏にも、心よりお礼を申し上げます。プロトタイプモジュールの製作を行って下さった株式会社ジー・エヌ・デーの宮沢 正和氏に感謝いたします。京都大学の坂本 宏氏、信州大学の長谷川 庸司氏、東京都立大学の福永 力氏、ICEPP(東京大学素粒子物理国際研究センター)の小林 富雄氏、蓮子 和巳氏、KEKの近藤 敬比古氏、新井 康夫氏には、各々の専門家としての立場から数え切れない程の有益なアドバイスを頂き、様々な面でご支援を頂きました。心より感謝の意を表明いたします。

さらに ATLAS TGC エレクトロニクスグループで、状況に応じて適切なアドバイスを頂き、多くの事を学ばせて頂いた、狩野 博之氏(東京都立大)、西田 昌平氏(京都大)、仁木 太一氏(東京大 ICEPP)、戸谷 大介氏(同)にお礼を申し上げます。また、同グループでともに研究を行い議論をした、香取 勇一氏(東京大 ICEPP)、溝内 健太郎氏(京都大)、田中 賢一氏(東京都立大)、石田 康明氏(同)、戸塚 真義氏(信州大)、ありがとうございました。最後になりましたが、ATLAS Collaboration の皆様及び、神戸大学で共に研究生活を送り多くの助言を頂いた、大和 一洋氏、田中 康之氏を始めとする神戸大学の皆様にお礼を申し上げます。

## 参考文献

- [1] ATLAS Collaboration *ATLAS Detector and Performance Technical Design Report, ATLAS TDR 14,15, CERN/LHCC/99-14,15(1999)*
- [2] ATLAS Collaboration *First-Level Trigger Technical Design Report, ATLAS TDR 12, CERN/LHCC/98-14(1998)*
- [3] ATLAS Collaboration *Muon Spectrometer Technical Design Report, ATLAS TDR 10, CERN/LHCC/97-22(1997)*
- [4] *Naming and Numbering Scheme for the Endcap Muon Trigger System (draft, 5 June 2000)*  
[http://www.icepp.s.u-tokyo.ac.jp/~hasuko/atlas/doc/numbering\\_draft.pdf](http://www.icepp.s.u-tokyo.ac.jp/~hasuko/atlas/doc/numbering_draft.pdf)
- [5] Ph. Farthouat, *Interfaces & overlaps in the LVL1 muon trigger system, Revision 9 (8 Nov. 2000)*  
[http://edmsoraweb.cern.ch:8001/cedarnew/doc.info?document\\_id=114604&version=3](http://edmsoraweb.cern.ch:8001/cedarnew/doc.info?document_id=114604&version=3)
- [6] 西田 昌平, 修士学位論文 *ATLAS TGC エレクトロニクス読みだし系の開発* (2000)
- [7] 戸谷 大介, 修士学位論文 *ATLAS 実験ミュオントリガシステムにおける運動量選別用 ASIC の開発* (2000)
- [8] 仁木 太一, 修士学位論文 *ATLAS 実験ミュオントリガー用タイミング調整 IC の開発* (2000)
- [9] 香取 勇一, 修士学位論文 *アトラス実験ミュオントリガシステム用 IC の開発* (2001)
- [10] Xilinx "Virtex-E 1.8V FPGA Datasheet Ver1.7"  
<http://www.xilinx.com/partinfo/ds022.pdf>
- [11] Xilinx "Virtex-E 1.8V Extended Memory FPGA Datasheet Ver1.2"  
<http://www.xilinx.com/partinfo/ds025.pdf>
- [12] Xilinx "XC95288XL High Performance CPLD Ver1.2"  
<http://www.xilinx.com/partinfo/95288xl.pdf>