修士学位論文
アトラス実験ミューオントリガーシステム用ICの開発

東京大学理学系研究科
物理学専攻

96055
香取 勇一

2001年1月

概 要

2005 年に CERN (欧州原子核研究機構) において世界最大の重心系衝突エネルギー (14TeV) を達成する LHC (大型陽子陽子衝突型加速器) 実験が開始される予定である。そのビーム衝突点に設置される ATLAS 検出器は、素粒子の質量獲得起源を担う Higgs 粒子、また重力を含めた 4 つの力を統一する理論の手がかりになる超対称性粒子の発見を主な目的とする汎用粒子検出器である。この実験では 40MHz と高頻度にバンチ衝突が起こり、また多くのバックグランドイベントが生成されるため、この中から効率良く重要なイベントを選別するトリガーシステムが重要である。本論文ではトリガーシステムのうち、TGC(Thin Gap Camber) と呼ばれる ATLAS 検出器のエンドキャップに設置されるミューオン検出器、及びそれを用いたミューオントリガーシステムの開発に関して述べる。TGC からの出力チャンネル数はシステム全体で 30 万以上になり、これらからの信号の処理を迅速に行なうため、信号処理系の IC 化が必須となる。

そこで特に、信号のタイミング調整のために用いられる Patch Panel ASIC、またその次段の運動量の判定を行なう Slave Board ASIC、High-pT ASIC に関しての設計、開発、また試作 IC のテストに関して述べていく。さらに放射線環境下での IC の安定した動作を確認するための IC と放射線照射試験に関しても述べる。

目 次

第 1 章　LHC と ATLAS 実験の概要 **10**

1.1 エネルギーフロンティア物理 10
1.2 LHC 10
1.3 ATLAS 実験の物理 12
1.3.1 ヒッグス粒子 12
1.3.2 超対称性粒子 13

第 2 章　ATLAS 検出器 **17**

2.1 概要 17
2.2 磁石系 19
2.3 内部飛跡検出器 19
2.4 カロリーメーター 19
2.5 ミューオン検出器 19
2.6 トリガーとデータ収集 24
2.6.1 トリガーの方法 24
2.6.2 データ収集システム 25
2.6.3 検出器の制御 25
2.7 レベル 1 トリガーシステム 26
2.7.1 ミューオントリガーシステム 26
2.7.2 CTP(Central Trigger Processor) 26
2.7.3 TTC(Timing,Trigger and control distribution) 26

第 3 章　TGC ミューオントリガーシステム **28**

3.1 TGC のレイアウト 28
3.2 TGC の構造と特性 30
3.3 トリガー処理の方法 32
3.4 トリガー用エレクトロニクス 34
3.4.1 概要 34
3.4.2 ASD 34
3.4.3 PS-Pack 36
3.4.4 PS-board 37
3.4.5 High-pT Board 38
3.4.6 Secter Logic 38
3.4.7 MUCTPI(Muon Trigger Interface to CTP) 38
3.4.8 タイミング調整 40
3.4.9 放射線の影響 42

第 4 章　ASIC 開発 **45**
4.1 CMOS プロセス . . . 45
4.2 回路の表現 . . . 46
4.3 開発の流れ . . . 48

第 5 章　JTAG プロトコル **50**
5.1 JTAG プロトコルの仕組み . . . 50
5.1.1 概要 . . . 50
5.1.2 JTAG の制御線 . . . 50
5.1.3 バウンダリースキャンセル (BSC) . . . 52
5.1.4 データレジスタ . . . 52
5.1.5 バイパスレジスタ . . . 54
5.1.6 Tap Controler . . . 54
5.1.7 Instruction Decoder . . . 55
5.1.8 TDO selector . . . 56
5.2 多数決論理回路 . . . 58
5.3 JTAG の HDL コードを自動生成するスクリプト . . . 59
5.4 JTAG プロトコルに関するまとめ . . . 59

第 6 章　Patch Panel ASIC の開発 **60**
6.1 Patch Panel ASIC の機能と構成 . . . 60
6.2 Patch Panel ASIC の設計 . . . 62
6.2.1 LVDS レシーバー . . . 62
6.2.2 遅延回路 (DLL) . . . 65
6.2.3 BCID 回路 . . . 68
6.2.4 テストパルス回路 . . . 70
6.2.5 Patch Panel における JTAG . . . 72
6.2.6 Patch Panel ASIC の設計におけるデザインの検証 . . . 73
6.2.7 Patch Panel ASIC 全体のレイアウト . . . 76
6.3 Patch Panel ASIC 試作 IC の動作検証 . . . 78
6.3.1 LVDS レシーバーの動作検証 . . . 80
6.3.2 遅延回路 (DLL) の動作検証 . . . 82
6.3.3 テストパルス回路の動作検証 . . . 84
6.3.4 JTAG プロトコルの動作検証 . . . 85
6.4 Patch Panel ASIC に関するまとめ . . . 86

第 7 章　Slave Board ASIC,
High-pT ASIC の開発 **87**
7.1 Slave board ASIC,High-pT ASIC の機能と構成 . . . 87
7.1.1 Doublet Slave Board(DSB) . . . 88
7.1.2 Triplet Slave Board . . . 88
7.1.3 EI/FI Slave Board(EFSB) . . . 89
7.1.4 Slave Board ASIC に共通の機能 . . . 94
7.1.5 High-pT ASIC . . . 95
7.2 Slave Board ASIC の設計 . . . 98

7.2.1 概要 . . . 98
7.2.2 入力部 . . . 99
7.2.3 制御部 . . . 101
7.2.4 マトリックス部 . . . 103
7.2.5 読み出し部 . . . 104
7.2.6 シミュレーションによるデザインの検証 . . . 104
7.2.7 ASIC 全体のレイアウト . . . 105
7.3 High-pT ASIC の設計 . . . 106
7.3.1 制御部分 . . . 107
7.3.2 シミュレーションによる動作検証 . . . 107
7.3.3 High-pT ASIC のレイアウト . . . 108
7.4 Slave Board ASIC, High-pT ASIC に関するまとめ . . . 109

第 8 章 放射線耐性試験用 IC と放射線照射試験 110
8.1 概要 . . . 110
8.2 放射線試験用 IC . . . 112
8.2.1 単体 CMOS . . . 112
8.2.2 リングオシレーター . . . 114
8.2.3 照射試験用ボード . . . 115
8.3 放射線照射試験 . . . 116
8.3.1 γ 線照射時のセットアップ . . . 116
8.3.2 測定系のセットアップ . . . 119
8.3.3 測定結果 . . . 119
8.4 放射線耐性試験用 IC と放射線照射試験に関するまとめ . . . 122

第 9 章 全体のまとめ 123

付 録 A JTAG プロトコルに関する補足 125
A.1 JTAG プロトコルの動作例 . . . 125
A.1.1 データレジスタの読み書き . . . 127
A.1.2 BSC(SAMPLE/PRELOAD,EXTEST) . . . 129
A.2 JTAG のサンプルコード . . . 130

付 録 B Patch Panel ASIC に関する補足 138
B.1 Patch Panel ASIC におけるユーザー定義レジスタの機能 . . . 138
B.2 Patch Panel ASIC における JTAG の動作の検証 . . . 140

付 録 C Slave Board ASIC,High-pT ASIC に関する補足 143
C.1 Slave Board ASIC におけるユーザー定義レジスタの機能 . . . 143
C.2 Slave Board ASIC に関する JTAG の動作の検証 . . . 145
C.3 High-pT ASIC におけるユーザー定義レジスタの機能 . . . 151
C.4 High-pT ASIC における JTAG の動作の検証 . . . 151
参考文献 . . . 156
謝辞 . . . 157

図 目 次

1.1 LHC . . . 11
1.2 LHC の断面 . . . 11
1.3 ATLAS . . . 11
1.4 CMS . . . 11
1.5 ALICE . . . 11
1.6 LHCB . . . 11
1.7 標準 Higgs 粒子の質量と生成断面積関係 . . . 15
1.8 グルーオン融合 . . . 15
1.9 W,Z 融合 . . . 15
1.10 W,Z を伴う higgs 生成 . . . 15
1.11 t 生成を伴う場合 . . . 15
1.12 t 生成を伴う場合 . . . 15
1.13 Higgs 粒子の崩壊分岐比と質量の関係 . . . 16
1.14 重いクォークへの崩壊 . . . 16
1.15 4 つのレプトンへの崩壊 . . . 16
1.16 レプトンとニュートリノへの崩壊 . . . 16
1.17 2 つのフォトンへの崩壊 . . . 16

2.1 ATLAS 検出器の全体図 . . . 18
2.2 磁石系のコイル部 . . . 20
2.3 積分磁場強度 (η 平面) . . . 20
2.4 エンドキャップ部の磁束の構造 (XY 平面) . . . 20
2.5 内部検出器 . . . 21
2.6 内部検出器の XY 断面 . . . 21
2.7 内部検出器の RZ 断面 . . . 21
2.8 カロリーメーターの構造 . . . 22
2.9 アコーディオン型電磁カロリーメーターの構造 . . . 22
2.10 タイル型ハドロンカロリーメーターの構造 . . . 22
2.11 ミューオン検出器 . . . 23
2.12 ミューオン検出器の XY 断面 . . . 23
2.13 ミューオン検出器の RZ 断面 . . . 23
2.14 トリガー処理の流れ . . . 24
2.15 LVL1 トリガー処理の流れ . . . 26
2.16 LVL1 ミューオンシステム . . . 27

3.1 TGC のレイアウト (RZ 断面) . . . 29
3.2 TGC のレイアウト (XY 断面) . . . 29

3.3 TGC の構造 . . . 30
3.4 TGC のタイムジッター . . . 31
3.5 TGC オクタントの分割 . . . 32
3.6 TGC のワイヤ方向の分割 . . . 32
3.7 p_T の判定の仕方 . . . 33
3.8 TGC トリガーエレクトロニクスの流れ . . . 35
3.9 ps-pack の配置 . . . 36
3.10 PS-board の構成 . . . 37
3.11 TGC トリガーシステムの配置 . . . 39
3.12 Sector Logic のブロック図 . . . 39
3.13 Sector Logic 上の処理のながれ . . . 39
3.14 ASD から Slave Board ASIC までの信号のタイミングチャート . . . 40
3.15 クロックと信号のエッジが重なる場合のタイミングチャート . . . 41

4.1 CMOS の断面図 . . . 45
4.2 回路の表現 . . . 47
4.3 ASIC 開発の流れ . . . 48

5.1 JTAG プロトコルの概要 . . . 51
5.2 boundary scan cell . . . 52
5.3 データレジスタ RW モードユーザーレジスタ、インストラクションレジスタ) . . . 53
5.4 データレジスタ (R モードユーザーレジスタ) . . . 53
5.5 tap_state の遷移図 . . . 54
5.6 EXTEST を使った相互接続テスト . . . 56
5.7 TDO セレクタの回路図の例 . . . 57
5.8 多数決論理回路の回路図 . . . 58

6.1 Patch Panel ASIC の機能の概要 . . . 60
6.2 LVDS の信号レベル . . . 62
6.3 LVDS レシーバーの回路図 . . . 64
6.4 LVDS レシーバーのレイアウト . . . 64
6.5 LVDS レシーバーの SPICE シュミレーションの結果 . . . 64
6.6 DLL 回路の概要 . . . 65
6.7 位相検出器の動作 . . . 66
6.8 DELAY CELL の回路図 . . . 67
6.9 位相検出器の回路図 . . . 67
6.10 チャージポンプの回路図 . . . 67
6.11 ゲートを拡張した場合のタイミング . . . 68
6.12 BCID 回路の回路図 . . . 69
6.13 BCID 回路の HDL シュミレーションの結果 . . . 69
6.14 テストパルス回路の回路図 . . . 71
6.15 テストパルス回路のレイアウト . . . 71
6.16 テストパルス回路の SPICE シュミレーションの結果 . . . 71
6.17 自動配置配線によるレイアウ . . . 73
6.18 BCID 回路のレイアウト . . . 74

6.19 BCID 回路の回路図 . . . 74
6.20 JTAG のテスト波形 tap state の遷移 . . . 75
6.21 JTAG のテスト波形 EXTEST . . . 75
6.22 Patch Panel ASIC ver0 のレイアウト . . . 76
6.23 Patch Panel ASIC のブロック図 . . . 77
6.24 DLL 回路テストチップのレイアウト . . . 78
6.25 Patch Panel ASIC アナログ回路テスト ASIC . . . 79
6.26 Patch Panel ASIC(version (-1)) . . . 79
6.27 LVDS レシーバーの計測のセットアップ . . . 80
6.28 LVDS レシーバー オフセット電圧に対する出力信号の遅延 . . . 81
6.29 LVDS レシーバー オフセットと振幅の定義 . . . 81
6.30 DLL 回路の特性を測定するためのセットアップ . . . 82
6.31 制御電圧 VCON に対する遅延時間 . . . 83
6.32 テストパルス回路の計測のセットアップ . . . 84
6.33 テストパルス回路 閉じた CMOS の個数に対する電圧の値 . . . 84
6.34 JTAG の動作確認のセットアップ . . . 85

7.1 Doublet Slave Board のブロック図 . . . 90
7.2 Doublet Slave Board の $low - p_T matrix$ の構造 . . . 90
7.3 Doublet Slave Board の $low - p_T matrix$ の詳細な構造 . . . 90
7.4 Triplet Slave Board(wire) のブロック図 . . . 91
7.5 Triplet Slave Board(wire) の論理 . . . 91
7.6 Triplet Slave Board(strip) のブロック図 . . . 92
7.7 Triplet Slave Board(strip) の論理 . . . 92
7.8 Triplet Slave Board(EI/FI) のブロック図 . . . 93
7.9 Triplet Slave Board(EI/FI) の論理 . . . 93
7.10 デクラスタリングのアルゴリズム . . . 94
7.11 High-pT ASIC のブロック図 (wire) . . . 96
7.12 High-pT ASIC のブロック図 (strip) . . . 96
7.13 Doublet Slave Board のブロック図 . . . 97
7.14 Doublet Slave Board の $low - p_T matrix$ の詳細な構造 . . . 97
7.15 Slave Board ASIC のブロック図 . . . 98
7.16 プロセスによるシフトレジスタの動作の違い . . . 99
7.17 マスク部 1 チャンネル分の回路図 . . . 100
7.18 位相補正回路 . . . 101
7.19 位相補正回路のタイミングチャート . . . 101
7.20 TTC 信号の受信用回路 . . . 102
7.21 テストパルス信号の受信回路 . . . 103
7.22 バッファーと負荷による動作速度の違い . . . 103
7.23 Slave Board ASIC でのクロックの分配の仕方 . . . 103
7.24 Slave Board ASIC のレイアウト . . . 105
7.25 High-pT ASIC のブロック図 . . . 106
7.26 high-pT ASIC のレイアウト (日立ゲートアレイ 0.35μm) . . . 108

8.1 γ 線が半導体に影響を与える仕組み . . . 110

8.2 放射線耐性試験用 IC のレイアウト . . . 111
8.3 放射線耐性試験用 IC のブロック図 . . . 111
8.4 単体 CMOS のレイアウト . . . 113
8.5 電源パット . . . 113
8.6 リングオシレーターの回路図 . . . 114
8.7 リングオシレーターのレイアウトの一部 . . . 114
8.8 照射試験用のボード . . . 115
8.9 照射時のセットアップ (写真) . . . 117
8.10 照射時のセットアップ . . . 117
8.11 照射時のセットアップ (回路図) . . . 118
8.12 テストボード運搬時の配線 . . . 118
8.13 半導体パラメータアナライザーを使った測定系のセットアップ . . . 119
8.14 γ 線照射量に対するリングオシレーターの周波数 . . . 120
8.15 PMOS の動作特性 . . . 121
8.16 PMOS の動作特性 . . . 121
8.17 γ 線照射量に対するリーク電流の変化 . . . 121

A.1 サンプルの構成 . . . 125
A.2 JTAG プロトコルの動作波形 (データレジスタの読み書き) . . . 126
A.3 ユーザーレジスタの読み書き . . . 127
A.4 JTAG プロトコルの動作波形 (EXTEST、SAMPLE/PRELOAD) . . . 128
A.5 ユーザーレジスタの読み書き . . . 129

B.1 JTAG のテスト波形 tap state の遷移 . . . 141
B.2 JTAG のテスト波形 インストラクション . . . 141
B.3 JTAG のテスト波形 EXTEST . . . 141
B.4 JTAG のテスト波形 データレジスタ FUSE . . . 142
B.5 JTAG のテスト波形 RW モードデータレジスタ . . . 142
B.6 JTAG のテスト波形 リセット . . . 142

C.1 JTAG プロトコルのテスト波形、EXTEST . . . 147
C.2 JTAG プロトコルのテスト波形、INTEST . . . 147
C.3 JTAG プロトコルのテストテスト波形、データレジスタ bypass . . . 148
C.4 JTAG プロトコルのテストテスト波形、データレジスタ clkinv . . . 148
C.5 JTAG プロトコルのテストテスト波形、データレジスタ delay . . . 149
C.6 JTAG プロトコルのテストテスト波形、データレジスタ L1Aveto . . . 149
C.7 JTAG プロトコルのテストテスト波形、データレジスタ MASK1P . . . 150
C.8 JTAG プロトコルのテストテスト波形、データレジスタ OVERFLOW . . . 150
C.9 JTAG プロトコルの波形、データレジスタ DELAY0 . . . 153
C.10 JTAG プロトコルの波形、データレジ スタ DELAY1 . . . 153
C.11 JTAG プロトコルの波形、データレジスタ DELAY2 . . . 154
C.12 JTAG プロトコルの波形、データレジ スタ DELAY4 . . . 154
C.13 JTAG プロトコルの波形、データレジスタ DELAY5 . . . 155
C.14 JTAG プロトコルの波形、データレジ スタ GLINK . . . 155

表 目 次

1.1 LHC の主なパラメータ 10

3.1 TGC の主なパラメータ 31
3.2 TGC 読みだしチャンネル数 32
3.3 放射線量の見積もりと安全係数 42
3.4 TGC における中性子量と SEU の頻度 43

5.1 JTAG 命令 55

6.1 ユーザー定義レジスタの一覧 72

7.1 Slave Board ASIC の種類 87
7.2 high-pT ASIC 機能の種類 87
7.3 ユーザー定義レジスタの一覧 102
7.4 ユーザー定義レジスタの一覧 107

第1章　LHCとATLAS実験の概要

1.1　エネルギーフロンティア物理

現代物理学の最も大きな成功の一つに標準理論が挙げられる。標準理論は強い相互作用と電弱相互作用を記述する理論で、20 世紀の後半に多くの理論的実験的な検証がなされ、それは 100GeV までのエネルギースケールにおいて、すばらしい精度で実験結果に一致することがわかった。しかしながら標準理論の要となるヒッグスボソンは、未だに発見されていないため、標準理論の検証も完全ではない。ヒッグスボソンは素粒子の質量を説明するためのもので、もし存在するのであれば、その質量は理論的に 1TeV 以下であるとされている。さらに超対称性理論などの標準理論を越える物理のためにも、TeV 領域の素粒子の探索は、素粒子物理学の為に不可欠である。
欧州原子核研究機構 (CERN) に建設が予定されている大型陽子陽子衝突型加速器 (LHC) は、この TeV 領域の探索が可能な最初の加速器で、エネルギーフロンティアのための最も強力な手段になるものである。

1.2　LHC

LHC は超電導磁石を用いた陽子陽子の衝突型加速器で、14TeV の重心系衝突エネルギーを達成するものである (図 1.1、図 1.2)。この加速器の主なパラメータを表 1.1 に示す。

パラメータ	
重心系エネルギー (陽子陽子)	7TeV+7TeV
輝度 (低輝度時)	$10^{33} cm^{-2} s^{-1}$
輝度 (高輝度時)	$10^{34} cm^{-2} s^{-1}$
バンチ衝突頻度	40.08MHz
バンチ衝突間隔	24.95 nsec
トンネル周長	26.66km

表 1.1: LHC の主なパラメータ

LHC には 4 つのバンチ衝突点があり、それぞれに ATLAS(A Troidal Lhc ApparatuS) (図 1.3)、CMS(The Compact Muon Solenoid) (図 1.4)、ALICE(A Large Ion Collider Experiment) (図 1.5)、LHC-B (図 1.6) が設置される。これらのうち、この論文では ATLAS 検出器について述べていく。ATLAS 検出器はヒッグスボソン探索、超対称性粒子探索を主な目的として、その他にも B 崩壊を使った CP 非保存や、W やトップクォークの精密測定、新しいゲージボソンの探索、クォークやレプトンの構成粒子の探索、さらには全く新しい物理の発見も期待されている。

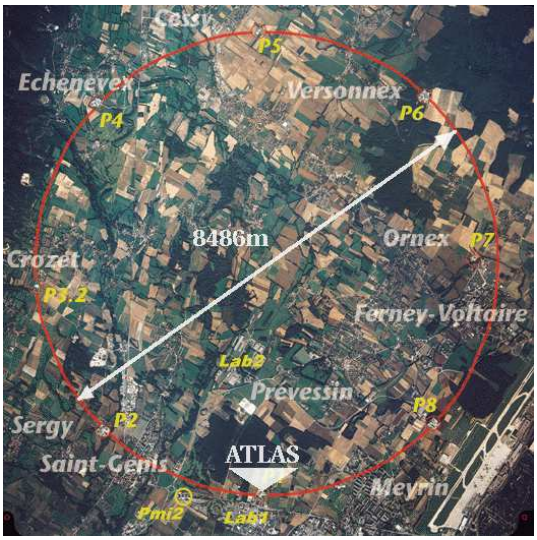


図 1.1: LHC

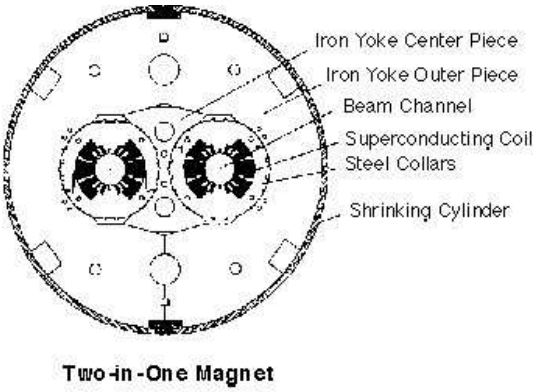


図 1.2: LHC の断面 (磁石部)

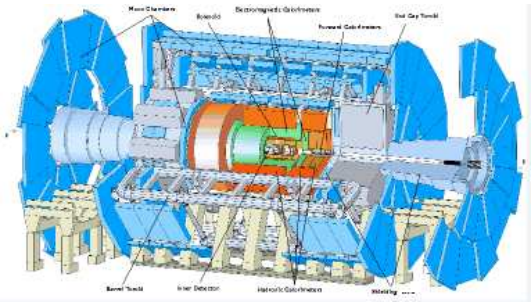


図 1.3: ATLAS

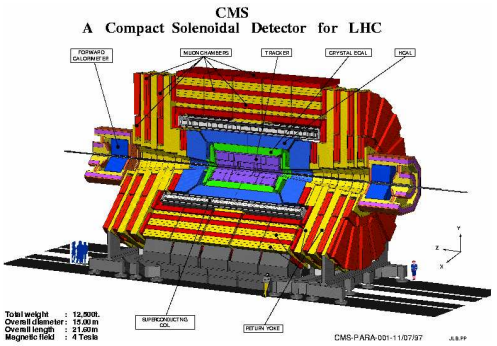


図 1.4: CMS

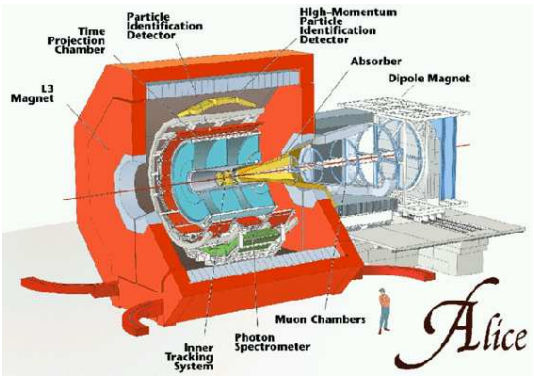


図 1.5: ALICE

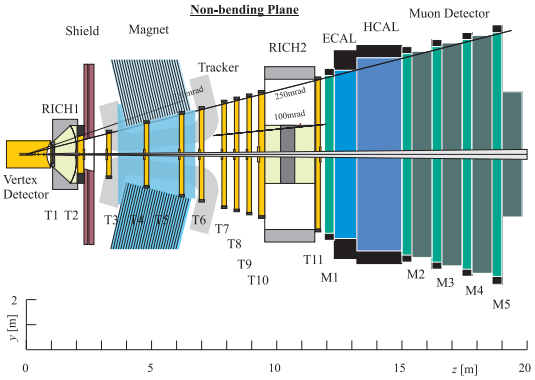


図 1.6: LHCB

1.3 ATLAS 実験の物理

ここでは ATLAS 実験の最も大きなテーマであるヒッグス粒子と超対称性粒子の物理に関して概要を述べる。

1.3.1 ヒッグス粒子

ヒッグス粒子の探索は電弱対称性の破れる機構を理解するために不可欠で、標準理論においてはヒッグス場としてスカラー場 2 重項を仮定しており、電荷中性のスカラーのヒッグス粒子が 1 つ存在することが導かれている。また超対称性理論のうちの最も簡単なモデルである MSSM(Minimal Supersymmetric Standard Model) では、少なくとも 2 つのヒッグス 2 重項が必要でそこからは、2 つの荷電ヒッグス ($H^{\pm}$) と 3 つの電荷中性ヒッグス (h,H,A) が導かれる。

標準理論ヒッグス粒子

図 1.7 に LHC における標準理論 Higgs 粒子の質量と生成断面積の関係を、図 1.8 から 1.12 に主な生成過程のファインマン図を示す。標準理論ヒッグスの生成は主に重いクォークを介するグルーオン融合によりなされる。図 1.13 に標準 Higgs 粒子の質量と Higgs 粒子の崩壊モードの分岐比の関係を、図 1.14 から 1.16 に主な崩壊モードのファインマン図を示す。

- $H \to \gamma\gamma$、($100\mathrm{Gev} \leq m_H \leq 150\mathrm{GeV}$)
  この崩壊モードは希崩壊モードで、バックグランドは $(q\bar{q} \to \gamma\gamma)$,$(gg \to \gamma\gamma)$ などがあり、バックグランド上の狭いピークを測定する必要がある。そのため電磁カロリーメーターにはすぐれたエネルギーと角度の分解能が求められる。

- $H \to b\bar{b}$、($m_H \leq 2m_W$)
  この崩壊モードは $m_H \leq 2m_W$ の領域で分岐比の 90%を占めるので重要なモードである。しかしグルーオン融合による生成から来る場合 $(gg \to H \to b\bar{b})$ は、QCD2jet からのバックグランドの為に有効なトリガーができない。この崩壊モードの場合は生成時にレプトン崩壊をする W,Z または $t\bar{t}$ ペアを含む場合が有効である。

- $H \to ZZ^* \to 4l$、($130\mathrm{Gev} \leq m_H \leq 2m_Z$)
  4 つのレプトンのエネルギーを測定することにより、きれいな信号が得られると期待される。1 対のレプトンの不変質量が m_z に等しくなるという条件を課し、バック・グラウンドを落すことができる。

- $H \to ZZ \to 4l$、($2m_Z \leq m_H \leq 800\mathrm{GeV}$)
  この領域は LHC の最も得意とする領域である。それぞれのレプトン対の不変質量に条件付けができるため、信頼性の高いモードである。ただし Higgs 粒子の質量が大きくなるにつれて、崩壊幅が急激に大きくなるため、Higgs 粒子の質量があまり大きくなると有効性が落ちる。

- $H \to WW \to ll\nu\nu$、($500\mathrm{GeV} \leq m_H \leq 700\mathrm{GeV}$)
  Higgs 粒子の質量が 600GeV 以上の領域では、$H \to 4l$ よりも約 6 倍レートが高い。ν に起因する消失横方向エネルギー E_T^{miss} の[1]測定を行なうことになる。

[1]バンチ衝突により生成された粒子の横方向運動量 (横方向はビーム軸に垂直な方向) の和は 0 になるが、ニュート

超対称性ヒッグス粒子

超対称性理論の中で、最も単純な超対称性標準模型 (Minimal Supersymmetric Standard Model) では、2 つの Higgs 二重項が必要とされ、その結果 5 つの Higgs 粒子 (h, H, $H^{\pm}$, A) が存在する。MSSM で特徴的な崩壊モードに関して説明する。

- $H/A \to \tau\tau$
  標準 Higgs 粒子の場合は、$H \to \tau\tau$ モードはレートが低く測定に適さないが、MSSM では高いレートが期待される。生成された τ 粒子の両方がレプトンに崩壊するチャンネルと、一方はハドロンに崩壊するチャンネルの 2 種類のモードが利用できる。

- $H/A \to \mu\mu$
  上に比べて、分岐比は $(m_\mu/m_\tau)^2$ 倍低いが、精度よく測定が行なえることから、$\tau\tau$ モードでの測定を補う役割が期待される。

- $H \to hh$
  崩壊モードは、$hh \to bb\bar{b}\bar{b}$ が支配的だが、このモードでは効率の良いトリガーが行なえないため、$hh \to \gamma\gamma b\bar{b}$ チャンネルで観測されることが期待される。イベント・レートは低いが、2 つの異なる Higgs の反応という意味で非常に興味深い。

- $A \to Zh$
  2 つの Higgs が関係した反応として、興味深い。$Zh \to llb\bar{b}$ など Z の崩壊で生じる 2 つのレプトンでトリガーを行なう方法が有効である。

1.3.2 超対称性粒子

LHC では、強い相互作用をするスクォークやグルイーノの対が大量に生成されるなど、超対称性粒子の探索が期待されている。R-parity 保存則から、超対称性粒子は必ず対で生成され、最終的に超対称性粒子の中で一番軽い質量を持つ LSP (Lightest Supersymmetric Particle) に崩壊する。LSP の候補としては、最軽量ニュートラリーノが考えられる。この粒子は検出されず、イベントの解析において消失横方向エネルギー E_T^{miss} として現れることになるので、終状態の多数のジェットとともに、この E_T^{miss} を指標として探索を行なう。主な崩壊モードとしては、

- Multi-jets + E_T^{miss} モード
  $(\tilde{g} \to \tilde{q}q + \tilde{\chi}_1^0 \to jets + E_T^{miss})$、$(\tilde{g} \to q\tilde{\chi}_1^0 \to jets + E_T^{miss})$ など

- 同符合の 2 レプトンモード
  $2\tilde{g} \to 2(q\bar{q}\tilde{\chi}_i^{\pm}) \to 2(q\bar{q}W^{\pm}\tilde{\chi}_i^0) \to 2(jets + l^{\pm} + E_T^{miss})$ など

- 3 レプトンモード
  $\tilde{\chi}_1^{\pm}\tilde{\chi}_2^0 \to l\nu\tilde{\chi}_1^0 + ll\tilde{\chi}_1^0 \to 3l + E_T^{miss}$など

がある。

以上のようにヒッグス粒子、超対称性粒子の崩壊過程には、レプトンを含む過程が多数存在する。なかでもミューオンは粒子識別が比較的容易であるので、高い精度で確実にとらえることが

リノなど、検出器でとらえにくい粒子が、大きな運動量を持って生成された場合、検出器上では横方向運動量が 0 でないように見える。そこで検出器で捉えた粒子のエネルギーベクトルの和の逆ベクトルを、消失エネルギーとして定義し、検出器で捉えにくい粒子の間接的な測定に使う。消失横方向エネルギーはその横方向成分。

できる。分解能としては 2 つまたは 4 つのミューオンから、1%程度の精度で質量、運動量を再構成することが求められている。また横方向運動量 p_T を使った閾値が 10∼20GeV のトリガーの判定が求めらる。

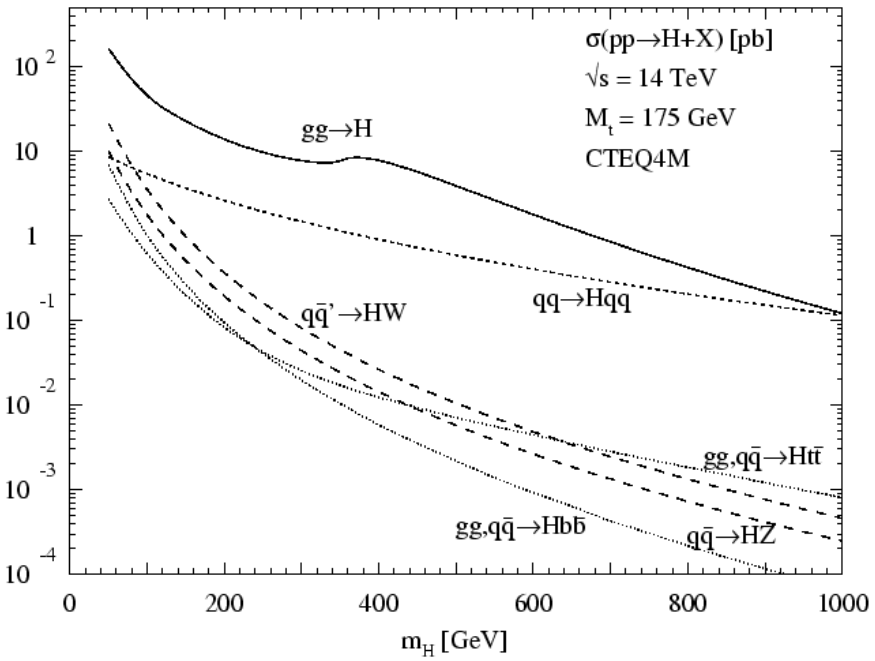


図 1.7: LHC における標準 Higgs 粒子の質量と生成断面積の関係。(トップ・クォークの質量=175GeV として計算)

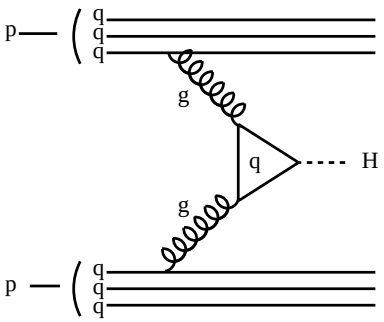


図 1.8: グルーオン融合

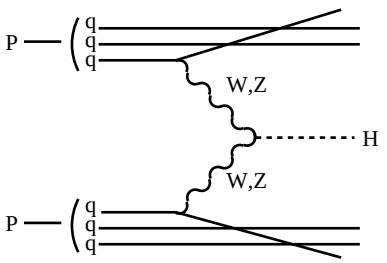


図 1.9: W,Z 融合

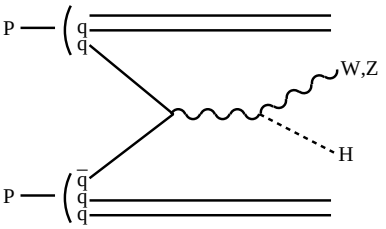


図 1.10: W,Z を伴う higgs 生成

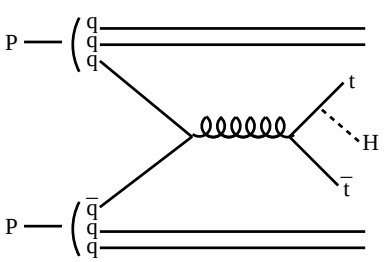


図 1.11: t 生成を伴う場合

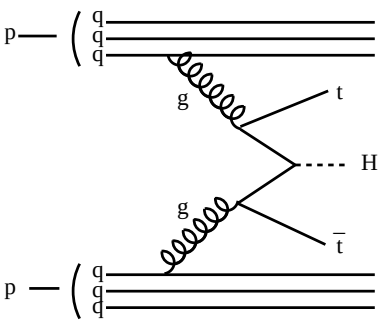


図 1.12: t 生成を伴う場合

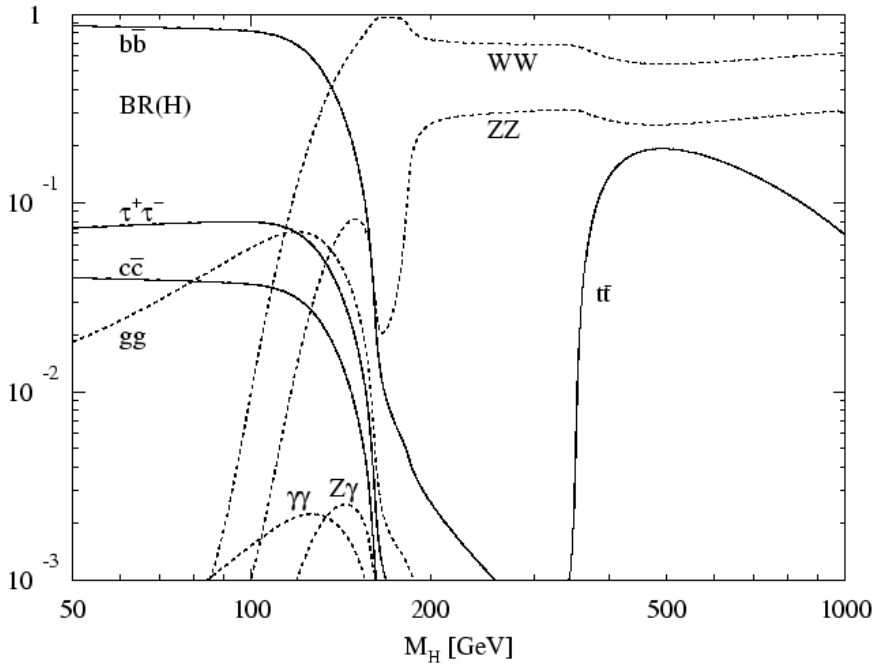


図 1.13: Higgs 粒子の崩壊分岐比と質量の関係 (トップ・クォークの質量=175GeV として計算)

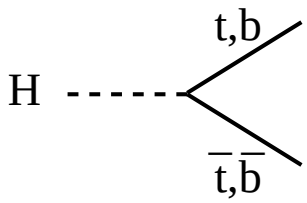


図 1.14: 重いクォークへの崩壊

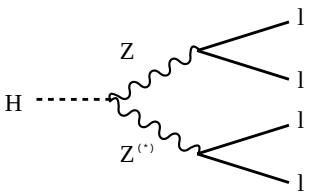


図 1.15: 4 つのレプトンへの崩壊

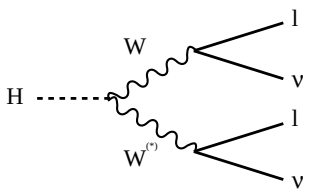


図 1.16: レプトンとニュートリノへの崩壊

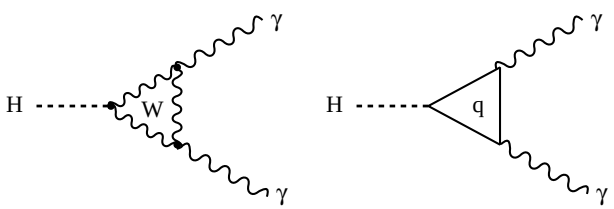


図 1.17: 2 つのフォトンへの崩壊

第2章　ATLAS検出器

ATLAS 測定器 (図 2.1) は、LHC の高いルミノシティにおいても、電子、光子、ハドロン、τ 粒子、ミューオン、ジェット、消失横方向エネルギー E_T^{miss} などの信号を確実に得ることができるように設計されており、LHC で起こる様々な物理現象を的確に捕らえることができると期待されている。ATLAS 検出器の大きさは直径 22m、長さ 44m、総重量 7000 トンで主に、内部飛跡検出器、カロリメータ、ミューオン検出器から成る。ここでは、各検出器の特徴について説明する。

2.1　概要

ATLAS 検出器は、LHC 実験に用いられる汎用の陽子陽子反応用の検出器で、その大きさは直径 22m、長さ 44m、総重量 7000 トンと巨大なもので、以下に示すような要求を全て満たすものである。

- 電磁カロリーメーターによる高精度の電子とフォトンの測定が可能である。ハドロンカロリーメーターによる高精度なジェットと消失横方向エネルギー (E_T^{miss}) の測定が可能である。
- ミューオンスペクトロメーターによる高精度のミューオンの運動量測定が可能である。
- 飛跡検出による high-pT レプトン運動量の測定と電子とフォトンの識別と完全なイベント再構成が可能である。
- 大きなラピディティ η と[1] 完全な方位角をカバーする。
- 高頻度でやってくる電子、光子、ミューオン、ジェットなどをパイルアップすることなく処理する。
- 大量のバックグランド粒子に対する耐放射線性を持つ。

図 2.1 に ATLAS の各検出器の配置を示す。磁石系は内部検出器を囲む超電導ソレノイド磁石、またカロリーメーターの外部を囲む 8 回対称に配置された超伝導中空トロイダル磁石からなる。内部飛跡検出器は、2T の磁場をつくる長さ 7m 半径 1.15m のソレノイド磁石の内部に設置され、ピクセル検出器、ストリップ型検出器、及びストローチューブ検出器からなる。内部飛跡検出器の外側にはハドロンカロリーメータ、電磁カロリーメータが設置され、電子、光子、ジェットなどのエネルギーの測定を行なう。またトロイダル磁石の周りと中空コイルの中にはミューオン検出器が設置される。

[1] η は pseudo-rapidity と呼ばれ、θ をビーム軸との角度とすると η=−ln(tan(θ)/2) で定義される。

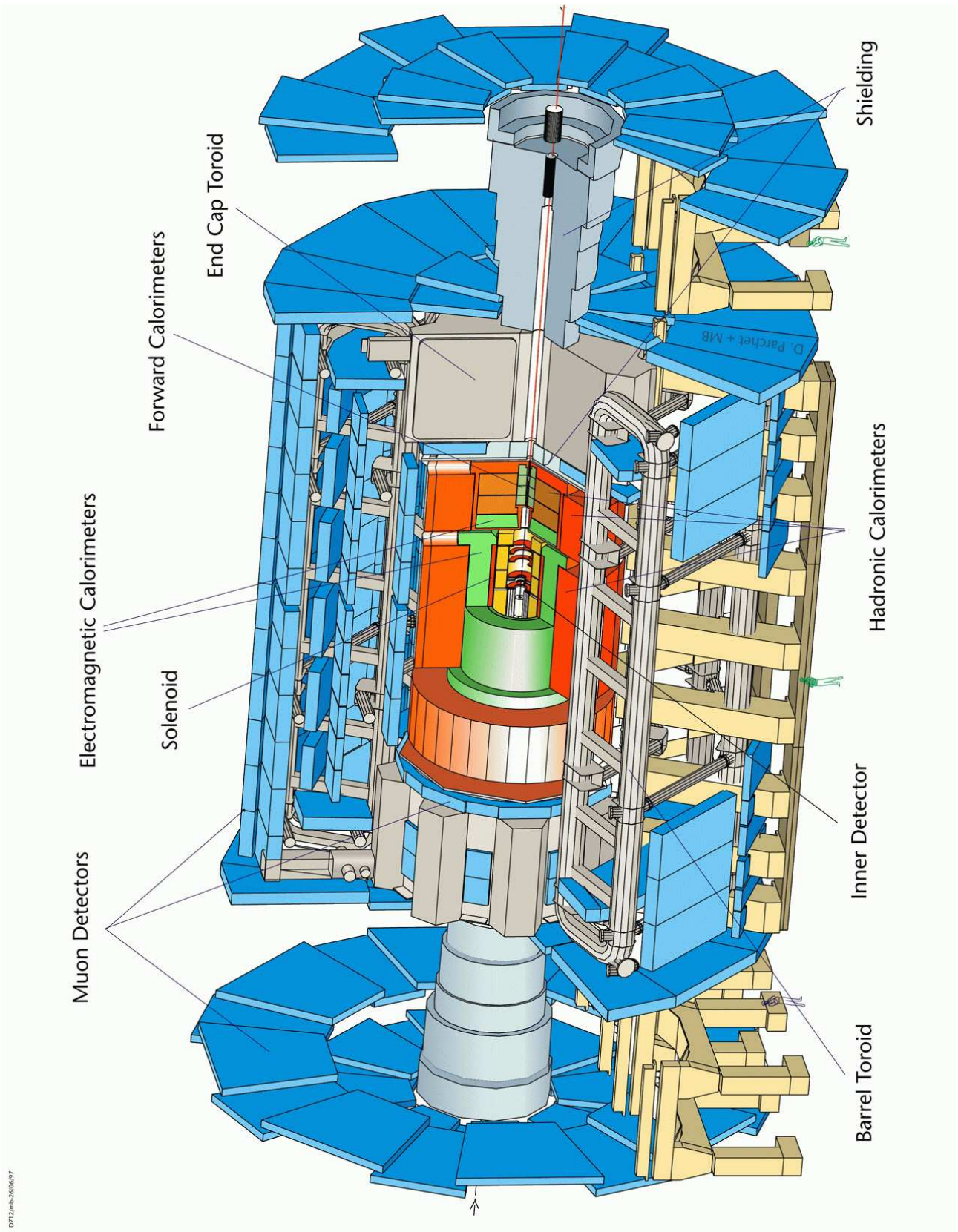


図 2.1: ATLAS 検出器の全体図

2.2 磁石系

ATLAS 検出器の磁石系は中央ソレノイド磁石 (Central Solenoid(CS))、エンドキャップ部トロイダル磁石 (end-cap toroids(ECT)), バレル部トロイダル磁石 (barrel toroid(BT)) からなり、いずれも超伝導磁石を使っている。CS は内部検出器のための磁場を作る。ECT 及び BT はミューオン検出器の為の磁場を作る。磁石系は全体で長さ 26m 直径 20m になる。

各磁石の磁場は CS が 2T、BT が 3.9T (ピーク値)、ECT が 4.1T (ピーク値) となる。磁石の粒子を曲げる能力を表す磁場を粒子の飛跡に沿って積分した値は BT で 2 から 6Tm （η:0∼1.3) ECT で 4 から 8Tm(η:1.6∼2.7) となる (図 2.3)。またトロイダル磁場は ϕ 方向成分のみではなく R 成分も持つ (図 2.4)。

2.3 内部飛跡検出器

内部検出器は中央ソレノイドの内部、2T の磁場中に設置され、高分解能の検出器により飛跡を検出する。図 2.5, 図 2.6, 図 2.7 に内部飛跡検出器を示す。シリコンマイクロストリップ検出器 (SCT) とピクセル型検出器 (Pixel) は半導体検出器で、高分解能、高効率の検出器で、最も反応点の近くに設置される。これらの検出器の層の数は、物質量の制限により数層に限られている。これらの半導体検出器の外側には、ストローチューブ型検出器 (TRT) が設置される。これは飛跡検出点あたりの物質量、およびコストが小さく出来るもので、 1 つの飛跡あたりの検出点が多く出来るものである。内部検出器ではこれらの検出器を組み合わせて高分解能の測定を行なう。

2.4 カロリーメーター

カロリーメーター (図 2.8) はラピディティー $|\eta|$ の広い範囲で、高い位置分解能、高いエネルギー分解能で粒子のエネルギーを測定するもので、電磁カロリーメーターと、ハドロンカロリーメーターからなる。内側には放射線耐久性にすぐれたアコーディオン型電磁アルゴン・カロリメータ (図 2.9) を設置する。外側のバレル部にはシンチレータと鉄の組合せのタイル型ハドロン・カロリメータ (図 2.10) を配置する。またエンドキャップ部、フォワード部には、液体アルゴンを使ったハドロン・カロリメータを設置する。

2.5 ミューオン検出器

ミューオン検出器を図 2.11, 図、2.12, 図 2.13 に示す。ミューオン検出器は超伝導中空トロイダル磁石と共に用いられ、トロイダル磁石の作る ϕ 方向の磁場によりミューオンがどれだけ曲げられるかを測定することにより、その運動量を求めるものである。ミューオン検出器は主に精密測定用の MDT(Monitored Drift Tube)、CSC(Cathode Strip Chamber) とトリガー判定用の RPC(Resistive Plate Chamber), 及び TGC(Thin Gap Chamber) から成る。

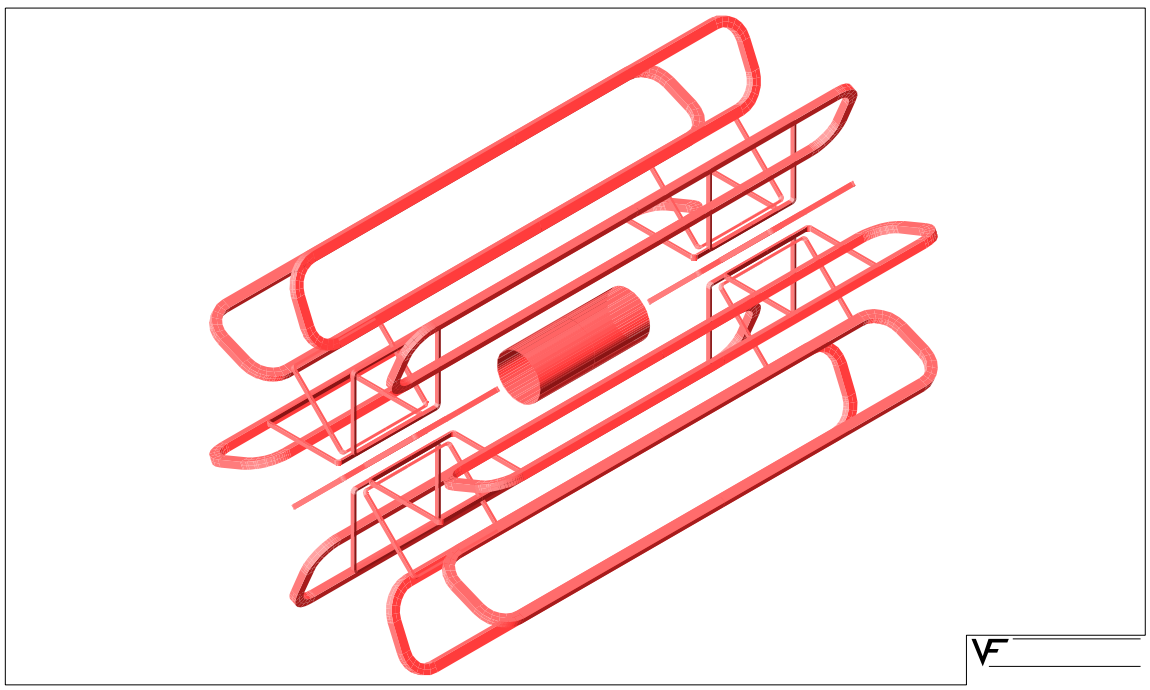


図 2.2: 磁石系のコイル部

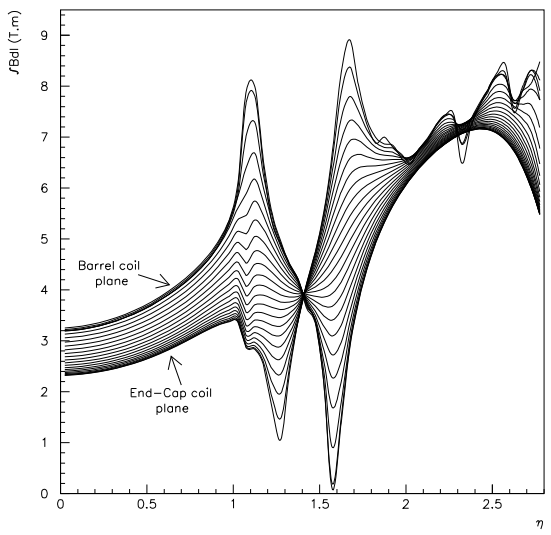


図 2.3: 積分磁場強度 (η 平面)
ECT から BT までの角度 (ϕ 方向) に関してプロットしたもの。

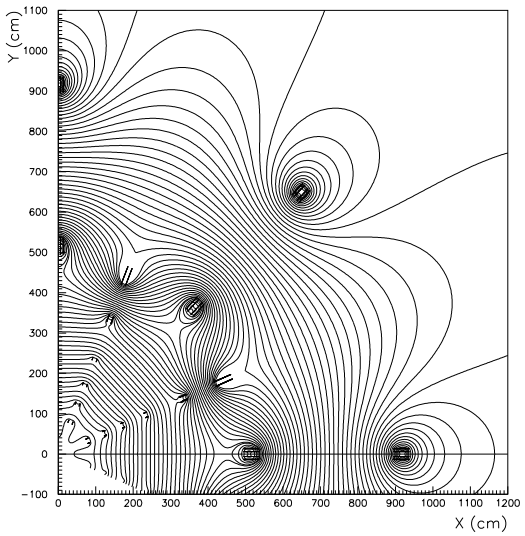


図 2.4: エンドキャップ部の磁束の構造 (XY 平面)(Z=10.5m:ECT の中心)

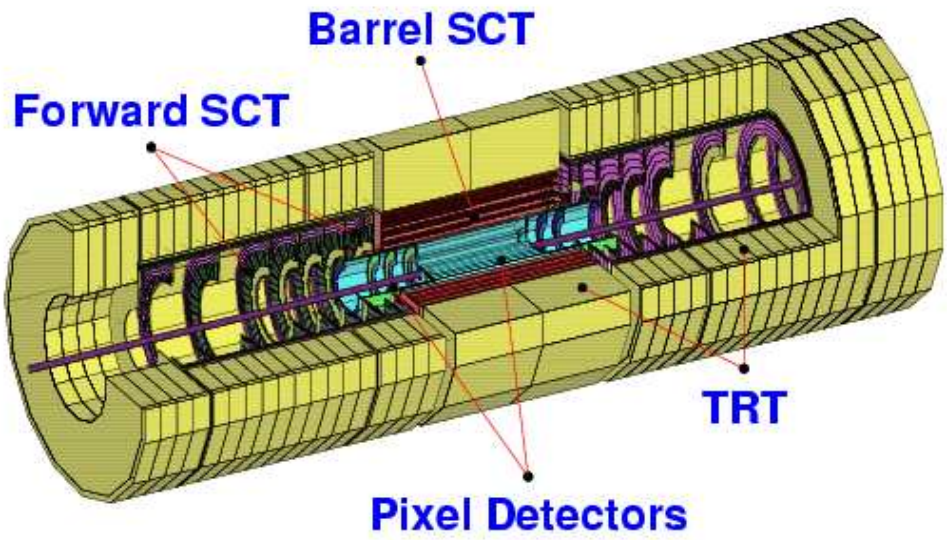


図 2.5: 内部検出器

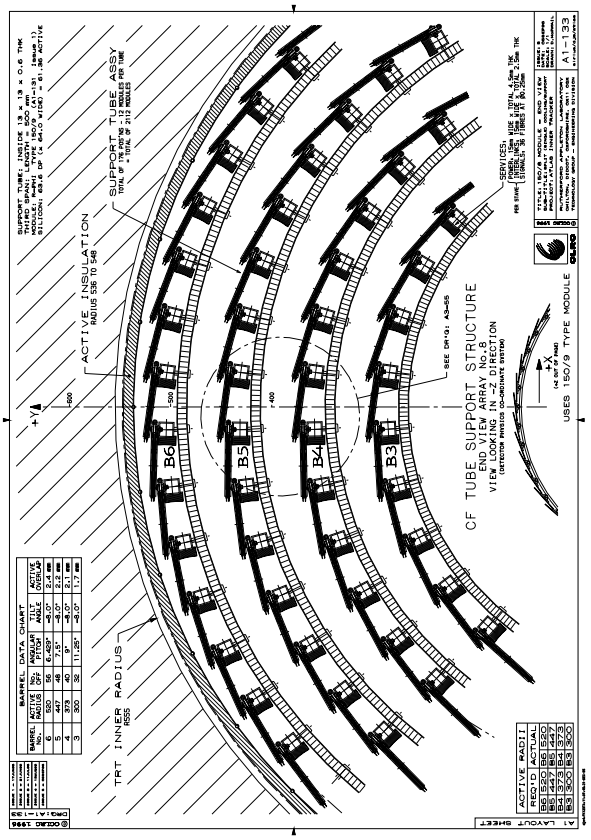


図 2.6: 内部検出器の XY 断面

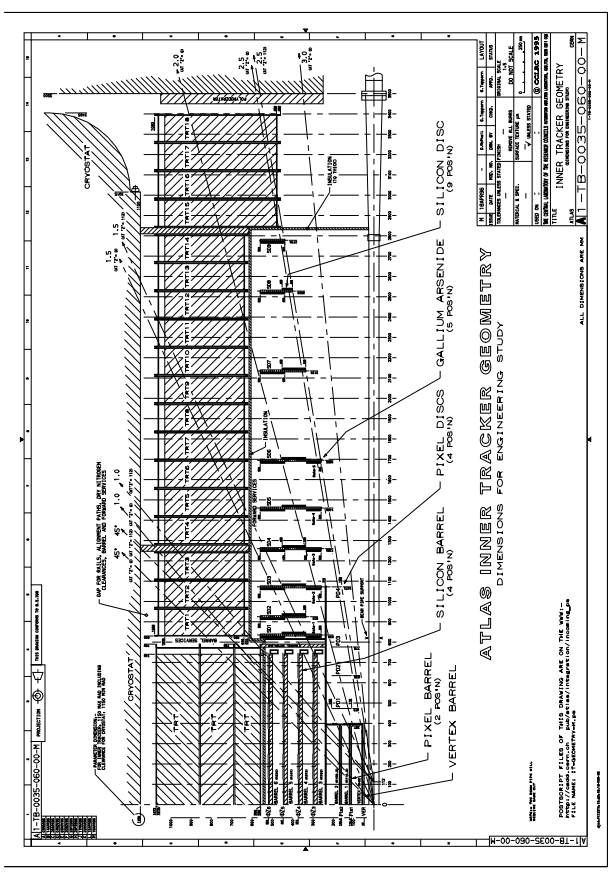


図 2.7: 内部検出器の RZ 断面

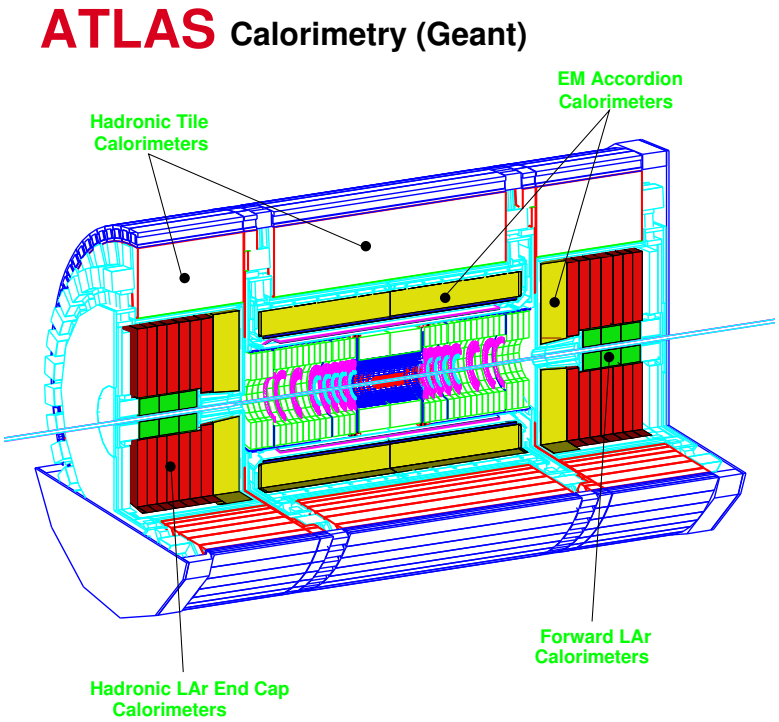


図 2.8: カロリーメーターの構造

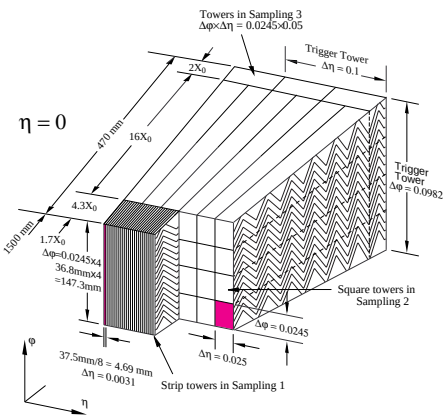


図 2.9: アコーディオン型電磁カロリーメーターの構造

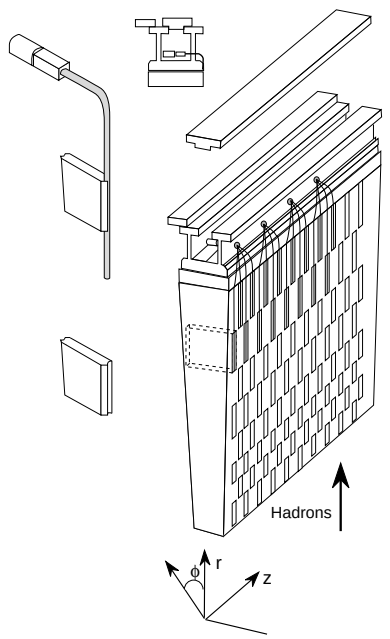


図 2.10: タイル型ハドロンカロリーメーターの構造

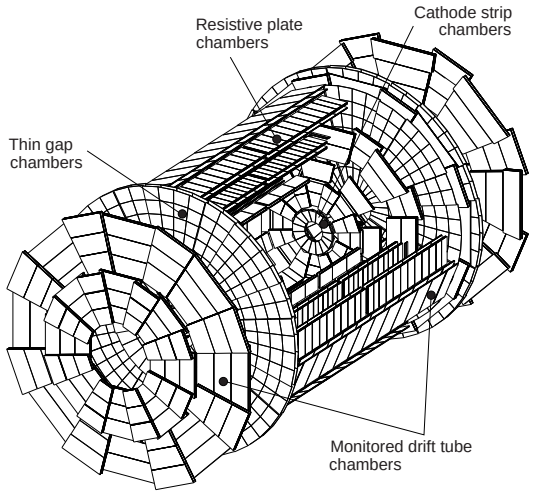


図 2.11: ミューオン検出器

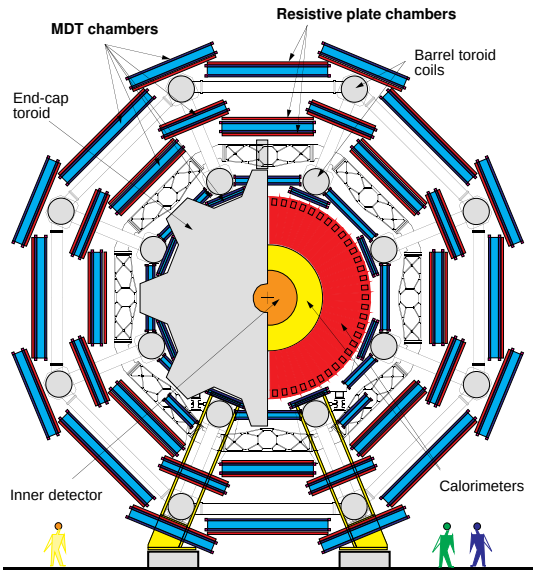


図 2.12: ミューオン検出器の XY 断面

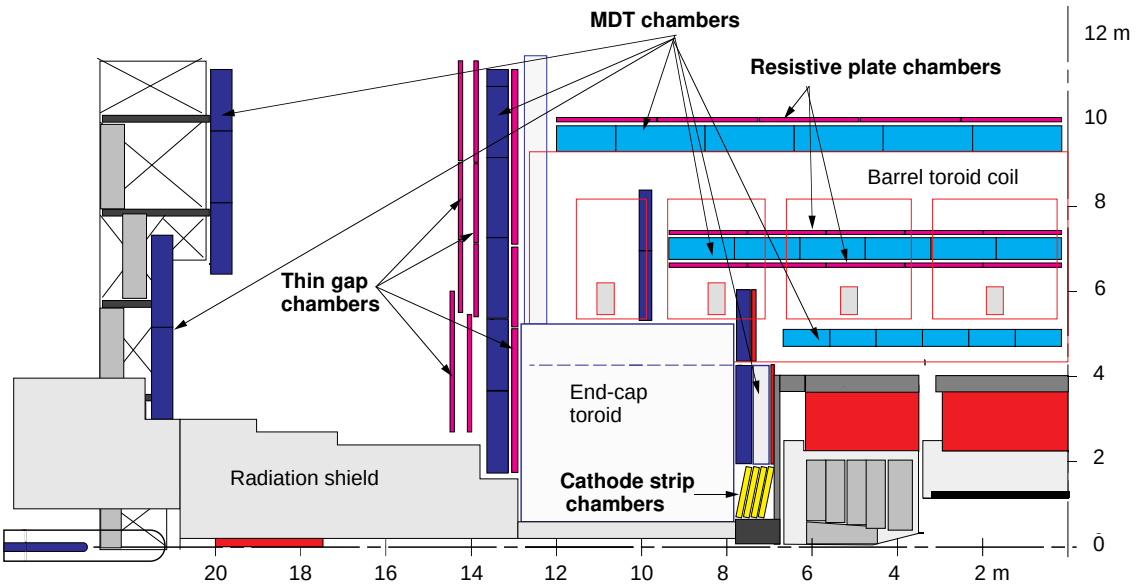


図 2.13: ミューオン検出器の RZ 断面

2.6 トリガーとデータ収集

アトラス実験では 40MHz のバンチ衝突によって、1GHz 程度の反応が起こるとされている。トリガー及びデータ収集システムでは、多数のチャンネルからなる各検出器からのデータをもとに、多くのバックグランドを落しながら、且つ高効率で有用なイベントを選別し、記録していかなければならない。1 イベントあたりを記録するのに必要なデータ容量は、平均 1Mbyte と見積もられており、計算機資源や記録装置からの制限から記録されるデータの頻度は 100Hz 程度に限られている。

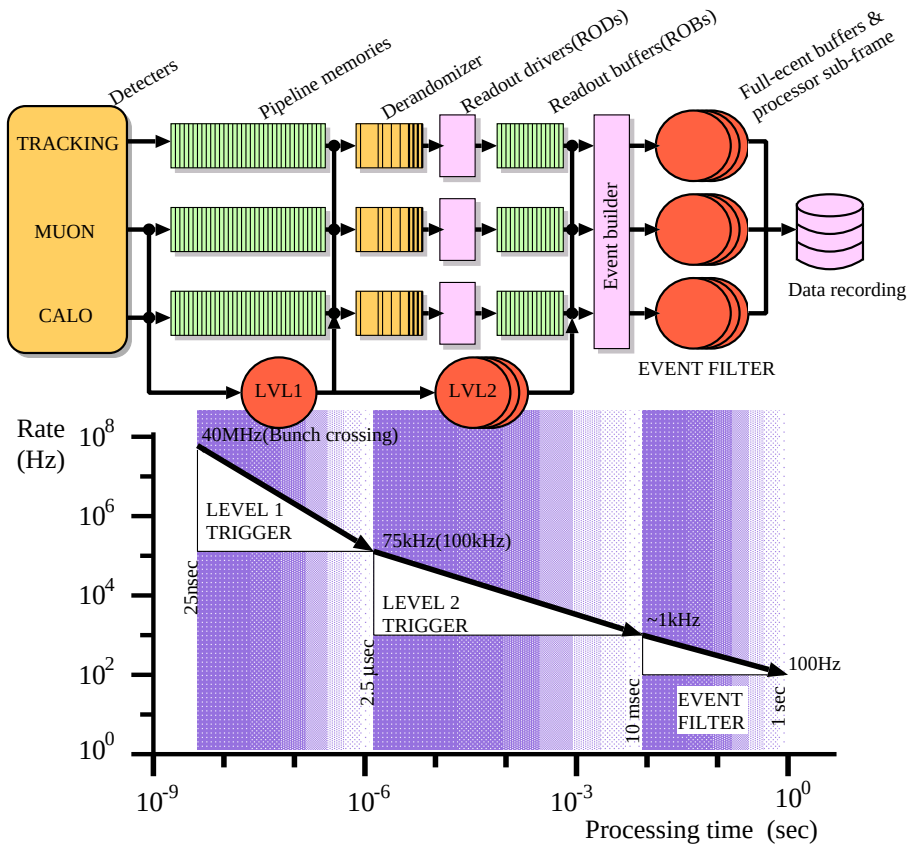


図 2.14: トリガー処理の流れ

2.6.1 トリガーの方法

アトラス実験のトリガー、データ収集システムは図 2.14 に示すように LVL1 (レベル 1) 、LVL2 (レベル 2) 、EF (イベントフィルター) の 3 段階からなる。

- LVL1(レベル 1)
  LVL1 ではカロリーメーター及び、トリガー用ミューオンチェンバー (TGC,RPC) からの情報を使い、トリガー判定を行なう。LVL1 では 40MHz で入力される信号に対して、L1A(レベル 1 アクセプト) を受けた頻度が 75kHz(100kHz) 程度の信号を、LVL2 に向けて出力する。またこの LVL1 トリガーの処理に要する時間は、2.5(2+0.5(予備))μsec で、LVL1 トリガー処理を行なう各部は、その間の信号情報を貯えるバッファーを搭載している。

- LVL2 (レベル 2)
  LVL2 ではカロリーメーター、ミューオン検出器に加え、内部検出器からの完全な位置情報に基づいて処理を行なうが、データ転送、処理資源の制約から、有用な情報を含む領域 (RoI : Region of Interest) からの信号のみを使って、トリガー判定を行なう。RoI は LVL1 での電子、フォトン、ジェット、ミューオンを使って、High-pT と判断された現象の存在する領域である。また LVL2 での判定に要する時間は 10∼100[msec] で可変であり、トリガーの頻度は 1kHz 程度に落される。

- EF (イベントフィルター)
  LVL2 で、LVL2accept の判定を受けた信号は、イベントビルダーを経て EF に送られる。EF は各検出器からの完全な位置情報を基にトリガー判定を行なう。処理に要する時間は約 1[sec] で、トリガー頻度は 100Hz 程度である。また 1 イベントを記録するのに必要な容量は、平均 1Mbyte なので毎秒 100MByte 程度の情報を記録することになる。

2.6.2 データ収集システム

検出器からの信号は、まず L1A (レベル 1 アクセプト) 信号が来るまで、各チャンネル毎にパイプラインメモリー (レベル 1 バッファー) に保持される。レベル 1 の処理時間の 2.5μs の間、25nsec 毎にやってくる信号を保持するため、各チャンネルに対して 100 イベント分のバッファーが必要になる。L1A 判定を受けた信号は、デランダマイザー (derandomizer) [2] に送られ、読み出されるまで、そこに保持される。デランダマイザーから読み出された信号は、順次 ROD(Read Out Driver) に送られる。また、この時、信号情報と共に、その信号がどのバンチに属するかという BCID (バンチ ID) と、どの L1A に属するかという L1ID (レベル 1ID) も送られる。ROD は BCID、L1ID 情報を基に、情報の整合性を確かめて、信号を ROB(Read Out Buffer) に送る。

LVL2 では ROB に貯えられた情報のうち、RoI のデータを使って処理を行なう。LVL2 でアクセプトの判定を受けた信号はイベントビルダーに進み、さらに EF でもアクセプトを受けたイベントは、テープなどに記録される。

2.6.3 検出器の制御

ATLAS 検出器では検出器の制御、保守のために DCS(Detecter Control System) と呼ばれるシステムを用いる。DCS は主に電源、グランドの供給、電流モニター、フロントエンドエレクトロニクスへの閾値電圧の設定、ガス系の圧力、流量の監視などがある。

[2] derandomizer は、ランダムで起こるイベントを一定の間隔で読み出せるようにするものである。

2.7 レベル 1 トリガーシステム

ここでは詳細を後述する TGC トリガーシステムの属する LVL1 トリガーに関して述べる。図 2.15 に LVL1 トリガー処理の流れを示す。LVL1 トリガーはカロリーメーター、トリガー用ミューオン検出器からの信号をもとに、カロリーメータートリガーとミューオントリガーでそれぞれ処理を行ない、CTP(Central Trigger Processor) により、LVL1 の最終的トリガーの判定が行なわれ、RoI の情報を LVL2 に送る。

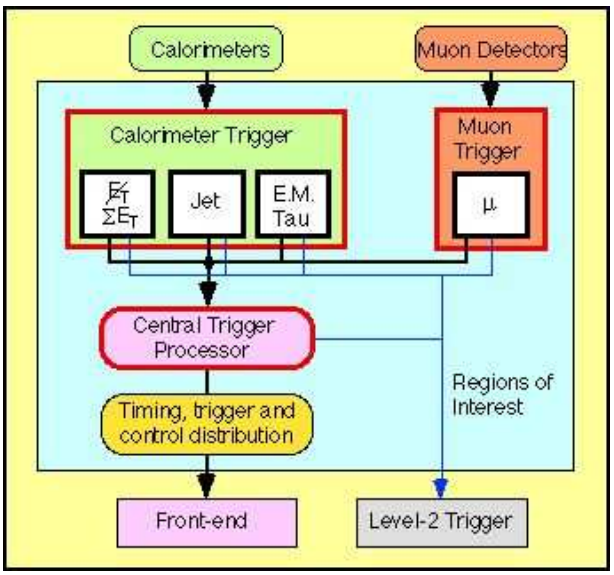


図 2.15: LVL1 トリガー処理の流れ

2.7.1 ミューオントリガーシステム

ミューオントリガーは図 2.16 に示すように、ATLAS 検出器全体を覆うように設置される 3 層の位置検出器からなり、トロイダル磁場により、ミューオンの飛跡がどう変位するかを検出することにより、運動量を測定する。飛跡の変化はミューオンの運動量と、磁場強度を飛跡に沿って積分した量により決まる。また物質内部の多重散乱や、エネルギー損失にも依存する。検出器はエンドキャップ部が TGC(Thin Gap Chamber)、バレル部が RPC(Resistive Plate Chamber) からなる。

2.7.2 CTP(Central Trigger Processor)

CTP はカロリーメーター、トリガー用ミューオン検出器からの情報をもとに、LVL1 トリガー判定を行なう部分である。ミューオン検出器からはミューオンの情報、カロリーメーターからは電子、フォトン、ジェット及びミッシングエネルギーの情報を受け、それぞれに設定された数段階の p_T の閾値に応じてトリガー判定を行なう。

2.7.3 TTC(Timing,Trigger and control distribution)

TTC はカロリーメーター、ミューオン検出器のトリガー用エレクトロニクスに向け、制御用信号を配信する部分である。制御信号には 40MHz の LHC クロック、LVL1 トリガーの判定を伝え

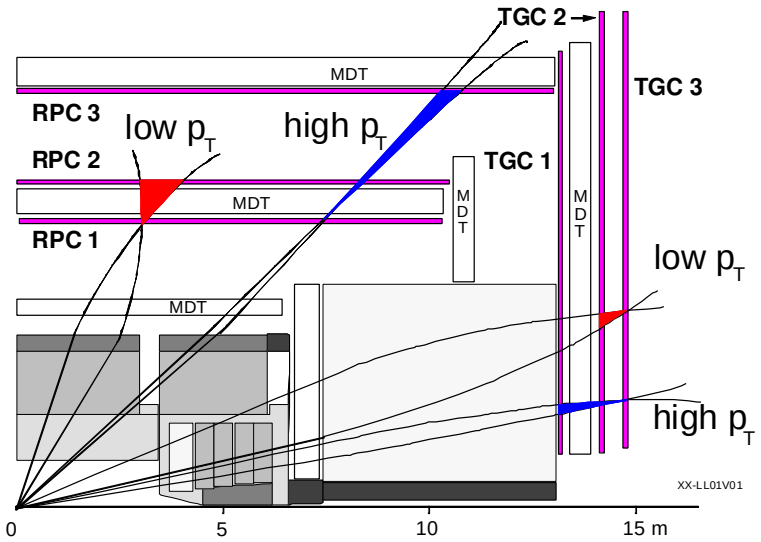


図 2.16: LVL1 ミューオンシステム

る LVL1Accept、バンチ衝突番号のカウントをリセットする BCR(Bunch CounterReset)、イベント番号のカウントをリセットする ECR(Event Counter Reset) などがある。

第3章　TGCミューオントリガーシステム

ここでは本論文の主題である、TGC ミューオントリガーシステムのエレクトロニクスを中心として、TGC やトリガーの仕組みを述べていく。

3.1　TGC のレイアウト

トリガー用ミューオン検出器は、バレル部とエンドキャップ部から構成されるが、このうちのエンドキャップ部に設置されるのが、図 3.1, 図 3.2 に示す TGC である。TGC は 1 ユニットの 1 辺が 1∼2m 程度の大きさの台形状のチェンバーで、これを円形状に配置して、エンドキャップ部を覆う。TGC システムは M1,M2(middle),M3(pivot) 及び EI(Endcap Inner),FI(Forward Inner) の各層から構成される。Z=±(13∼15m) の[1]領域に設置する M2,M3 はそれぞれ 2 層 (doublet) のチェンバーから、M1 は 3 層 (triplet) のチェンバーから構成され計 7 層に、また Z=±7m の領域には 2 層の EI/FI チェンバー (インナーチェンバー) が設置される。TGC は $1 < |\eta| < 2.7$ の領域をカバーし、このうち $|\eta| < 1.9$ の領域をエンドキャップ、$|\eta| > 1.9$ の領域をフォワードと呼ぶ。また EI チェンバーはトロイダル磁石と部分的に交差する位置に設置されるため、その部分にはチェンバーが配置されず、ϕ 方向の 70%をカバーするのみとなる。

TGC は MWPC 型の 2 次元の位置読み出しが可能なチェンバーで、ワイヤー (wire) により R 方向の位置を検出し、ストリップ (strip) により ϕ 方向の位置を検出する (チェンバーの構造に関しては、3.2 節で述べる)。doublet からはワイヤーとストリップから、それぞれ 2 層分の信号を読みだし、triplet からはワイヤーが 3 層分とストリップ 2 層分の信号を読み出す。TGC はこうして XY(Rϕ) 方向の 2 次元の位置情報を読みだし、それぞれの層からの情報をもとに、ミューオンの横方向運動量 p_T を測定する。TGC システムには各層を使ったコインシデンス条件 (middle 及び pivot では 3-out-of-4 (4 層のうち 3 層が信号をだす。)、triplet 部の strip 部に関しては 1-out-of-2、wire 部では 2-out-of-3) があり、これによりバックグランドによるフェイク信号の影響を減らし、チェンバーの低検出効率部分 (チェンバーの構造支持部品の領域で検出効率が下がる。) による検出効率の低下を防ぐ。これらのコインシデンス処理は R 方向、ϕ 方向で独立に行なわれる。またトリガーには high-p_T と low-p_T の 2 種類あり、それぞれ 3 段階の閾値がある。典型的な閾値は high-p_T が 20GeV で low-p_T が 6GeV である。またこのトリガー判定は R 方向、ϕ 方向の情報を合わせて行なわれる[2]。

[1] ビーム衝突点を原点とするビーム軸の方向で、LHC トンネルを上から見た時、反時計周りの方向にあたるのが Z 軸。また Z 軸にに垂直な平面が、XY 平面で LHC トンネルの中央を向いているのが X 軸である。

[2] 現在のトリガーに関するシミュレーションの結果では、EI/FI チェンバーを使用しなくても、十分なトリガー効率が得られるとされているが、また低エネルギー粒子によるバックグランドの影響も懸念されており、これは EI/FI チェンバーを使うことで落せると考えられている。このことに柔軟に対処するため、EI/FI チェンバーは必要であり、また EI/FI チェンバーからの情報は、後述する Sector Logic で使われるため、比較的簡単に、その設定を変更することが可能となっている。

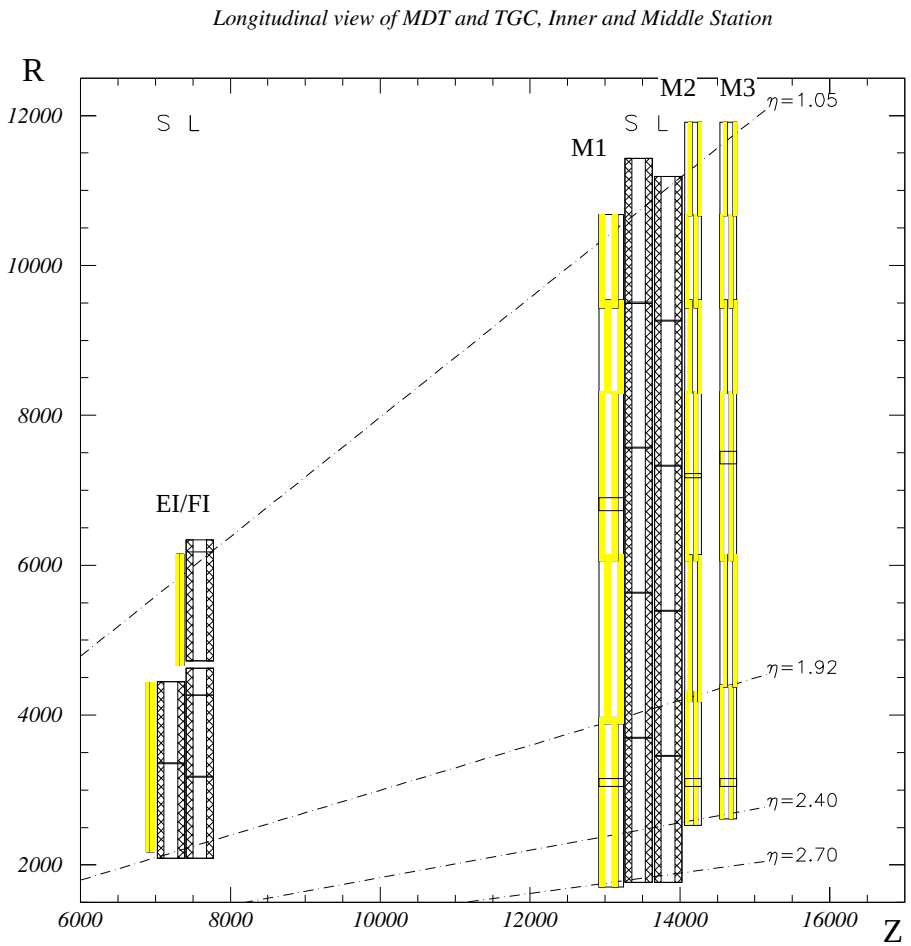


図 3.1: TGC のレイアウト (RZ 断面)

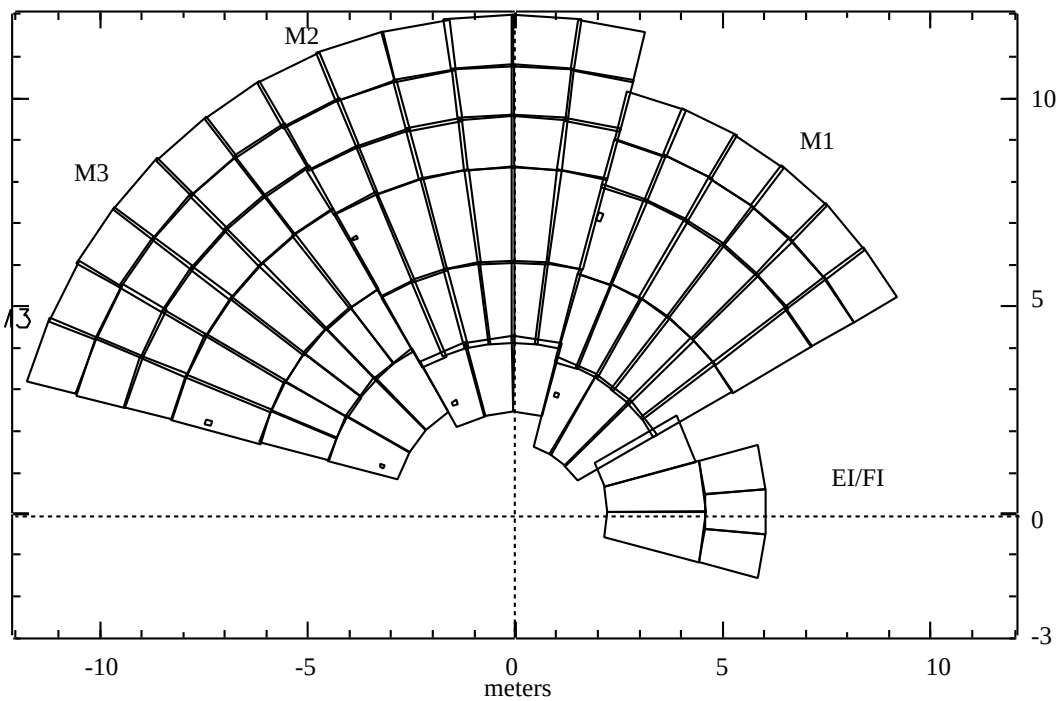


図 3.2: TGC のレイアウト (XY 断面)
図は M1,M2,M3,EI/FI チェンバーの 1 オクタント分を、それぞれ表示したもの。

3.2 TGC の構造と特性

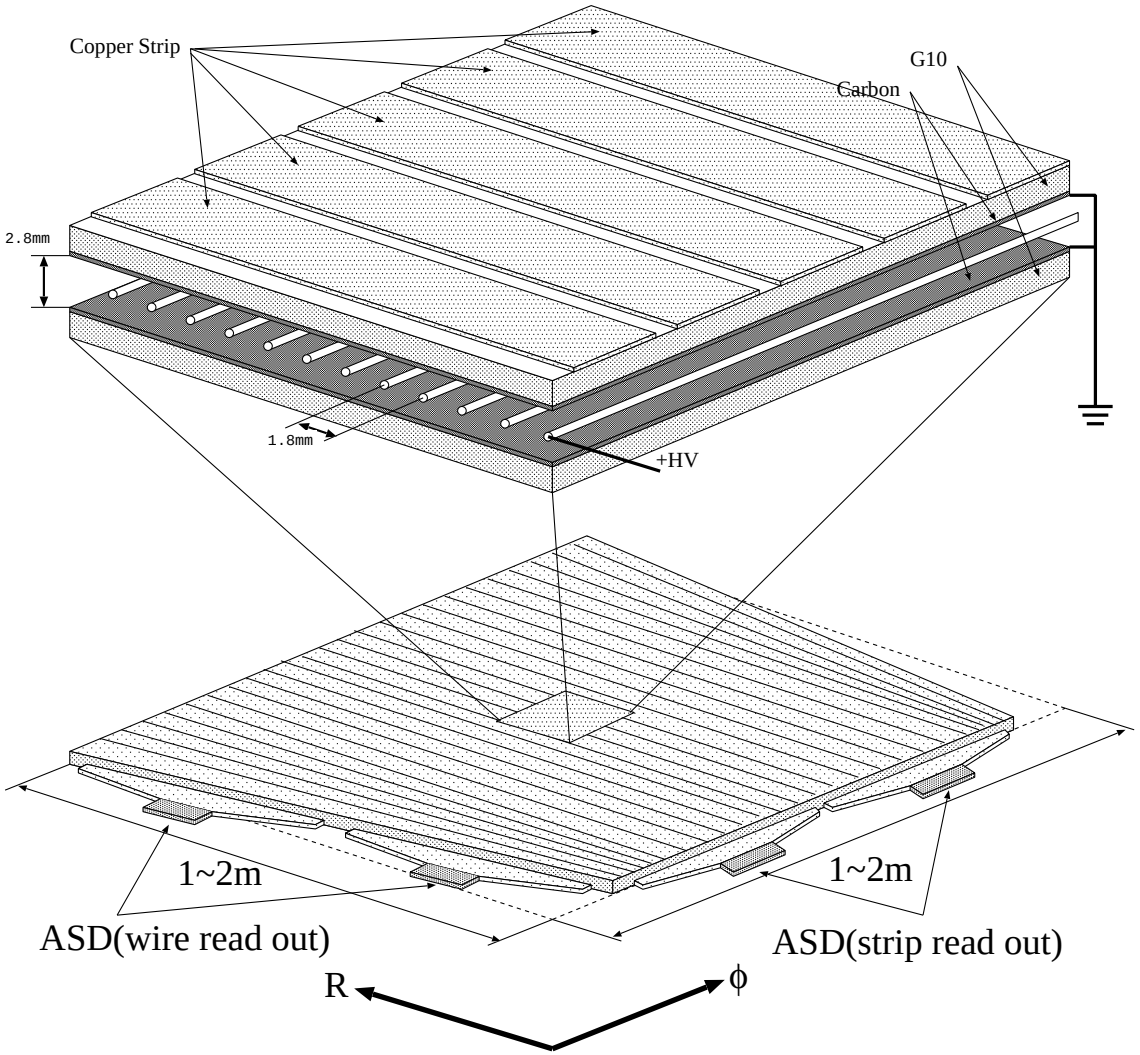


図 3.3: TGC の構造

図 3.3 に TGC(Thin Gap Chamber) の 1 層 (singlet) 分の構造を、表 3.1 に TGC の主なパラメータを、表 3.2 に TGC の読みだしチャンネル数をそれぞれ示す。TGC は MWPC(Multi Wire Propotional Chamber) の一種で、その特徴的な点はアノードワイヤーとカソード面の間隔が 1.4mm と狭いことである。TGC は図 3.3 に示すように、高電圧がかけられたアノードワイヤーを、グランドの平面で挟んだ構造している。この間隔には CO_2 と n-pentane の混合ガスが満たされており、ガス中を荷電粒子が通過すると、その経路にあるガス分子がイオン化される。ここで生成された電子は、アノードワイヤーとカソード面の間の電気力線に沿ってアノードワイヤーに向かう。ワイヤーの近傍では電場が大きく、電子はさらにその周辺のガス分子をイオン化し、2 次電子を生成し電子雪崩を起こし、これが信号として読み出される。ここでさらに電場が強い場合、この電子と陽イオンの再結合により紫外線が発生し、これがさらにガス分子のイオン化を引き起こし放電にいたるが、これを防ぐ為に、TGC では CO_2 に混ぜられた n-pentane により、紫外線を吸収し放電を起こりにくくしている (クエンチ効果)。またカソード面には高抵抗のカーボン塗装を施しており、その外側のストリップに誘起された電荷を読み出す構造になっている。

TGC ではアノードワイヤーからの信号で R 方向の位置を読みだし、ストリップから ϕ 方向の位置を読み出す。ワイヤーは 6 本から 20 本を 1 グループとして (幅にして 10.8mm から 36mm) 読み出す。またストリップは 1TGC あたり 32 本あり、エンドキャップ領域では 4mrad、フォワード領域では 8mrad に相当する幅 (15.1∼53.4mm) のストリップで信号を読み出す。それぞれの読みだしは、チェンバーの側面に取り付けられた ASD(Amplifier Shaper Discriminator) によって行われる。ASD に関しては 3.4.2 節で述べる。

また図 3.4 に TGC のタイムジッター (荷電粒子がチェンバーに入射してから信号を出すまでの時間のばらつき) を示す。これは 3GeV の π を用いたビームテストの結果で、チェンバーに対するビームの入射角度を変えて測定した結果である。図から見てとれるようにタイムジッターはほぼ 25nsec に収まっている。25nsec は LHC クロックの周期であり、このことは後述するタイミング調整に関係する重要な事項である。

表 3.1: TGC の主なパラメータ

パラメータ	値、種類
Gas gap	2.8mm
Anode wire pitch	1.8mm
Wire diameter	50μm
Wire potential	3100V
Gas mixture	$CO_2/n-C_5H_{12}$ (55%/45%)
Gas amplification factor	10^6
Anode read out pitch	10.8∼36.0mm
Time resolution(0[deg] incidence)	99% efficiency for 25ns gate
Read out strip angle	4mrad(Endcap),8mrad(Forward)

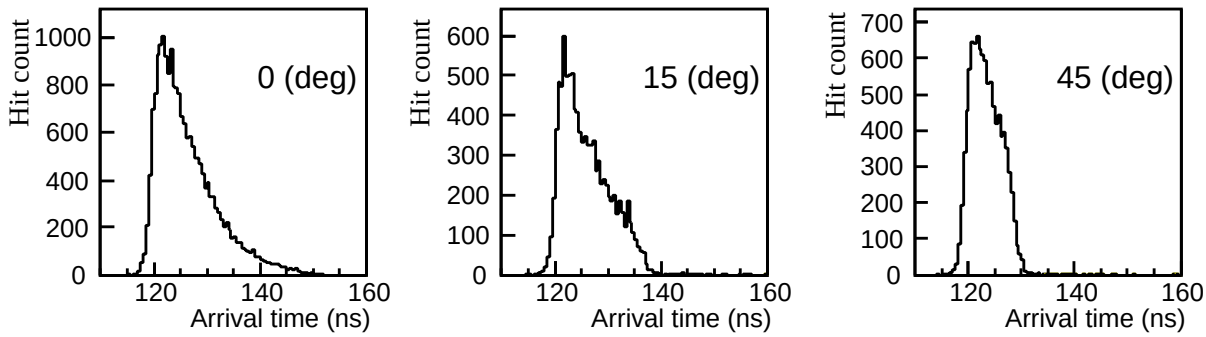


図 3.4: TGC のタイムジッター
ここで角度 (deg) はチェンバー面と垂直な方向 (粒子の入射する側) と、粒子の入射方向のなす角度である。ビームテスト時には TGC の前方に、粒子の通過時刻を検出する検出器を設置している。図の横軸はこの検出器からの信号と、TGC からの信号の到着時間の差を、TDC で読んだ値である。この分布の 99%を占める時間幅をタイムジッターと呼ぶ。

3.3 トリガー処理の方法

TGC トリガーシステムでは、1 辺が 1~2m あるチェンバーを両側 (A サイド、C サイド)[3] で 3600 枚使い、$6296m^2$ の大面積を覆っているが、トリガー処理のためオクタント (octant), トリガーセクター (trigger sector), サブセクター (sub-sector) などの分割の単位がある。図 3.5 に pivot (ビーム衝突点から最も遠い層) のオクタントを示す。オクタントはさらにエンドキャップ ($|\eta| < 1.9$) とフォワード ($|\eta| > 1.9$) の領域に分けられ、またエンドキャップ部では ϕ 方向に 6 分割した領域を、フォワード部では 3 分割した領域を、それぞれトリガーセクターと呼ぶ。また図 3.6 に示すように triplet,middle 上にも、pivot 上の分割に対応する分割がある。トリガー処理は 1 つのトリガーセクターごとに独立して行なわれ、またトリガーセクターは、エンドキャップ部では η 方向に 37 分割、ϕ 方向に 4 分割したものを、フォワード部では η 方向に 16 分割、ϕ 方向を 4 分割したものを、それぞれサブセクターと呼ぶ。サブセクターは 8 ワイヤーグループと 8 ストリップに対応し、これらはトリガー処理の最小単位になり、また 1 つの RoI(Region of interest) に対応している。また実際のチェンバーからの信号は 16ch 毎に ASD を介して読み出されるが、この 16ch はサブセクター 2 つ分に対応しているわけではなく、3.4.4 節で述べる PS-Board 上で、サブセクターの分割に変換して扱われる。

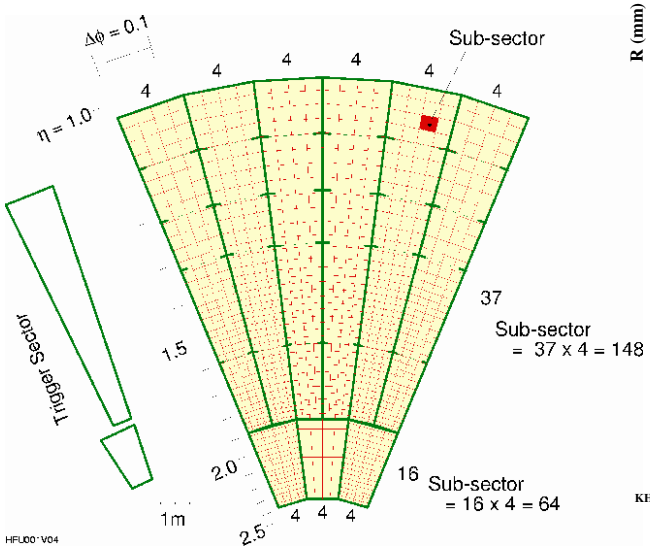


図 3.5: TGC オクタントの分割

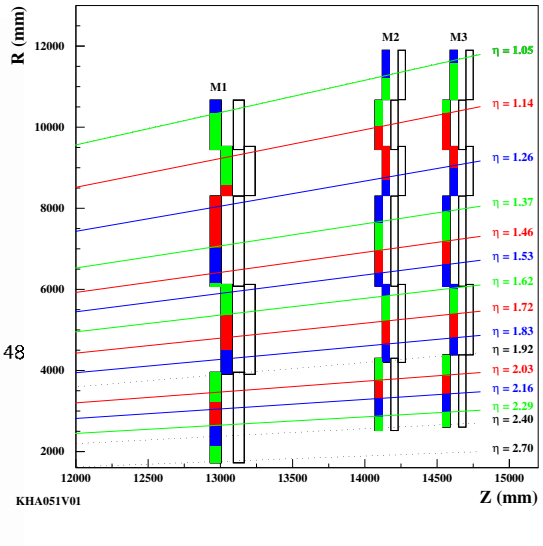


図 3.6: TGC のワイヤ方向の分割

表 3.2: TGC 読みだしチャンネル数

	M1	M2	M3	I	total(2side)
Wire	37152	35804	34512	2856	220448
Strip	13824	16896	16896	3072	101376
total	50976	52800	51408	5728	321824

[3] A サイドは LHC トンネルを上から見たとき、反時計周りの方向にあたる側、C サイドはその逆。

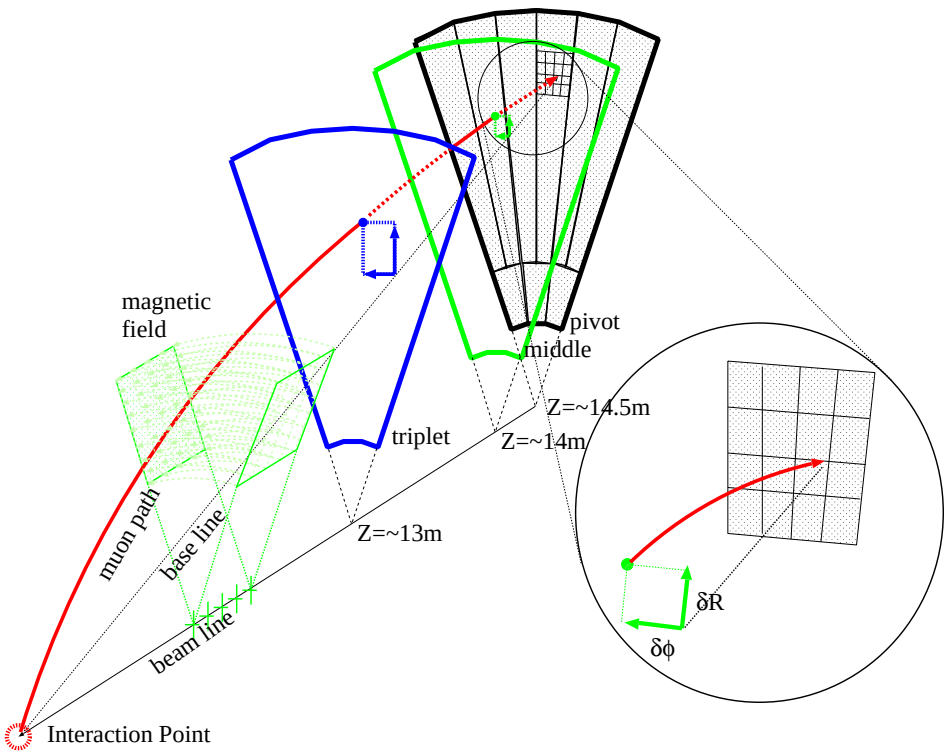


図 3.7: p_T の判定の仕方

トリガー処理に使われる p_T の判定は図 3.7 に示すように、pivot 層とビーム衝突点を結ぶ直線を基準線として、middle 及び triplet チェンバー上での粒子の検出された点が、どれだけずれているかを測定することで行なわれる。また low-p_T に関しては 3.4.4 節で述べる Slave Board ASIC 上で、High-p_T に関しては 3.4.5 節で述べる High-pT ASIC 上で処理が行なわれる。Slave Board ASIC では middle 及び pivot からの情報をもとに、High-pT ASIC 上では triplet と pivot からの情報をもとに、R 方向、ϕ 方向それぞれのミューオンの位置の差、$\delta R, \delta\phi$ を[4]判断することにより p_T を測定する。PS-Board,high-pT Board 上での処理を受けた信号は Sector logic に送られ、ここでは R 方向、ϕ 方向の情報がまとめられ、コインシデンス処理、p_T 閾値ごとの判定が行なわれる。

[4] トロイダル磁場は理想的には ϕ 方向のみの磁場であるが、実際は R 方向の成分も存在するため、ミューオンは ϕ 方向にも曲げられる。そのため ϕ 方向の変位 $\delta\phi$ も決める必要がある。

3.4 トリガー用エレクトロニクス

3.4.1 概要

図 3.8 に TGC の LVL1 エレクトロニクスの流れを示す。TGC からの信号はまず、ASD(Amplifier Shaper Discriminator) によって増幅、整形され、次段の PS-Board 上の Patch Panel ASIC でタイミング調整、LHC クロックとの同期がとられ、チェンバー境界を考慮した OR 論理を通過し Slave Board ASIC に入る。doublet 用 Slave Board ASIC では、4 層からなる middle,pivot チェンバーからの信号をもとに、コインシデンス処理が行なわれ、また low-p_T の判定が行なわれる。一方、triplet 用 Slave Board ASIC では 3 層からなる triplet チェンバーからの信号をもとに、コインシデンス処理を行なう。また Slave Board ASIC には LVL1 判定が出るまでの間、信号情報を貯えるバッファーがあり、L1A(レベル 1 アクセプト) の判定があり次第、L1A を受けた信号が読み出される。Slave Board ASIC の次段の high-pT ASIC では、triplet からの信号と合わせて p_T 判定が行なわれ、次に Sector Logic に送られる。Sector Logic では R 方向、ϕ 方向の情報を合わせて、p_T の判定が行なわれる。

3.4.2 ASD

ASD(Amplifier Shaper Discriminator)board はチェンバー側面に取り付けられるもので、1 チップで 4 チャンネル分の処理を行なう ASD チップ 4 個から構成される。ASD はチェンバーからの信号を受け増幅、整形を行ない、これを Patch Panel ASIC に送る。またこの際の閾値電圧が PS-Board から供給される。また ASD と PS-Board は回路の設置位置が異なるため、この間のグランドレベルが異なる可能性を考慮して、6 章で詳細を述べる LVDS(Low Voltage Differential Signaling) で信号を送る。また ASD は回路系の校正、診断のためにテスト信号を出す機能も持ち、このテスト信号のトリガーは Patch Panel ASIC より送り出される。

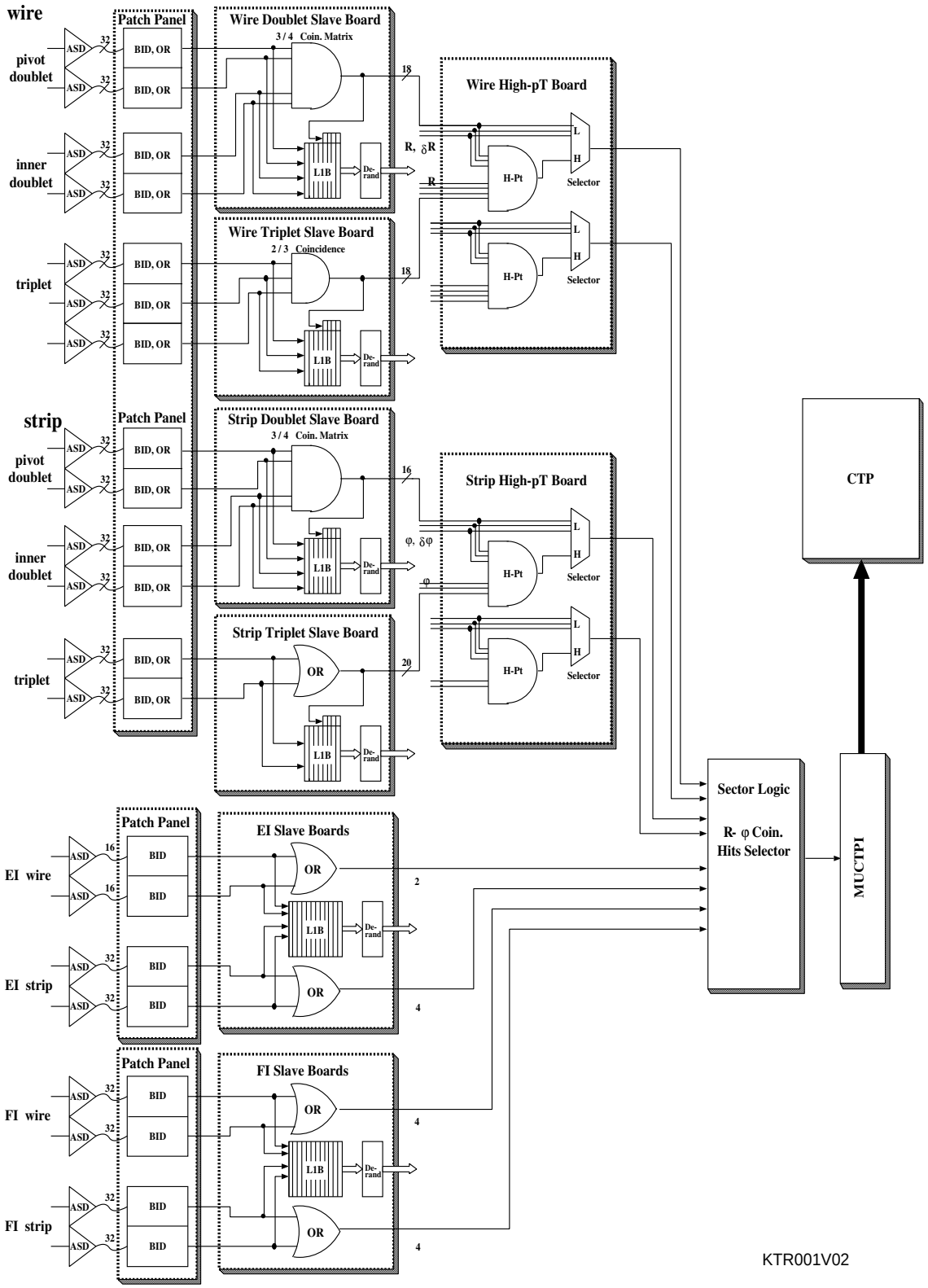


図 3.8: TGC トリガーエレクトロニクスの流れ
図では Patch Panel と Slave Board が別々にあるが、これは 1 つのボード,PS-board で実現されることになった。

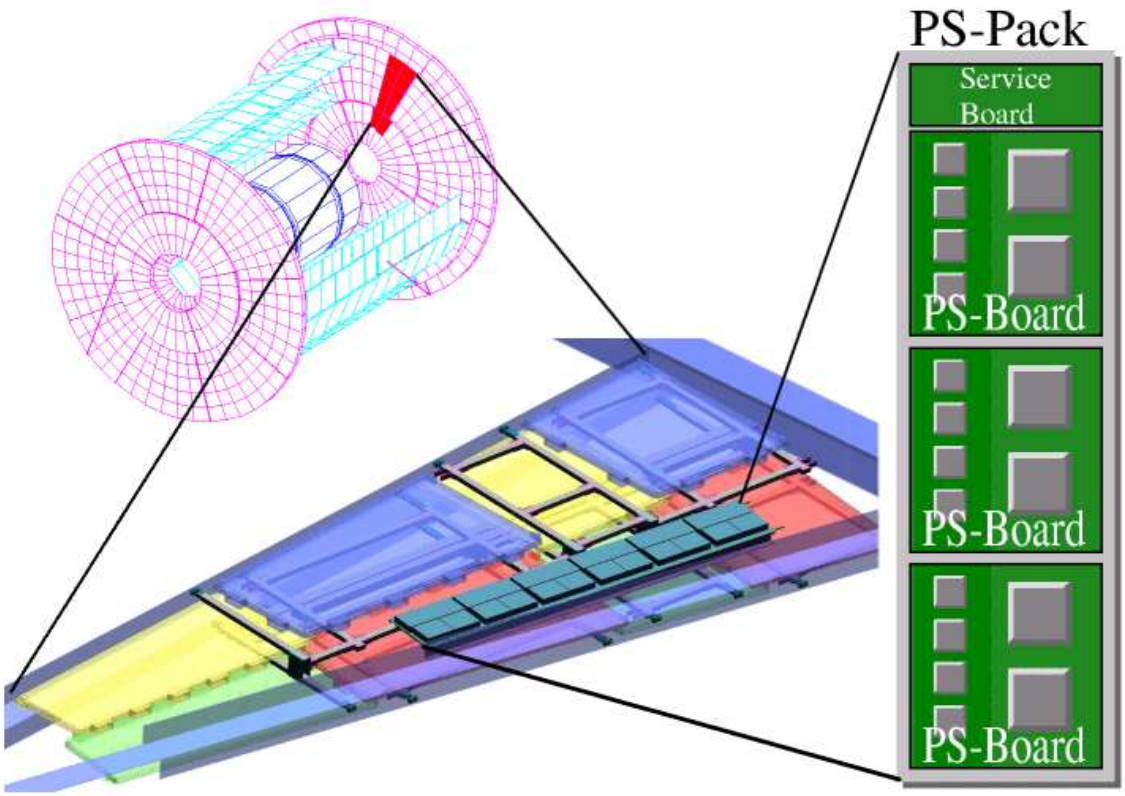


図 3.9: ps-pack の配置

3.4.3 PS-Pack

PS-board は信号のタイミング調整を行なう Patch Panel ASIC と、コインシデンス処理を行なう Slave Board ASIC を設置するボードで、図 3.9 に示すようにチェンバーと並べて triplet の側面、及び doublet の側面に設置される。オクタント毎に 3 つ (1/24 TGC Wheel[5]に 1 つ) 設置され、triplet 側では 1 つの PS-pack に、52 個の Patch Panel ASIC と 23 個の Slave Board ASIC が搭載され、また doublet 側には 1 つの PS-pack に 76 個の Patch Panel ASIC と 36 個の Slave Board ASIC が搭載される。これらの ASIC は PS-board 上に設置さるさらに PS-pack には、これらのエレクトロニクスの制御を行なうための Service board が設置される。図からもわかるように PS-pack 上のエレクトロニクスは TGC の側面に設置されるため、強い放射線環境下に置かれることになる。このため、これらのエレクトロニクスには、放射線に対する影響を、十分にテストすることが求められ、またそれに対する工夫が必要になる。放射線の影響に関しては 3.4.9 節で述べる。

[5]TGC Wheel は 1 層の円型に配置されたチェンバーのことである。

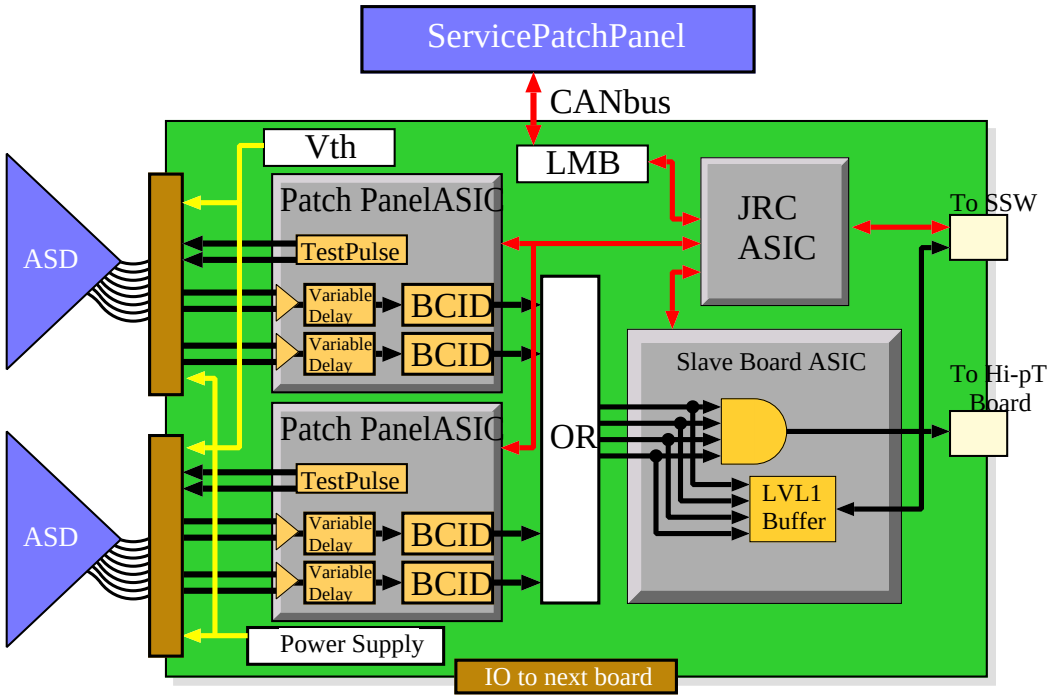


図 3.10: PS-board の構成

3.4.4 PS-board

- PS-board
  PS-board[6]の構成を図 3.10 に示す。PS-board の機能は大分して信号処理と制御がある。信号処理部ではまず、ASD からの信号を受け、サブナノ秒の精度でのタイミング調整を行ない、LHC クロックとの同期がとられる。次にチェンバー境界を考慮した OR 論理回路を通過し、コインシデンス処理、(double チェンバーに対しては p_T 判定も行なう。) の後、信号は high-pT ボードに向け送られる。また PS-Board には LVL1 バッファーも備えられていて、L1A があり次第、信号は読み出され、スタースイッチ (PS-Board からの信号を束ねて光信号に変換するボード) に向け送られる。制御系では、Service board(Service Patch Panel) からの CAN プロトコル[7]の制御信号を、LMB(Local Monitor Board) が受け、個々の ASIC の制御を行なうための JTAG プロトコルの制御信号を出す[8]。またこの JTAG の制御信号は JRC(JTAG Routing Control)ASIC により、PS-board の内の各 ASIC に分配される。また ASD を対象として、テストパルスを出す機能、閾値電圧の設定、電源の供給の機能がある。

- Patch Panel ASIC
  Patch Panel ASIC 上では主に、信号のタイミング調整とバンチ識別 (信号を LHC クロックに同期させる。) を行なう。ASD からの LVDS 信号は LVDS レシーバーにより TTL 信号に変換され、サブナノ秒の精度のタイミング調整を受け、BCID(Banch crossing ID) 回路により LHC クロックと同期がとられ、出力される。また ASD に向けテストパルスを発生するテ

[6]PS-board は Patch Panel Board と Slave Board で別々の基板で実現される予定だったものが、1 つになったものである。

[7]CAN(Controller Area Network) はマルチマスターのシリアルバスシステムで、アドレスの概念がなく、送信側は識別子をつけてメッセージを配信し、受信側がその識別子をみてメッセージを受けるかどうかを決めるプロトコルである。

[8]JTAG に関しては第 4 章で詳細を述べる。

ストパルス回路も搭載させる。このテストパルスの振幅、タイミングは可変で、これらの設定値も JTAG プロトコルにより外部から制御可能である。Patch Panel ASIC に関しては 6 章で詳細を述べる。

- Slave Board ASIC
  Slave Board ASIC は Patch Panel からの信号を受け、コインシデンス処理、(doublet チェンバーからの信号に関しては low-p_T 判定) を行なう。これらの処理は triplet,doublet,wire,strip,EIFI と別々の処理が必要になるが、Slave Board ASIC ではこれらを 1 種の ASIC で実現し、使い方により機能を切替えて用いる。また LVL1 バッファーとして深さ 128bit のメモリを備え、LVL1accept に応じてこれを読み出す回路を搭載した。またケーブル遅延などによるタイミングのずれを補正するため、1/2 クロック単位の遅延回路を搭載した。また Slave Board ASIC 上の様々なパラメータも JTAG プロトコルによって制御できるようにした。Slave Board ASIC に関しては 7 章で詳細を述べる。

3.4.5 High-pT Board

High-pT board 上では Slave Board ASIC を介してやってくる、doublet と triplet からの信号を受け、High-pT の判定を行なう。Slave Board ASIC からの信号はエンコード[9]されているので、High-pT ASIC 上ではまずこれをデコードし、トリガーマトリックスによって δR, ($\delta\phi$) を求め、High-pT のヒットが複数あった場合の処理を行ない、さらにエンコードし出力する。図 3.11 に示すように、High-pT Board はオクタントの端に設置され、信号は光信号に変換され、セクターロジックに送られる。ケーブル遅延によるタイミングのずれを補正するため、1/2 クロック単位の遅延回路を搭載している。またこのパラメータも JTAG プロトコルによって制御される。High-pT ASIC に関しては 7 章で詳細を述べる。

3.4.6 Secter Logic

Sector Logic では、これまで R 方向、ϕ 方向で独立に処理されてきた信号を、ここで始めて統合し pT の判定を行なう。Sector Logic は主に Rϕ コインシデンス、Truck pre-selector, Truck-selector から構成される (図 3.12)。High-pT からの信号はまずデコードされ、ssc(sub-sector cluster(R 方向に 2 つ、ϕ 方向に 4 つの sub-sector の集まり)) 毎に 6 段階 (high-pT,low-pT でそれぞれ 3 段階) の pT 判定が行なわれ、1trigger secter 分 (Endcap の場合 19ssc,Forward の場合 8ssc) の判定が、6 段階の各 p_T に対応する 6 つの Track pre-selector に送られる。6 つの Track pre-selector ではそれぞれ p_T の大きな順に 2 つづつ選択され、計 12 トラックが次段の Track selector に送られる。Track selector ではこの 1trigger secter の中から p_T の大きな 2 つを選んで、6 段階の p_T 判定と位置情報を MUCTPI に送る。(図 3.13) また secter logic は FPGA を使って実現され、High-pT Board からケーブル長にして 80m 離れた、エレクトロニクス室 (USA15) に設置される。

3.4.7 MUCTPI(Muon Trigger Interface to CTP)

MUCTPI は Sector Logic からの trigger sector の情報をもとに、また RPC の情報も合わせて、ミューオントリガーの最終的な判定を行なう部分である。この信号は CTP に送られる。

[9]Slave Board ASIC から High-pT ASIC へは位置情報と位置変位情報が送られるが、High-pT ASIC では 7 つの Slave Board ASIC から信号を受ける必要がある。また ASIC の入出力のピン数には限界があるため、これらのデータはエンコードし bit 数を減らして転送される。

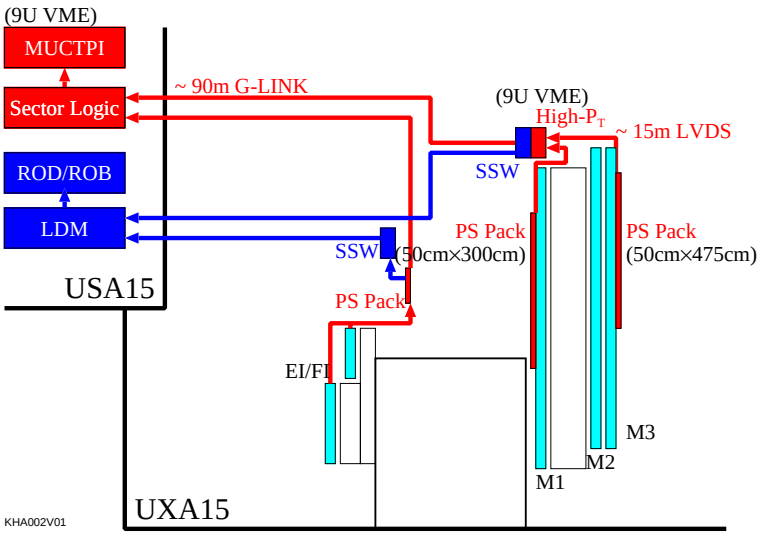


図 3.11: TGC トリガーシステムの配置

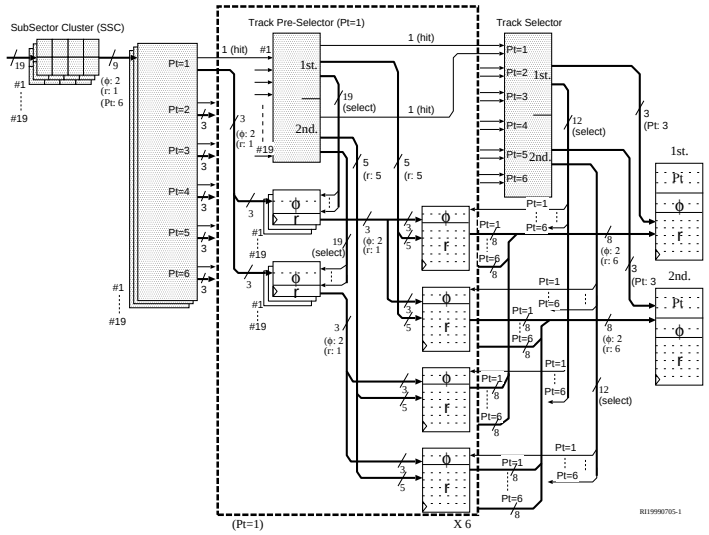


図 3.12: Sector Logic のブロック図

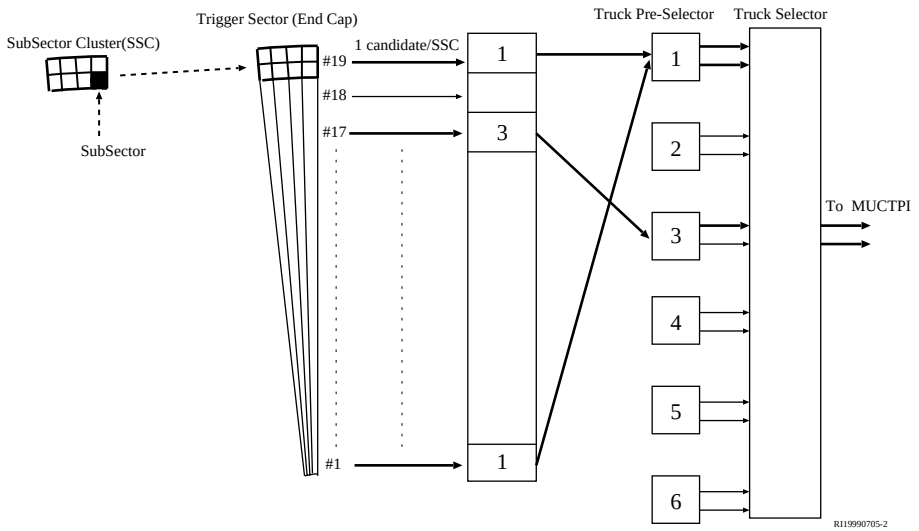


図 3.13: Sector Logic 上の処理のながれ

3.4.8 タイミング調整

LHC のバンチ衝突頻度は 40MHz、周期は 25nsec であり、これは光速の飛程に換算すると 7.5m になる。また ATLAS 検出器の全長が 47m と大きいことを考えると、あるバンチ衝突により生成された粒子が検出器を通過するよりも前に次のバンチ衝突が起こることになる。このような早いタイミングで起こるバンチ衝突の中、各検出器及び信号処理系のデバイスは各バンチ衝突を正確に見分けながら処理を行なう必要がある。また各信号経路には必ず遅延の要素があり、信号処理を行なうデバイスはそれらを考慮して開発される必要がある。

TGC トリガーシステムの場合、ミューオンの TOF, TGC のタイムジッター (TGC がミューオンを受けてから信号を出力するまでの時間のばらつき) 、各部の信号伝搬遅延などがあり、さらに各処理の基準になる LHC クロックがデバイスの各部に到着するまでの時間のばらつきも考慮する必要がある。これらのタイミングは 30 万以上のチャンネルからなる検出器の各チャンネル毎に異なるのである。TGC トリガーシステムには、これらのタイミングの調整を行なうために処理系の各部に遅延回路を搭載している。ここではそのタイミング調整に関して述べる。

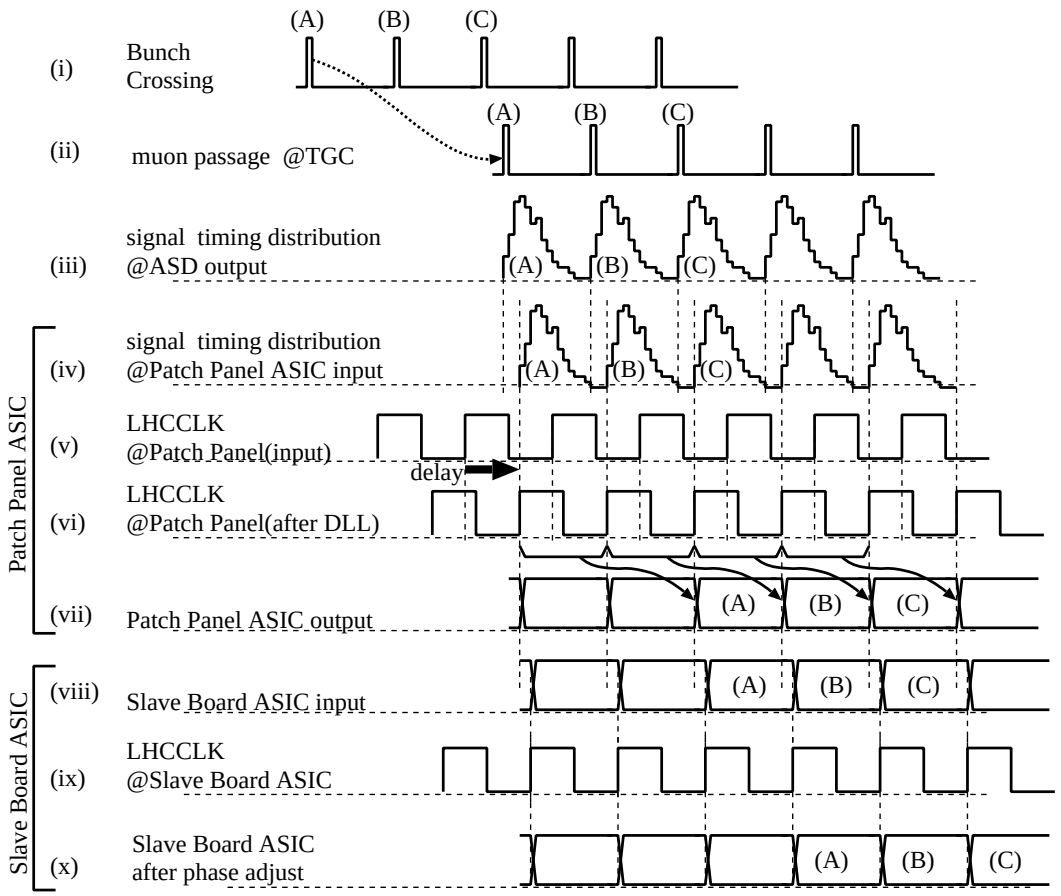


図 3.14: ASD から Slave Board ASIC までの信号のタイミングチャート

Patch Panel ASIC でのタイミング調整

Patch Panel ASIC は、来た信号がどのバンチ衝突から来たものなのかを決定する部分で、高い精度でタイミング調整が必要な部分である。信号が Patch Panel ASIC に来るまでには以下のような遅延要素が考えられる。

- ミューオンの TOF(Time Of Flight)
- TGC のタイムジッター (TGC がミューオンを受けてから信号を出すまでの時間のばらつき)
- ASD, Patch Panel ASIC 間の配線遅延

この内、ミューオンの TOF と配線遅延は、読み出す検出器の位置と配線長が決定すれば、イベント毎のタイミングのずれは無視出来るが、TGC のタイムジッターは TGC の特性のところでも述べたように 20nsec 程度あり、それはミューオンの入射位置 (この位置とはワイヤ間のどの部分かという意味で、どのチャンネルかという意味ではない)、角度によって異なるため、その時間のばらつきはイベント毎に異なる。よってバンチの識別はこの 20nsec の分布を囲むようなゲートで行なう必要があり、このため Patch Panel ASIC には、サブナノ秒の精度で遅延を調節できる DLL(Delay Lock Loop) 回路が搭載されている。

図 3.14 の (iv)∼(vii) にこのタイミング調整の様子を示す。図の (i)∼(iv) は信号到着時間のタイミングの分布を表す (信号の波形ではない)。(iv) のような分布をもって Patch Panel ASIC に入ってくる信号を、(v) のようなタイミングのクロックでバンチ識別を行なうと、識別を行なうクロックの上がりエッジは信号の分布を囲んでいないので、正しくバンチ識別されない。そこで遅延回路を使って、(v) のクロックを信号の分布を囲む (vi) のタイミングまでずらすことで、正しくバンチ識別を行なう。Patch Panel ASIC の出力は (vii) のようなクロックに同期した信号で、これは Slave Board ASIC へ送られる。

Slave Board ASIC,High-pT ASIC でのタイミング調整

ここでのタイミング調整の理由の 1 つは、信号とそれを受けとるクロックのエッジが重なる場合に、対処するものである。図 3.15 に示すように、図中の Input に対して CLK でラッチをするとき、Input のエッジに対して、クロックのエッジが近接していると、わずかなタイミングの差で、出力のタイミングが異なる場合がある。例えば図の Input を CLK2 でラッチすると、出力は OUT2 のようになるが、別のチャンネルなどで、わずかにクロックが早く、CLK1 のタイミングで来ると、出力は OUT2 のようになってしまう。これに対処するため、CLK の下がりエッジでラッチをかけるなどする必要がある。図 3.14 の (viii) のタイミングの信号に対して、Slave Board ASIC 上のクロックが (ix) のタイミングである場合、クロックのどのエッジでラッチがかかるかが不確定になる。図の (x) の出力は信号を一度クロックの下がりエッジでラッチした後に、上がりエッジのラッチをかけた場合のものである。タイミング調整のもう一つの理由は、ケーブル遅延により生じる信号到着時間の不一致に対処するためである。Slave Board ASIC の場合、pivot チェンバーと middle チェンバーの距離の差を考慮して、遅延時間を 1/2 クロック単位として、可変の幅を 0∼1.5 クロックとした。High-pT ASIC の場合は pivot-middle 間よりも長い、doublet-triplet 間のケーブル遅延を考慮して、幅を 0∼3.5 クロックとした。

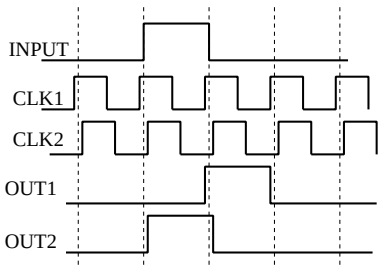


図 3.15: クロックと信号のエッジが重なる場合のタイミングチャート

3.4.9 放射線の影響

TGC エレクトロニクスを含む ATLAS 検出器は強い放射線環境下に設置される。このため各部の検出器、信号処理系には、放射線環境下での長期の安定した動作が求められている。またこの基準は RHA(Radiation Hardness Assurance) working group により与えられ [15],[16]、TGC エレクトロニクスが設置される部分の放射線量は表 3.3 のようにまとめられる。

表 3.3: 放射線量の見積もりと安全係数

	SRL(10 yrs)	SF1	SF2	SF3	RTC(10 yrs)
TID	$2.27 \times 10^0 [Gy]$	3.5	5	4	$1.59 \times 10^2 [Gy]$
NIEL	$2.58 \times 10^{11} [n_{(\sim 1MeV)}/cm^2]$	5	1	4	$5.16 \times 10^{12} [n_{(\sim 1MeV)}/cm^2]$
SEE	$6.54 \times 10^9 [h_{(>20MeV)}/cm^2]$	5	1	1	$3.27 \times 10^{10} [n_{(>20MeV)}/cm^2]$

ここに SRL(Standard Radiation Level) は RHA によるシミュレーションの結果である。安全係数 SF1 は SRL のシミュレーションの誤差で、SF2 は LDRE(Low Dose Rate Effect) で γ 線などを長期間にわたり照射した場合の効果である。SF3 はデバイス製造時のロット間で生じる、パラメータの違いを考慮したものである。また RTC(Radiation Tolerance Criteria) は、SRL の値にこれらの安全係数をかけた値で、各デバイスには、これらのこの照射に対して安定して動作することが求められる。以下に主な放射線の影響に関して述べる。

TID (Total Ionising Dose)

X 線、γ 線によるイオン化の影響を積算して評価される量で、TGC の設置される場所では 10 年間で 2.27[Gy(Si)] の積算吸収線量が見積もられている。半導体を長期間放射線環境下に置くとイオン化の影響で次第にその特性が変化し劣化していく。CMOS の場合はゲートに使われるポリシリコンがイオン化されることで、半導体パラメータが変化しうる。このため今回は Patch Panel ASIC と同じプロセスの試験用の IC を使って γ 線の照射試験を行なった。これに関しては 8 章で詳細を述べる。

NIEL(Non Ionising Energy Loss)

NIEL は 1MeV 程度の中性子により半導体中のシリコンに影響を与え、その特性を劣化させるものである。TGC の設置される場所では 10 年間で $2.58 \times 10^{11} [n/cm^2]$ の照射が見積もられており、各デバイスには中性子の照射による、半導体のパラメータの変化を測定する試験が必要とされている。TGC エレクトロニクスでは既に ASD に関して、中性子の照射試験が行なわれ、安定して動作することが確かめられている。これから開発されてくる ASIC に対しても、照射する必要がある。

SEE (Single Event Effect)

SEE は積算吸収量には依存せず、単発の放射線粒子によって引き起こされる現象で、SEU(Single Event Upsets),SEL(single Event Latch-up),SEGR(Single Event Gate rupture),SEB(Single Event Burnout) がある。このうち SEL,SEGR,SEB は回路がショートし使えなく現象であるが、TGC の設置される領域ではこれらの強烈な SEE 現象は起こらないとされている [15,16]。TGC の設置され

る領域の電子デバイスへの影響で考慮すべきものは、SEE のうち SEU(Single Event Upset) である。集積回路では内部に流れる電流量は小さいので、放射線粒子の通過によって生じる電荷量が、その集積回路の電荷量と同程度になり集積回路が誤動作を起こすことがある [18]。特にメモリ素子などで記憶していた情報が反転することがあり、これを SEU と呼ぶ。TGC の領域での中性子量および SEU の断面積は表 3.4 のようにまとめられており [17]、SEU の頻度を求めることができる。

表 3.4: TGC における中性子量と SEU の頻度

中性子エネルギー	中性子量 (kHz/cm^2)	断面積 (cm^2/bit)	SEU 頻度 ($bit^{-1}sec^{-1}$)
$0.1 \sim 1keV$	1×10^{1}	2×10^{-20}	2×10^{-16}
$1 \sim 10keV$	5×10^{-1}	2×10^{-20}	1×10^{-17}
$10 \sim 100keV$	5×10^{-1}	2×10^{-20}	1×10^{-17}
$0.1 \sim 1MeV$	5×10^{-1}	2×10^{-20}	1×10^{-17}
$1 \sim 10MeV$	1×10^{-1}	1×10^{-13}	1×10^{-11}
$10 \sim 100MeV$	1×10^{-1}	2.0×10^{-13}	2×10^{-11}
$0.1 \sim 1GeV$	1×10^{-1}	2.0×10^{-13}	2×10^{-11}
TOTAL			5×10^{-11}

これをもとに、TGC エレクトロニクスで用いられる ASIC 上で起こる SEU の頻度を見積もってみる。Patch Panel ASIC, Slave Board ASIC, High-pT ASIC にはそれぞれ 1 チップあたり 54bit, 843bit, 24bit のパラメータ設定用のレジスタが搭載されており、これをもとに SEU 頻度を見積もると

- Patch Panel ASIC
  $54(bit/chip) \times 20000(chip) \times 86400(sec/day) \times 5 \times 10^{-11}(upset/bitsec)$
  =4.7(upset/day)

- Slave Board ASIC
  $843(bit/chip) \times 5000(chip) \times 86400(sec/day) \times 5 \times 10^{-11}(upset/bitsec)$
  =18.2(upset/day)

- High-pT ASIC
  $24(bit/chip) \times 1000(chip) \times 86400(sec/day) \times 5 \times 10^{-11}(upset/bitsec)$
  =0.1(upset/day)

となる。実験中に各 ASIC で SEU が起こると、その時点で処理している信号に影響が及ぶだけではなく、それ以降の全ての信号の処理に影響が及ぶ。このため各 ASIC のパラメータ設定用のレジスタには、多数決論理回路を搭載するなどの工夫をしている。これに関しては 5 章で詳細を述べる。

第4章　ASIC開発

ATLAS 実験における、TGC ミューオントリガーシステムの信号読み出し総チャンネル数は 30 万以上になる。またこれら信号の処理を行なうエレクトロニクスは、強い放射線環境下に設置されるので、SEU などに積極的に対処することのできる ASIC(Application Specific Integrated Circuit) で実現されるのが望ましい。ここでは次章からの、TGC の信号処理用 ASIC の説明の前に ASIC の設計、開発の手法に関して述べる。

4.1　CMOS プロセス

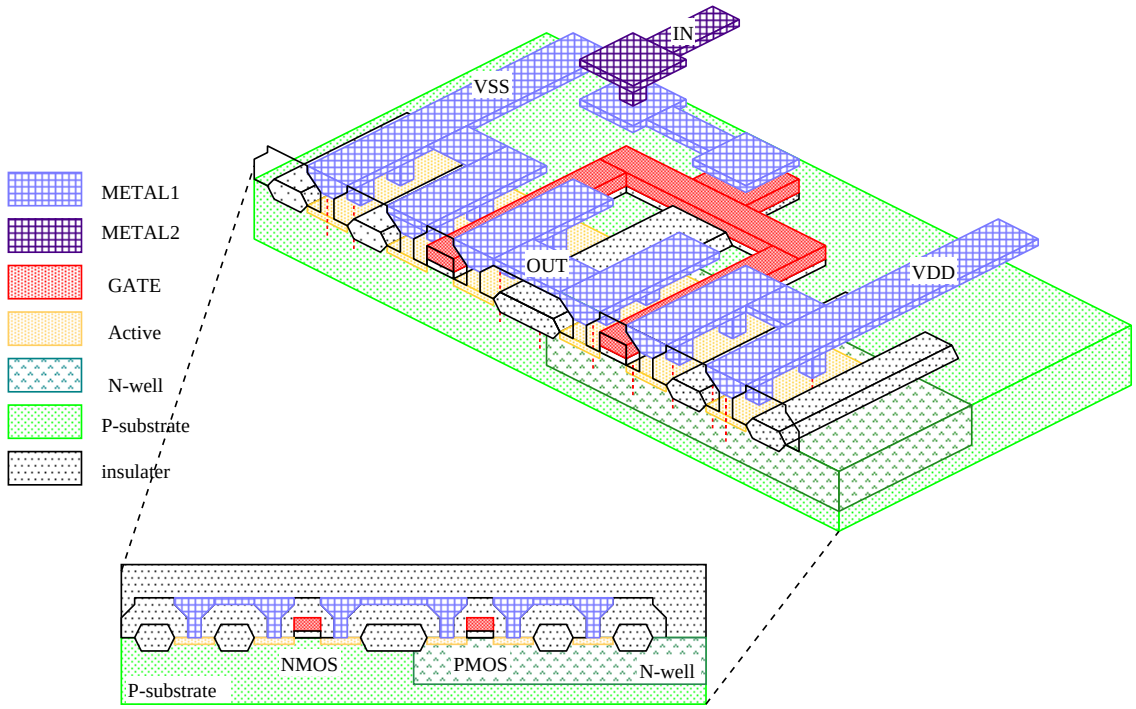


図 4.1: CMOS の断面図

図 4.1 に CMOS(complementary metal oxide semiconductor) プロセスの摸式図を示す。CMOS は導体、絶縁体、拡散層、ポリシシコン、などのトランジスタを形成する物質を何層か重ねてつくられる。トランジスタには NMOS と PMOS の 2 種類あり、NMOS は p 型の基板上 (p-substrate) に 2 つの n 型の拡散層をつくり、その間にゲートと呼ばれる絶縁体をはさんだポリシリコンの電極を配置した構造になる。また PMOS は n 型の基板 (通常は N-well) に 2 つの p 型の拡散層をつくり、その間にゲートを配置した構造になる。またゲートの間隔はそのプロセスの構造の細かさ、あるいは性能を表す目安になり、今回の ASIC の設計に用いるプロセスの場合は、ゲート間隔は $0.6 \sim 0.35 \mu m$ になる。

4.2 回路の表現

現在 IC の設計は主にソフトウェアを通して行なわれる。ソフトウェアで回路を表現する方法として用途に応じて以下のような記述の仕方がある。

- HDL(hardware discription languege) コード (図 4.2(i))
  デジタル回路の記述に使われる言語で verilog HDL と VHDL の 2 種類が主流である。今回の設計では全て verilog HDL を用いている。verilog HDL は c 言語や fortrun のような抽象度の高い言語で、サブルーチンに対応する module があり、変数を定義するように wire(ワイヤ) や reg(レジスタ) を定義する。また各種の論理演算や、if 文や for 文による条件分岐、繰り返しの記述が可能である。デジタル回路の大部分はこの verilog HDL で記述され、またその回路をテストするために入出力部の回路を記述し HDL シュミレータにかけその回路の動作を波形で確認することができる。

- ネットリスト (ゲートレベル記述) (図 4.2(ii))
  ネットリストは、製造テクノロジーに応じて用意されるライブラリを基に構成するものである。ライブラリには各種の論理回路やフリップフロップなどの機能と、モジュールや入出力部の名前が記述されており、ネットリストには、使用するモジュールとそれらを結ぶワイヤを記述する。またネットリストも HDL 記述と同様に、HDL シュミレーターにより動作を確認することが出来る。

- Schematic(回路図) (図 4.2(iii))
  Schematic による記述は主にアナログ回路の記述に用いられる。単体の CMOS を基に記述する。Schematic により記述された回路は、アナログ回路シュミレーターの SPICE により、その回路の動作を確認することができる。Schematic 記述によるアナログ回路シュミレーションには W や L の[1] トランジスタのパラメータは考慮されるが、配線容量や配線抵抗などの影響は考慮されていない。

- レイアウト (図 4.2(iv))
  実際の IC を構成する金属配線層や、トランジスタを構成するゲートポリシリコンや、WELL を 2 次元の図形で記述したもの。最終的に IC は、アナログ、デジタル全てをこの形式で記述することになる。またレイアウトをもとにすれば配線容量や配線抵抗などを考慮したアナログ回路シュミレーションが可能になる。

次節では、これらの回路表現を使った ASIC 設計の手順を簡単に述べる。

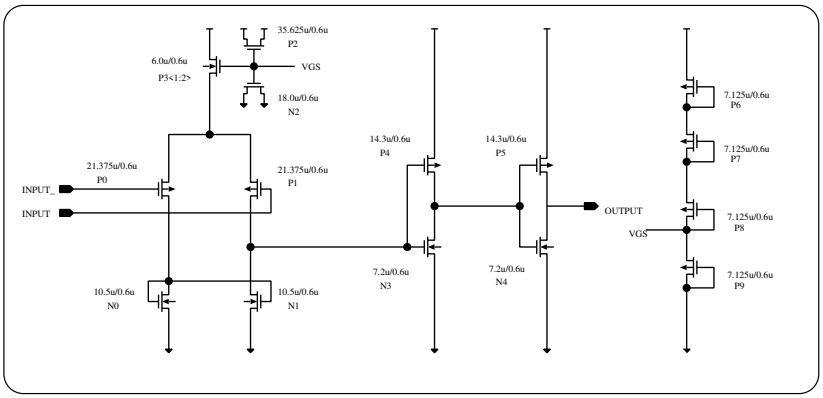
[1] W はトランジスタの電流がながれる部分の幅で、電流量はこの値に比例する。L はプロセスによって固定されているゲートの間隔で今回の場合は 0.6μm か 0.35μm である。

```
module seu_register(CLK, TCK, TDI, TDO_SEU, TRST, tap_state,
   SEU_count_RESET, SEU_mode, SEU_TOTAL);

input CLK, TCK, TDI, TRST;
input [3:0]tap_state, SEU_mode;
output TDO_SEU;
input SEU_count_RESET, SEU_TOTAL;
reg TDO_SEU;
reg SEU_count;
reg shift_reg;
reg condition;

always @(negedge SEU_count_RESET or posedge SEU_TOTAL)
 if(!SEU_count_RESET)
  SEU_count <= 1'b0;
 else SEU_count <= 1'b1;

always @(posedge TCK)
 if(SEU_mode[3] & (tap_state == Shift_DR))
  shift_reg <= TDI;
 else if(SEU_mode[3] & (tap_state == Capture_DR))
  shift_reg <= SEU_count;

always @(negedge TCK)
 if(SEU_mode[3])
  TDO_SEU <= shift_reg;
endmodule
```

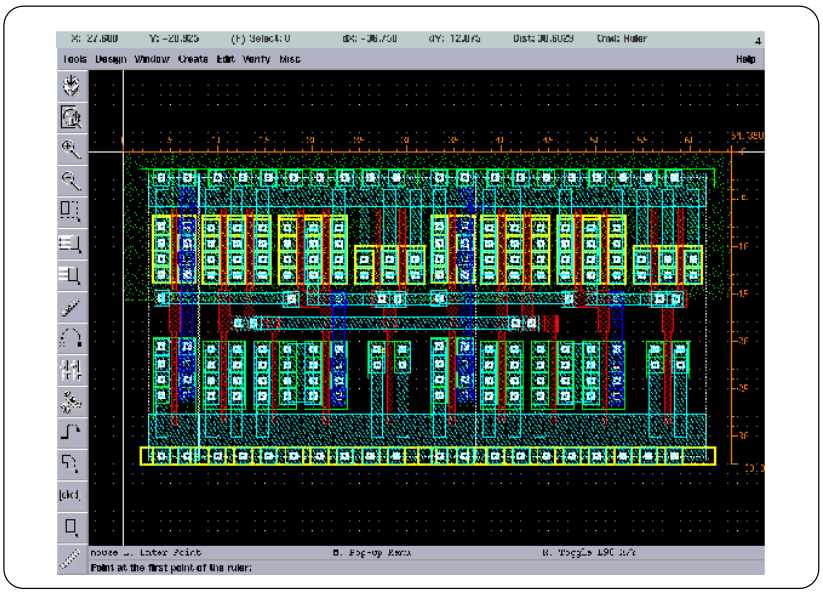
(i)sample file in verilog HDL

```
module dumvdffr(CLK,D,QV,SEUF,XR,XS);

input CLK,D,XR,XS;
output QV,SEUF;
wire Q1,Q2,Q3,VL1,VL2,VL3,SL1,SL2,SL3;

dfbrb1 reg1 ( .cp(CLK), .d(D), .q(Q1), .cdn(XR), .sdn(XS));
dfbrb1 reg2 ( .cp(CLK), .d(D), .q(Q2), .cdn(XR), .sdn(XS));
dfbrb1 reg3 ( .cp(CLK), .d(D), .q(Q3), .cdn(XR), .sdn(XS));

nd02d0 VOTEL1 ( .a1(Q1), .a2(Q2), .zn(VL1) );
nd02d0 VOTEL2 ( .a1(Q2), .a2(Q3), .zn(VL2) );
nd02d0 VOTEL3 ( .a1(Q3), .a2(Q1), .zn(VL3) );
nd03d0 VOTEL0 ( .a1(VL1), .a2(VL2), .a3(VL3), .zn(QV) );

nd12d0 SEUL1 ( .a1(Q1), .a2(Q2), .zn(SL1) );
nd12d0 SEUL2 ( .a1(Q2), .a2(Q3), .zn(SL2) );
nd12d0 SEUL3 ( .a1(Q3), .a2(Q1), .zn(SL3) );
nd03d0 SEUL0 ( .a1(SL1), .a2(SL2), .a3(SL3), .zn(SEUF) );

endmodule
```

(ii)sample file of netlist (gate level description)



(iii)sample of chematic diagram



(iv)sample of layout

図 4.2: 回路の表現

4.3 開発の流れ

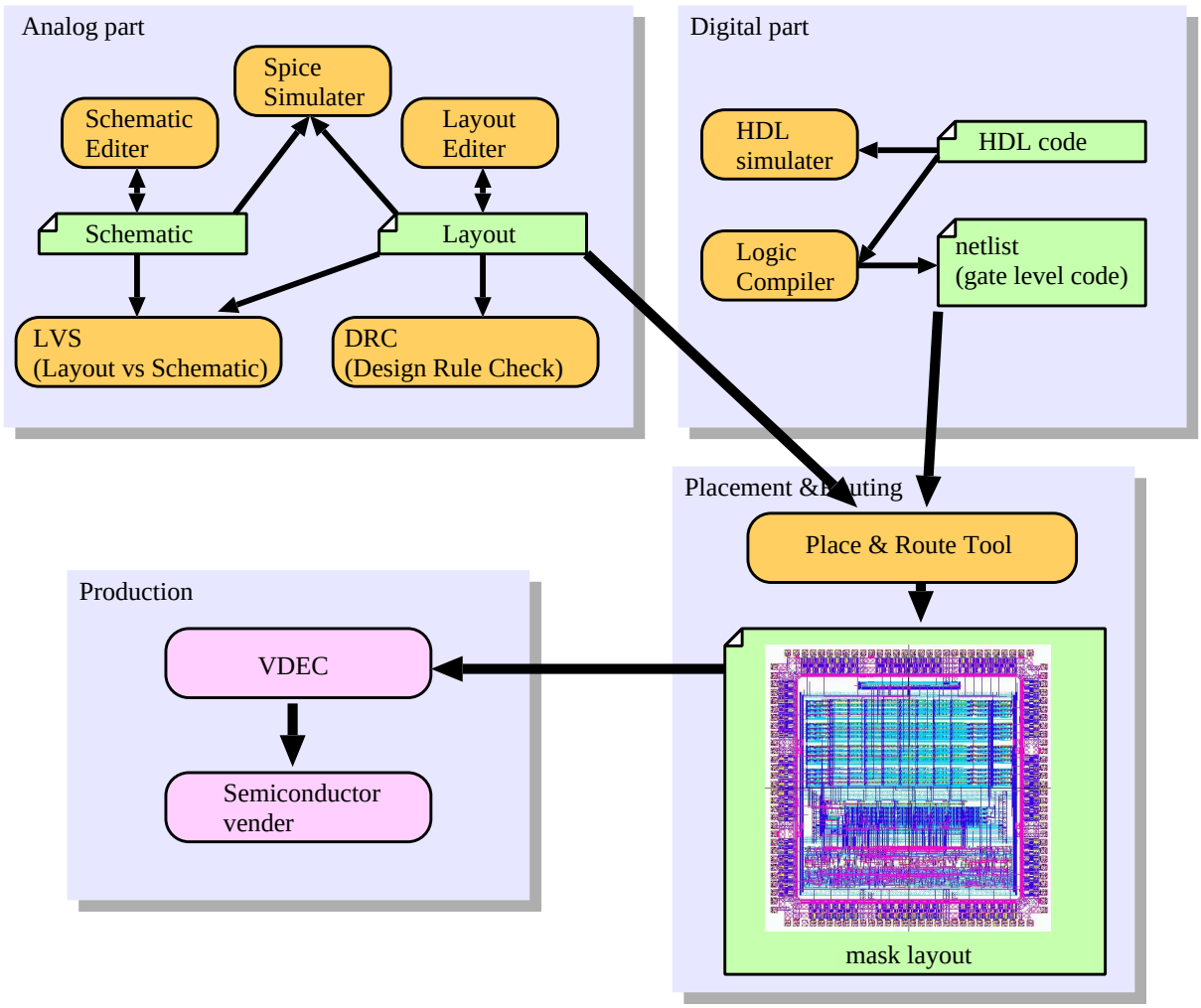


図 4.3: ASIC 開発の流れ

今回の ASIC 開発は VDEC (東京大学大規模集積システム設計教育センター) を通して行なわれた。VDEC は、VLSI の設計教育を支援するための全国共同利用センターで、研究、教育を目的とした IC の開発であれば、開発に必要なツール、データの提供を受けることが出来る。ここでは ASIC の開発の流れを簡単に説明する。

ASIC の開発は図 4.3 に示すように、アナログ部、デジタル部で各々設計、動作の検証を行ない、最後に配置配線を行なってマスクレイアウトを作り、これを半導体ベンダーに渡すという流れで行なわれる。以下に各過程に関して概要を述べる。

デジタル回路部の設計 (論理シミュレーション、論理合成)

デジタル回路では、まず verilogHDL などで記述し、論理シミュレーション、論理合成を行ない回路を設計していく。通常の IC をテストするときはまずチップをテスタにセットし、テストパターンを入力し、その IC の動作を測定するということを行なう。ASIC の開発時にも、これと同様のことを計算機上で模擬的に行なう。論理シミュレーションは、デジタル回路に関してこれを

行なうもので、電圧値を 1 と 0 に 2 値化し、高速に大規模な回路の検証が出来る。シュミレーションをするには回路を記述したコードの他に、回路に入力するパターンを記述したコードを用意する必要がある。テストする回路は verilogHDL などの抽象的なコード、及びネットリストで、入力パターンに対する出力の波形を確認することができ、論理やタイミングの検証ができる。論理シミュレーションの後には、論理合成を行なう。

論理合成とは、抽象度の高い HDL コードを実際の回路素子に対応させる変換のことであり、ロジックコンパイラーなどのツールを用いて行なう。またライブラリには各種の論理素子、フリップフロップなどの情報が含まれている。また抽象度の高い HDL 記述の場合、論理合成が出来ないこともあり得るので、HDL 記述の段階から対応する回路を考える必要がある。

アナログ回路設計 (レイアウト、回路図、**SPICE** シミュレーション)

アナログ回路の設計は、レイアウトエディタと回路図エディタを使って行なう。レイアウトはトランジスタを構成するゲートやメタル (配線) から、回路を記述するもので、記述には長い時間を要するが、性能の良い IC を設計することが出来る。またこのレイアウトや回路図の動作の検証は、SPICE などのアナログ回路シミュレータを使って行なう。

SPICE シミュレーションは、回路図 (Schematic) やレイアウトのデータから、半導体のパラメータ、抵抗、容量を考慮して行なうシミュレーションである。回路図のデータの場合は、記述した素子のパラメータのみを使ってシミュレーションを行なう。一方レイアウトの場合は素子のパラメータに加え配線の持つ抵抗、容量、またトランジスタの各部にある寄生容量、寄生抵抗を考慮したシミュレーションが可能である。

配置配線

配置配線はデジタル回路の設計で作ったネットリストや、アナログ回路の設計で記述したレイアウト、また半導体ベンダーから提供されるライブラリをもとに、セルを 2 次元上に配置しその間を配線していく処理である。最終的に半導体ベンダーに渡す IC のマスクレイアウトは配置配線で作る。

デザインの検証

設計の最終段階、設計過程において、DRC、LVS といったデザインの検証を行なう。

DRC(Design Rule Check) は、レイアウトに対して行なう回路の検証で、製造テクノロジーに応じて用意されるデザインルールを満たしているかを確認すものである。主にトランジスタの各要素、配線などの最小間隔、隣接する素子との間隔に関するルールがある。

LVS(Layout vs Schematic) はレイアウトと回路図 (またはネットリスト) が、対応していることを確認する検証である。レイアウトの編集により手配線で回路を設計する場合、レイアウトと同時に対応する回路図を記述し、それらが等価であることを確認する必要がある。

第5章　JTAGプロトコル

ここではPatch Panel ASIC, Slave Board ASIC, High-pT ASICにおいて、ICの制御用プロトコルとして共通して用いられる、JTAGプロトコルに関して述べる。

JTAG(Joint Test Action Group)(正式名称 IEEE1149.1)は、電子デバイスの内部ロジックや、外部配線のテストを行なうために開発されたものであり、近年の電子デバイスの大規模化により、外部からプローブをあてるなどといった従来型の方法では、デバイス自身やその周辺回路のテストに、膨大な時間と労力が必要になってきた。JTAGは製作するデバイスに、予めテスト用回路を組み込むことでこれを解決する。デバイスの通常の動作の時には、このテスト用回路を意識する必要は無く、またテスト用回路は、通常の回路に影響を与えることもない。JTAGは4本(又は5本)の信号線により制御され、またテスト用信号線は、テスト対象のデバイス間で直列に接続することが可能である。

TGCエレクトロニクスでは、JTAGのオプション機能であるユーザー定義レジスタを使い、信号処理用のASIC内部の様々なパラメータの設定を行なう。TGCトリガーシステムは全体で数万個のASICを使い、TGCからの信号の処理を行なうため、JTAGのような配線が少なくて済むプロトコルが有効である。ここでは標準のJTAGプロトコルに加えて、TGCエレクトロニクス用のオプションレジスタなどを含むJTAGの仕組みを説明する。またパラメータの設定用のJTAGのレジスタにはSEU対策として搭載される多数決論理回路に関しても述べる。

5.1　JTAGプロトコルの仕組み

5.1.1　概要

図5.1にJTAGの構造の概要を示す。JTAGは主に各種のレジスタと、それを制御するtap_controler、instruction decoder及び、どのレジスタの値を読み出すかを選択するTDO selectorからなる。tap_controlerはtap_state(4bit)を各部のレジスタに供給する。tap_stateにはUpdate(レジスタに値をロードする。), Shift(レジスタの値を次のレジスタにシフトする。), Capture(値をレジスタに読み込む)などがある。またInstruction decoderはInstruction registerの値に応じて、各部のレジスタに命令(mode)を供給する。命令はレジスタ毎に書き込み命令、読み込み命令などがある。全てのJTAGレジスタはtap_stateとInstructionの命令の組み合わせに応じて動作することになる。

5.1.2　JTAGの制御線

JTAGの制御線は以下のような4本(オプションを含めると5本)がある。

- TCK(Test Clock)
  TCKはJTAG各部を制御するためのクロックである。TGCエレクトロニクスでは10MHz

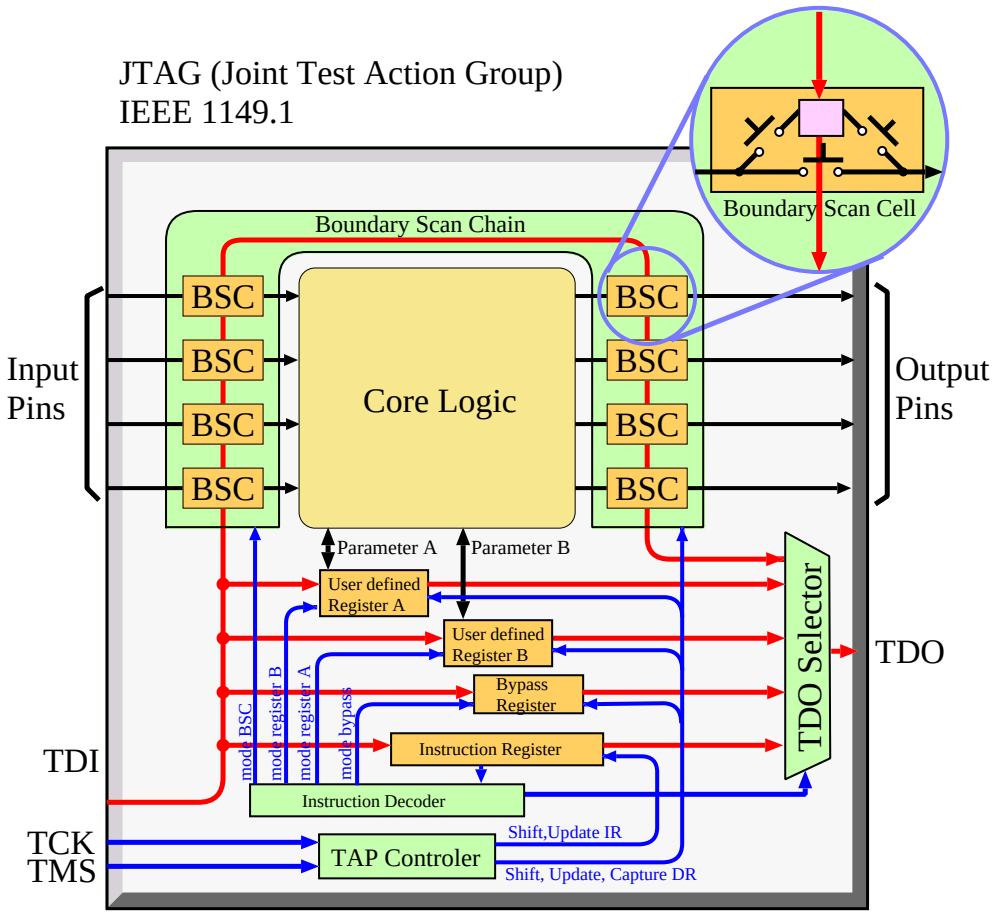


図 5.1: JTAG プロトコルの概要

程度の TCK を使う。TCK の上がりエッジでは各部レジスタのデータ読みとり用のフリップフロップ (Shift register) がデータを取り込み、また tap_controloer の状態が遷移する。下りエッジでは各部レジスタのデータロード用レジスタ (Update register) がデータをロードする。

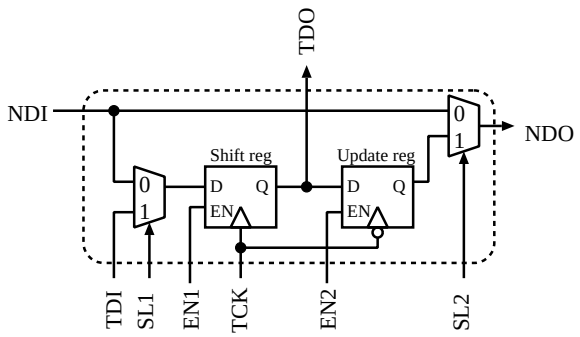
- TMS(Test Mode Select)
  Tap state の遷移先を決める。TCK の上がりエッジで読まれる。

- TDI(Test Data Input)
  各種データレジスタ、バウンダリースキャンセルに設定する値は TDI から入力する。またこれらのデータは TCK の上がりエッジで読み込まれる。

- TDO(Test Data Output)
  各種データレジスタ、バウンダリースキャンセルに設定する値は TDO から読み出す。TDO の値は TCK の下りエッジで変わる。

- TRST(Test Reset)
  tap_state を 1 クロック (TCK) で Test logic reset にするオプションの制御線である。TGC エレクトロニクスでは全ての ASIC にこれを搭載する。

5.1.3 バウンダリースキャンセル (BSC)



SL1 = mBSC & Shift_DR
EN1 = mBSC & (Shift DR + CaptureDR)
EN2 = mBSC & UpdateDR
SL2= {modeEXTEST (output side) / modeINTEST (input side)}

mBSC=mEXTEST
+ mINTEST
+ mSAMPLE/PRELOAD

図 5.2: boundary scan cell

バウンダリースキャンセルは、テスト対象の回路の IO の部分に取り付ける回路で、これを介して、入力側の BSC から IC の内部に任意のパターンをロードし、その結果を出力側の BSC で読み込む INTEST, また入力側のパターンを読み出したり、出力側に任意のパターンを出力する EXTEST が可能である。またそれらのパターンを BSC に送ったり、読み出したりするために SAMPLE/PRELOAD 命令がある。図 5.2 にバウンダリースキャンセルの回路図を示す。

BSC は入力側と出力側でほぼ同じ回路であるが、SL2 に入る信号が異なる。入力側の BSC では SL2 は mINTEST(INTEST モード) が接続さ、出力側の BSC には SL2 に mEXTEST(EXTEST モード) が接続される。通常の動作時には、図のセレクター SL1,SL2 がそれぞれ 0 になり、通常動作用の信号は、NDI から NDO へと何の影響も受けずに通り抜ける。SAMPLE/PRELOAD モードなどの時に、TDI からのテストデータを BSC にシフトさせるときには、SL1、EN1 が 1 になり、鎖状の BSC 内の shift reg がシフトレジスターになり、TCK に同期して動作する。NDI の値を BSC に取り込む場合は、EL1 を 0 に EN1 を 1 にして TCK の上がりエッジで取り込む。NDO にデータをロードするときは、shift reg まで予めデータをロードしておき、EN2,SL2 を 1 にして TCK の下りエッジでロードする。

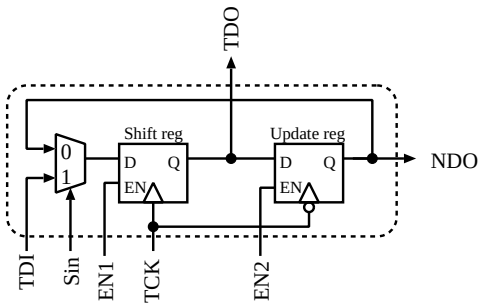
また、これらの動作が具体的にどのようなタイミングで行なわれるかは、付録 A で述べる。また BSC は実験中、パラメータを保持する必要がないので、SEU 対策の多数決回路は搭載していない。

5.1.4 データレジスタ

ここではインストラクションレジスタ、ユーザー定義レジスタの RW モード (読み書き両用のレジスタ) と R モード (読みだし専用レジスタ) に関して述べる。

ユーザー定義レジスタ (**RW** モード)

TGC エレクトロニクスにおいて、RW モードのユーザー定義レジスタは、様々なパラメータの設定に用いられる。回路図を図 5.3 に示す。このレジスタは実験中、放射線環境中においても設定されたパラメータを、正しく保持する必要があるので、図中の Update reg には多数決論理回路を用いる。またレジスタの値をリセット信号によって、初期化することも可能である。これらに関しては後述する。



SL1 = ShiftDR
EN1 = (mRead + mWrite)
& (ShiftDR + CaptureDR)
EN2 = mWrite & UpdateDR

} user defined register (RW)

SL1 = Shift IR
EN1 = (ShiftDR + CaptureDR)
EN2 = UpdateIR

} Instruction register (RW)

図 5.3: データレジスタ (RW モードユーザーレジスタ、インストラクションレジスタ)

さらに、このレジスタには、書き込んだ値を、確認のために読み出せるよう読みだしモードも搭載した。回路図の NDO を入力部に返しているのは、このためである。

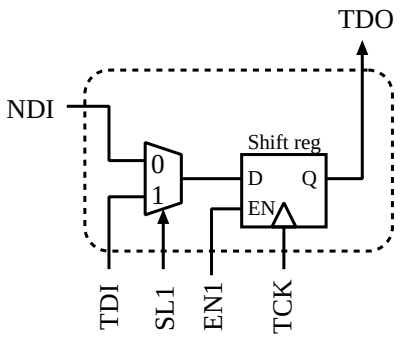
データの読み書き、シフトは BSC の場合と同じような経路で行なわれる。また命令 (モード) は Write mode と Read mode があり、これらはレジスタ毎にインストラクションデコーダーから 2bit づつ供給される。

インストラクションレジスタ

インストラクションレジスタは 8bit のレジスタで、その値はインストラクションデコーダでデコードされ、JTAG のレジスタの各部に、命令 (モード) を供給するために用いられる。回路は図 5.3 に示すように、RW モードのユーザー定義レジスタと同じであるが、動作に必要な命令が他のレジスタと異なり、インストラクションデコーダからの命令なしに、tap_state を受けるだけで動作する。

このレジスタにも多数決回路を用いている。また初期値も設定可能で初期値は 1111_1111 にし、これはバイパスモードに対応する。

ユーザー定義レジスタ (R モード)



SL1 = mRead & ShiftDR
EN1 = mRead & (ShiftDR + CaptureDR)

図 5.4: データレジスタ (R モードユーザーレジスタ)

R モードのユーザー定義レジスタは、データ読み込み専用のレジスタで、SEU のフラッグの検

出や、基盤上のパラメータを読み出すのに用いられる。回路図は図 5.4 に示すように簡単な構造で、NDI から CaptureDR のときに、TCK の上がりエッジで NDI の値を取り込む。

5.1.5 バイパスレジスタ

バイパスレジスタは複数のデバイス (ASIC) の TDI、TDO を直列に接続したときに、各デバイスの、レジスタチェーンを介することなく、テストデータをシフトさせるための 1bit のレジスタである。1 つ D フリップフロップで構成され、イネーブルに BYPASS mode が入っている。TCK の上がりエッジで動作する。

5.1.6 Tap Controler

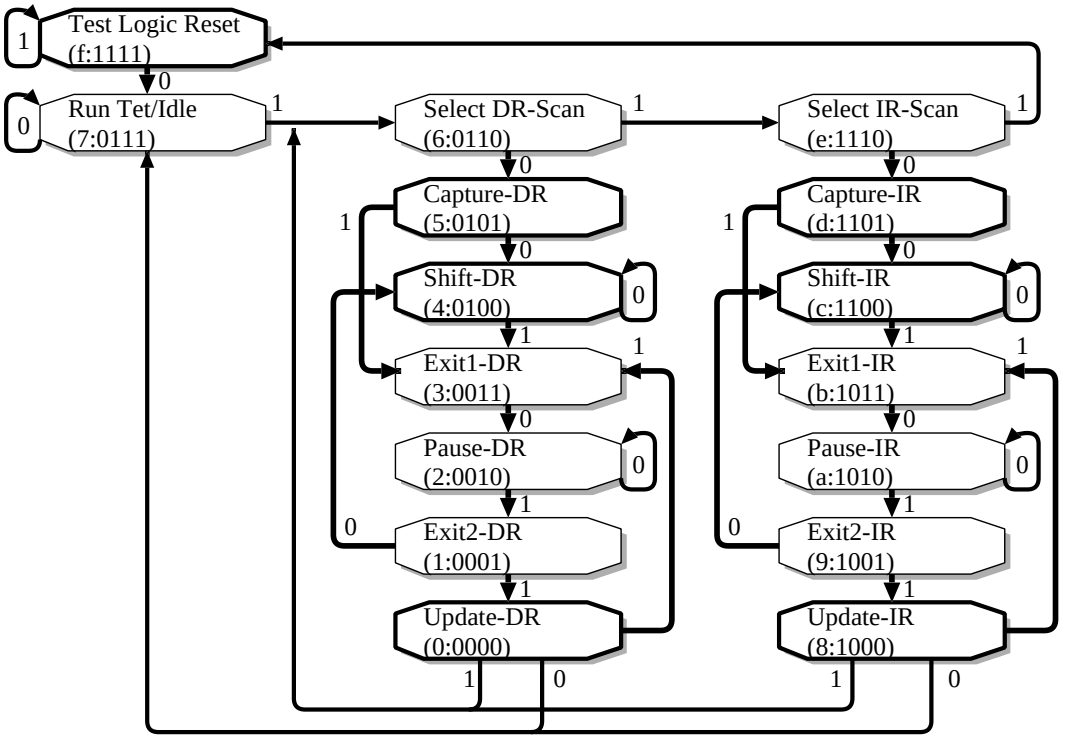


図 5.5: tap_state の遷移図

Tap controler は TMS、TCK を受け (オプションで TRST も受ける)4bit の tap state を出力するステートマシーンである。遷移は TMS に応じて行なわれ、TCK の上がりエッジで動作する。図 5.5 に tap state の遷移図を示す。(図中の各ステート名の下に書いた数字は Patch Panel ASIC,Slave Board ASIC, Hi-pT ASIC で共通に用いた 4bit の tap state の、値の 16 進表示と 2 進表示である。この値は ASIC 外部で使われることはないので、デバイス間で異なっても差し支えない。)

これらのステートのうち、重要なのは以下の 7 つで、それ以外のステートはこの 7 つの状態に遷移するための中間状態と考えて良い。

- Test logic reset
  ステートマシンの起点の状態。ここから各状態に遷移するステートマシンのいずれの状態か

らも TMS を 1 にして TCK を 5 クロック受けるとここに戻る。またオプションの TRST (負論理リセット) でもここに戻る。

- Capture DR (Capture IR)
  各レジスタにおいて NDI の信号 (RW モードのユーザーレジスタの場合は NDO の値) を shift reg に読み込む。BSC では ASIC 外部の入力、あるいは ASIC の出力を読み込む。読み込みは TCK の上がりエッジで行なわれる。

- Shift DR (Shift IR)
  各レジスタの Shift reg の値の TDO 側の隣のレジスタにシフトさせる。シフトは TCK の上がりエッジで行なわれる。シフトは tap state が Shift DR になった次の TCK の上がりエッジで行なわれる。各レジスタにデータをシフトさせるには TMS を 0 にしたまま必要な回数だけシフトさせる。実際に JTAG を動作させるときにはこの回数に十分注意する必要がある。

- Update DR (Update IR)
  各レジスタの Shift reg の値を Update reg にラッチする。TCK の下りエッジで行なう。

5.1.7 Instruction Decoder

インストラクションデコーダはインストラクションレジスタの値を受け、それに応じて各レジスタに命令を出す。命令には 5.1 に示すようなパブリック命令と、ユーザーが適宜設定可能な命令がある。TGC エレクトロニクスで使う ASIC ではパブリック命令の内、EXTEST,SAMPLE/PRELOAD, INTEST, BYPASS(これらはいずれも 1bit でバウンダリースキャン用の命令である。) を用いる。また各種のユーザーレジスタに対してそれぞれ命令を定義した。

表 5.1: JTAG 命令

インストラクション名	種類	ビット列
EXTEST	必須	0000_0000
SAMPLE/PRELOAD	必須	0000_0001
INTEST	オプション	0000_0010
RUNBIST	オプション	0000_0011
ID CODE	オプション	0000_0100
USER CODE	オプション	0000_0101
CLAMP	オプション	0000_0110
HIGHZ	オプション	0000_0111
BYPASS	必須	1111_1111

- EXTEST 命令
  EXTEST 命令は、入力側の BSC では入力ピンの状態を読みとり、出力側の BSC では、任意のパターンを出力ピンにロードする。このときロードしたいパターンは、あらかじめ SAMPLE/PRELOAD 命令によって設定しておく必要がある。また入力ピンからのデータの取り込みは Capture DR 状態のときに、TCK の上がりエッジで行なわれる。また出力ピンへのデータのロードは、Update DR 状態のときに、TCK の下がりエッジで行なわれる。

EXTEST は IC 外部の基板の検査や、デバイスの相互接続のテストに用いられる。TGC エレクトロニクスの場合は、ASD からのテスト信号を PP(Patch Panel ASIC) の BSC で読むことで、その間の接続を確認できる。また PP、SLB(Slave Board ASIC) 間で行なえば、その間の配線をテストすることが出来る。

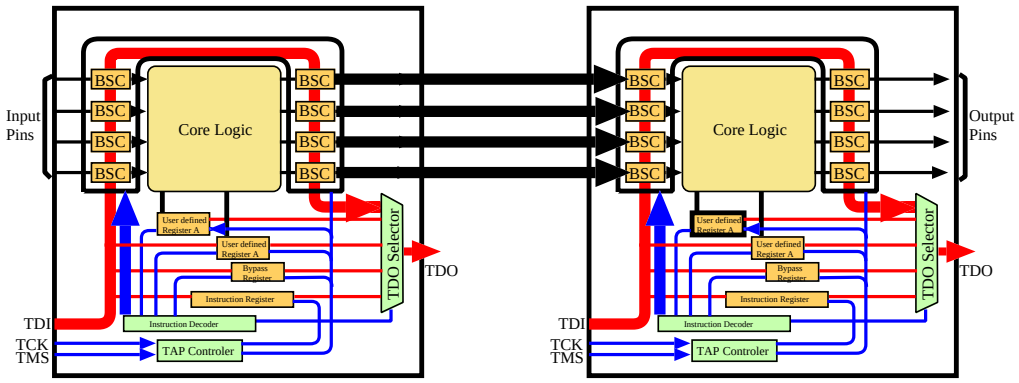


図 5.6: EXTEST を使った相互接続テスト

- SAMPLE/PRELOAD 命令
  デバイス本来の機能に全く影響を与えずに、データを BSC にシフトさせる命令。デバイスが通常の動作をしているときでも、その動作に影響を与えることなく、入力ピンの状態を読み出すことが出来る。また EXTEST,INTEST の際のデータを各 BSC の shift reg にまでセットするために用いる。

- INTEST 命令
  デバイスの内部をテストするための、オプション命令である。デバイスの内部回路の入力部に、任意のパターンをロードすることが出来て、その際にデバイスの出力をみることで、デバイスの内部回路のテストが出来る。

- BYPASS
  デバイスの TDI,TDO の間に 1bit のレジスタを挿入し、長い BSC などを通さずに次のデバイスにテストデータを送るための命令。

- ユーザーレジスタ用命令
  各レジスタに対して write 命令、read 命令がある。インストラクションコードは xxxx_xx01 が write 命令、xxxx_xx10 が read 命令にあてる。上位 6bit はレジスタ毎に設定する。

5.1.8 TDO selector

図 5.7 に TDO セレクタの回路図の例を示す。TDO セレクタは命令により各レジスタ、BSC からくる TDO 信号の内、動作状態にあるものの TDO 信号を 1 つ選んで出力する。

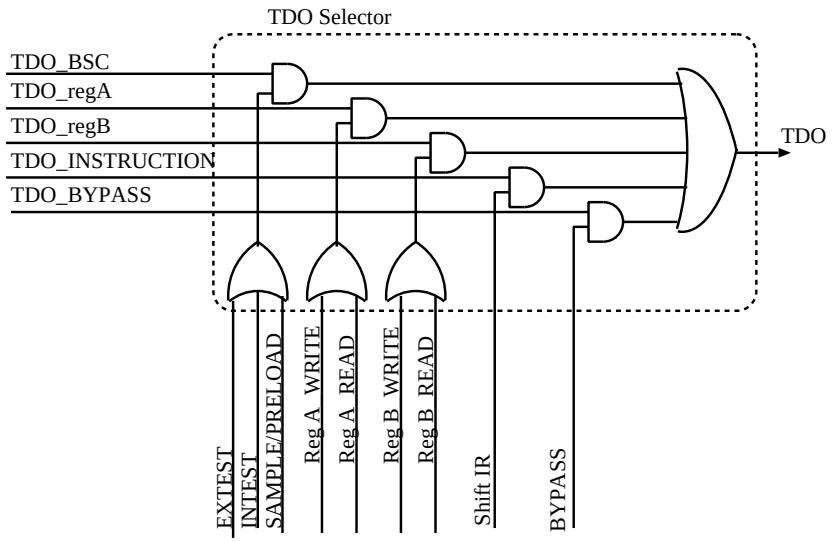


図 5.7: TDO セレクタの回路図の例

5.2 多数決論理回路

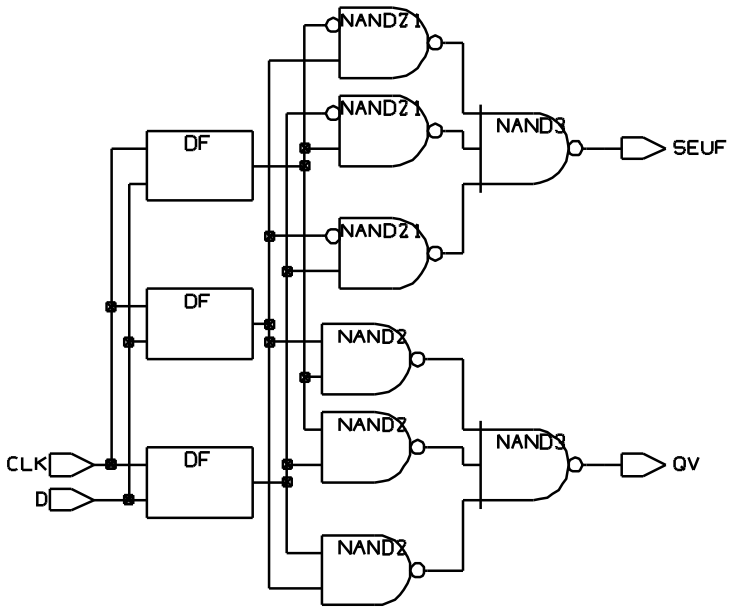


図 5.8: 多数決論理回路の回路図

ここでは放射線の影響で、レジスタの値が反転するという SEU 現象に対処するための回路として用いる、多数決論理回路に関して説明する。3 章でも述べたように、SEU は中性子などの影響により、IC 中のレジスタの値が反転される現象である。TGC エレクトロニクスの ASIC の、パラメータ設定用のレジスタで SEU 現象が起こると、その影響はその時、そこを通過している信号のみではなく、その後の信号処理全てに影響を与えることになる。そこで TGC エレクトロニクスでは、多数決論理回路を用いることでこれに対処している。この回路は IC 内の、様々なパラメータを設定するためのレジスタ、すなわち JTAG の RW モードのユーザー定義レジスタに用いられる。(図 5.3 の回路のうちの Update_reg に用いられる。)

多数決論理回路の回路図を図 5.8 に示す。この回路は、1bit の情報を 3 のレジスタ (D フリップフロップ) に記憶し、その 3 つレジスタの多数決を出力するというものである。これにより、3 つレジスタの内 1 つに SEU 現象が起きて、値が反転したとしても、その多数決論理回路の出力は、正確な値に保持される。また多数決が割れたことを検出する回路も搭載し、その出力は、ASIC 内部の全てのレジスタに関して OR がとられ、読み出すことが出来る。

TGC のエレクトロニクス上では、システム全体で 1 日に十数回の SEU 現象が起こると見積もられているが、同一のレジスタに、2 回以上 SEU が起こる確率は非常に小さいとして、3 つのレジスタで多数決をとることにしている。

5.3 JTAG の HDL コードを自動生成するスクリプト

これまで各 ASIC に搭載される JTAG プロトコルは、それぞれ良く似た回路であるにもかかわらず、全く個別に開発が進められてきた。またそのつど HDL の記述、デバックに多くの時間が費やされてきていた。そこで今回、この開発時間を短縮するため、また確実に動作する JTAG プロトコルを簡単に準備するために、JTAG の verilogHDL コードを自動的に生成するスクリプトを開発した。これは必要なレジスタの、種類、bit 数、名前のリストを作るだけで、その verilogHDL のコードを生成するものである。例えば今回、実際にこのスクリプトを使って開発された、high-pT ASIC 用の JTAG プロトコルの場合、

```
rw    3     DELAY0
rw    3     DELAY1
rw    3     DELAY2
rw    3     DELAY3
rw    3     DELAY4
rw    3     DELAY5
rw    3     DELAY6
rw    3     GLINK
r     1     SEU
```

というファイルを作り、このスクリプトを実行するだけで、verilogHDL の記述が完了する。またその動作も確かめられているものが生成されるので、デバックも簡単である。今後このスクリプトを使うことで、JTAG を含む IC を設計する場合に、その開発期間を大幅に短縮できるものと考えられる。今後、Patch Panel ASIC や Slave Board ASIC、あるいは JRC ASIC に搭載する JTAG も、このスクリプトで生成することができる。

5.4 JTAG プロトコルに関するまとめ

JTAG プロトコルは、Patch Panel ASIC, Slave Board ASIC, High-pT ASIC のパラメータの設定、読みだし、またバウンダリースキャンテストを行なうためのプロトコルで、ここでは標準の JTAG プロトコルに加え、TGC エレクトロニクスで必要になる RW モード、R モードのユーザー定義レジスタを加えた JTAG プロトコルの説明をした。また RW モードのユーザー定義レジスタには、SEU 現象に対処するための多数決論理回路を搭載した。また、これまでの ASIC 開発では JTAG プロトコルの部分は、各 ASIC ごと個別に開発されきており、そのつど長い開発期間を要したが、今回、JTAG の verilogHDL コードを、自動的に生成するスクリプトを開発したことにより、今後の各 ASIC に搭載される JTAG プロトコルの開発期間は、大幅に縮小されるものと期待できる。

第6章　Patch Panel ASICの開発

ここではTGCトリガーシステムにおいてタイミング調整、バンチ識別を行なうPatch Panel ASICに関して述べる。主にその機能と設計、及び試作ICの動作結果について述べる。

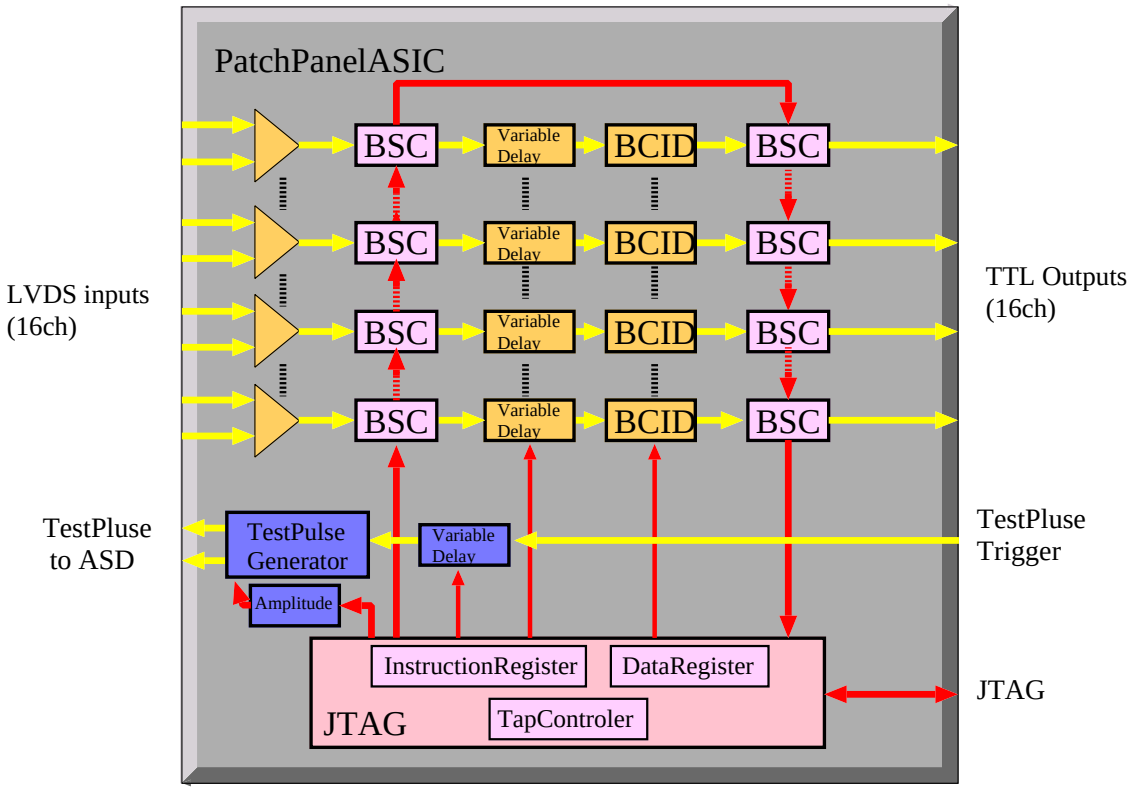


図 6.1: Patch Panel ASIC の機能の概要

6.1　Patch Panel ASICの機能と構成

図6.1にPatch Panel ASICの機能の概要を示す。Patch Panel ASICはASDから信号を受け、タイミング調整を行なった後、LHCクロックと同期をとり、Slave Board ASICに向けTTL信号を出力する部分である。またPatch Panel ASICは1チップで16チャンネル分の処理を行なう。以下に各部の機能の概要を述べる。

- LVDSレシーバー
  ASDからの信号は、LVDS(Low Voltage DifferentialSignaling)レシーバーで受信される。LVDSは低電圧の差動型の信号規格で、ASDとPatch Panel ASIC間のように、数メートルから数十メートル程度の距離があり、送信部、受信部でグランドレベルに若干の差があるよ

うな場合でも、正確に信号の送受信が可能な信号規格である。受信後は TTL 信号に変換され、次段の可変遅延回路に送られる。

- 可変遅延回路 (DLL 回路)
  可変遅延回路はタイミング調整の為の回路で、DLL(Delay Lock Loop) 回路により実現される。遅延の可変の幅は 25nsec で、精度はその 1/32 の 0.78nsec である。DLL 回路はアナログ回路で、これも LVDS レシーバーと同様に、レイアウトレベルから設計を行なう。

- バンチ識別回路 (BCID 回路)
  バンチ識別回路は可変遅延回路からの信号を受け、LHC クロックと同期をとる回路である。これは受けた信号が、どのバンチ衝突によって生成されたものなのかを決定することに相当する。クロック毎に開けるゲートの幅は、基本的には 1CLK 分の 25nsec で、1 クロック分のパルスを出力する。また、入力信号の到着時間の分布が 25nsec を超える場合に対処するため、実効的なゲート幅を、25∼50nsec に拡張できるようにもしている。この場合、拡張したゲートの部分に信号が入ると、バンチ識別回路は 2 クロック幅の信号を出力する。またバンチ識別回路はデジタル回路であるが、信号のタイミング調整に用いられる回路であるため、信号の通過時間のばらつきを極力押えるため、デジタル回路であるにも関わらず、レイアウトレベルから設計を行なっている。

- マスク回路
  TGC 検出器の一部のチャンネルが故障し、信号を出さなくなるか、粒子が来ていないのに信号を出すといったことが起こった時に、その部分のチャンネルの処理を無効にするマスク回路を搭載した。これは BCID 回路の一部として実装され、16 チャンネルそれぞれに対して、マスクの有無の設定ができるように設計した。

- テストパルス回路
  テストパルス回路は ASD に向けテスト信号を出力する回路で、実験前の様々な調整のときに用いられる。またこれらの調整にはタイミング調整も含まれるので、テストパルス回路自身も、その信号を出すタイミングをサブナノ秒の精度で調整することができる。このタイミング調整には、信号のタイミング調整に用いるのと同様の DLL 回路、及び 1 クロック単位で遅延を調節する、荒い遅延回路 (coarse delay) がある。またテストパルス回路はその振幅も可変にした。

- JTAG プロトコル
  JTAG プロトコルは本来 IC のテスト用に用いられるもので、IC の入出力部の直近に BSC(Boundary Scan Cell) を設置し、IC 内外のテスト (バウンダリースキャンテスト) を行なうものであるが、TGC エレクトロニクスでは、このバウンダリースキャンテストに加えて、JTAG プロトコルのオプション機能であるユーザー定義のレジスタを使い、IC のパラメータの設定にも使う。Patch Panel ASIC ではこのユーザー定義レジスタを使って、タイミング調整に使われる可変遅延回路の遅延値の設定や、テストパルスの振幅の設定などに用いる。またこのレジスタには SEU 対策として、多数決論理回路が使われており、また SEU の有無を読み出すことが出来る。また JTAG プロトコルは、完全なデジタル回路で、大部分が HDL 記述及び自動配置配線で設計される。

- ヒューズモニター
  PS ボード上から ASD に電源、グランドを供給する電源にヒューズがあり、ヒューズモニ

ターはこの断線の有無を検出する。またこの断線の有無は JTAG プロトコルで読み出すことが出来る。

6.2 Patch Panel ASIC の設計

Patch Panel ASIC の開発は、4 章で述べたような開発ツールを用いて行なった。使用するプロセスは、ローム社のゲート間隔が 0.6μm のフルカスタムチップで、サイズは 4.5mm×4.5mm で、使用可能なピンの数は 84 である。またライブラリや、設計時に使う SPICE パラメータなどの半導体パラメータは、VDEC から提供を受けたものである。

6.2.1 LVDS レシーバー

LVDS(Low Voltage Differntial Signaling) は、低電圧の差動型の信号規格で、LVDS の送信側、受信側のデバイスでグランドレベルが多少異なる場合でも、信号の送受信が正確に行なわれるのが特徴である。ここでは ASD からの LVDS を受信する LVDS レシーバーに関して述べる。

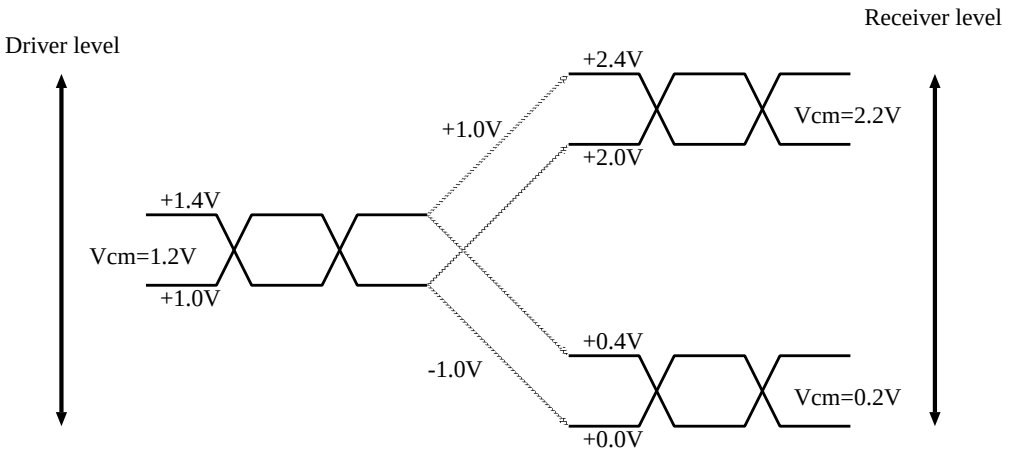


図 6.2: LVDS の信号レベル

図 6.2 に LVDS 信号規格による信号レベルを示す。図のように信号の中心電圧は、1.2V を中心とし、上下 1[V] のシフトを許す、振幅は 400mV を基準にし 200mV まで動作可能である。Patch Panel ASIC の LVDS レシーバーでは、この LVDS 信号を受け、TTL 信号に変換する。

回路図

図 6.3 に LVDS レシーバーの回路図を示す。回路図中の INPUT,INPUT_が LVDS の入力で、OUTPUT が TTL 信号の出力である。またトランジスタ (N0,P0,N1,P1) が差動アンプで、その後の (N3,P3),(N4,P4) が整形用の負論理回路である。また P3 は抵抗の役割をするゲートである。

レイアウト

図 6.4 に LVDS レシーバーのレイアウトを示す。LVDS レシーバーはアナログ回路であるため、デジタル回路では無視できるような寄生容量や寄生抵抗が、回路の動作に影響を与える。そのため LVDS レシーバーはレイアウトレベルから設計する。また Patch Panel ASIC の場合、IC 内で混在しているデジタル回路の動作のために局所的にグランドのレベルが揺らぐことがあり、これが LVDS レシーバーの動作に影響し、ノイズとなりうる。そのため LVDS レシーバーのみ、電源を他の回路と分けて供給するようにして対処している。

シミュレーション

上述のような回路デザインで、入力信号の中心電圧、振幅、周波数、また温度などのパラメータを変えて動作シミュレーションを行ない、その動作を確認している。図 6.5 にそのシミュレーション結果の例を示す。図は LVDS 信号規格の電圧範囲のテスト信号を入力し、その動作を確認したものである。

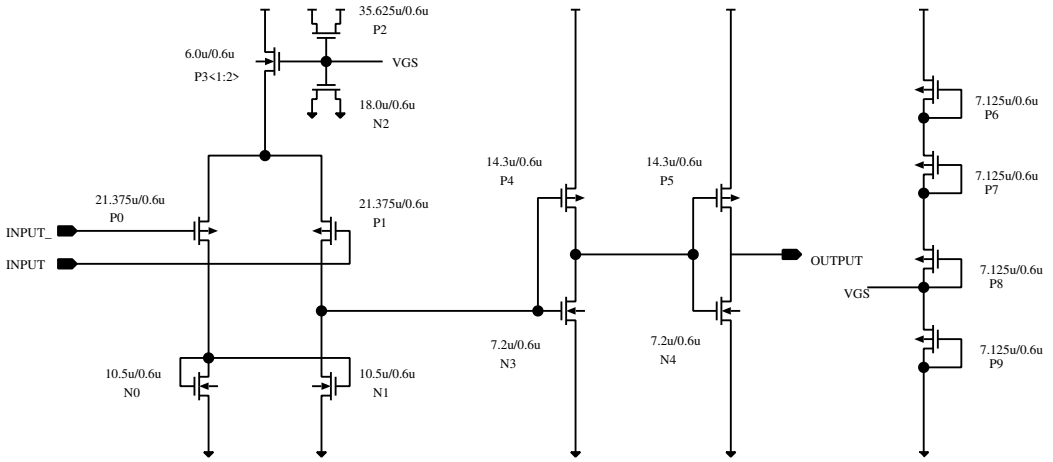


図 6.3: LVDS レシーバーの回路図

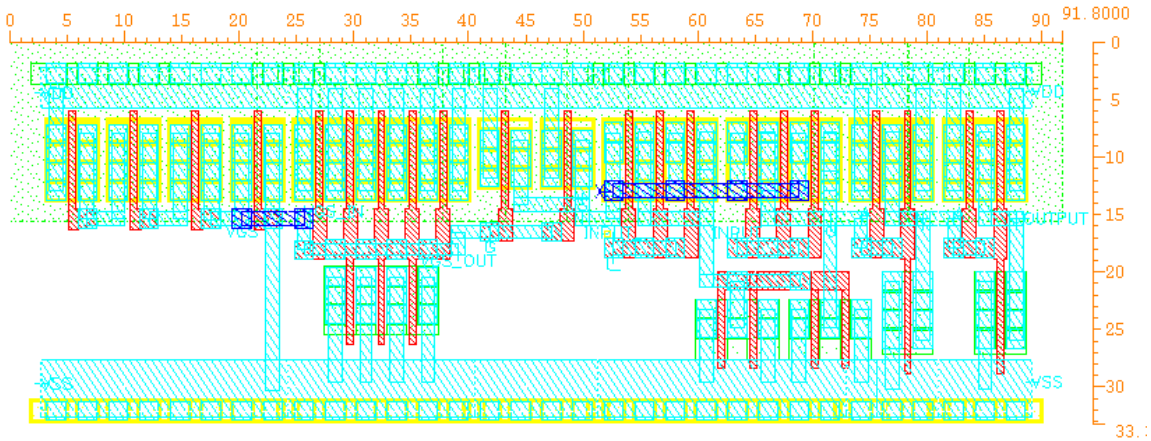


図 6.4: LVDS レシーバーのレイアウト

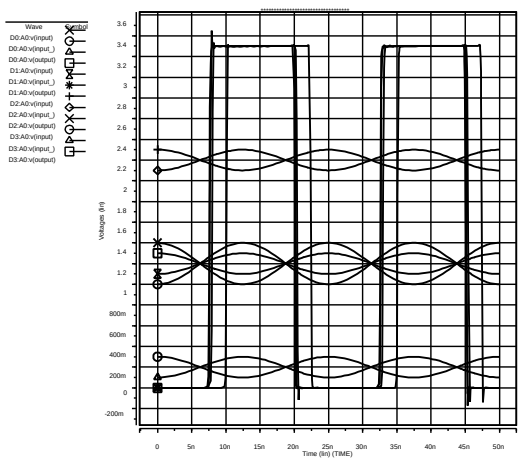


図 6.5: LVDS レシーバーの SPICE シュミレーションの結果
仕様の範囲の offset 電圧、振幅に関して出力の応答を測定した。

6.2.2 遅延回路 (DLL)

DLL(Delay Lock Loop) 回路は、LVDS レシーバーや、テストパルストリガーの TTL 信号を受け、0 から 25nsec までで 32 段階の可変の遅延を与えて出力する回路である。またこの 32 段の遅延のうち、どれを出力するかは JTAG のレジスターによって設定される。

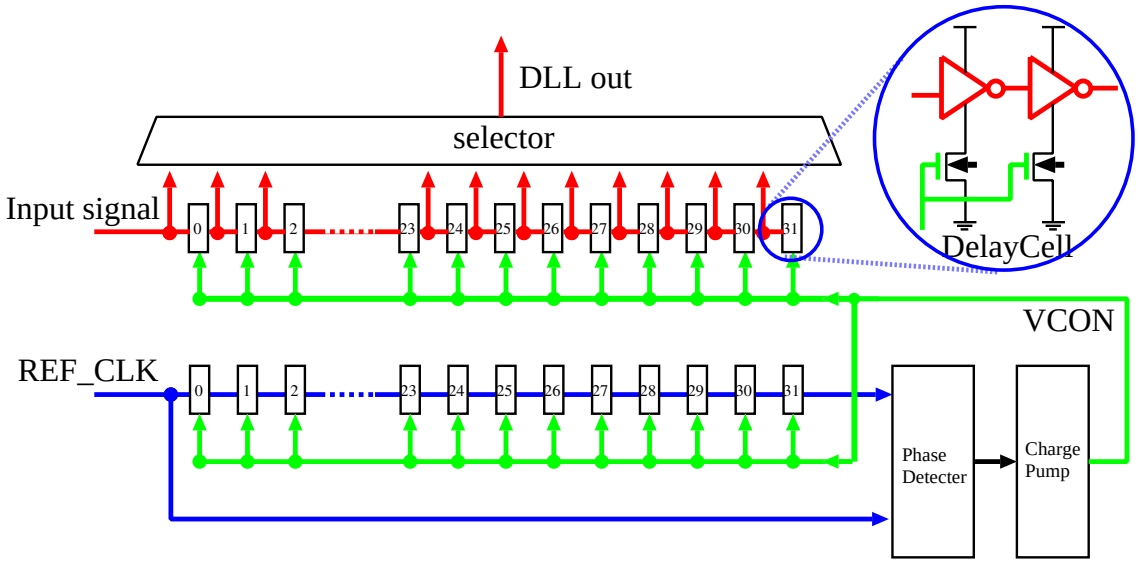


図 6.6: DLL 回路の概要

DLL 回路の動作原理

図 6.6 に DLL(Delay Lock Loop) 回路の構造の概要を示す。図中の Input signal から入力される信号は、32 段の遅延セルを通して遅延が与えられる。1 つの遅延セルは入力信号に 0.78nsec (0.78nsec は LHC クロックの周期である 25nsec の 32 分の 1) の遅延を与えて出力する。また 32 段階の遅延セルにより遅延を与えられた信号は、適宜セレクターにより選ばれて後段の回路に送られる。1 つの遅延セルは、主に 2 段の負論理回路からなり、0.78nsec はこの負論理回路を信号が通過する時間である。この負論理回路の電源は、制御用電圧 VCON により制御され、負論理回路を信号が通過する時間は、この VCON により決まる。また VCON の電圧値を決めるために、信号が通過するのと全く同等な、32 段の遅延セルをもう一つ用意し、この回路にクロック (REF_CLK) を入力している。この 32 段の遅延セルを通ったクロックは、遅延セルを通過していないクロックと共に、位相検出器に送られる。位相検出器は 2 つのクロックの位相を比べて、32 段の遅延セルによる遅延が、クロックの周期より大きいのか、小さいのかを判断し、この位相の比較に基づき、後段のチャージポンプに VCON 電圧の昇降の制御信号を出す。チャージポンプは位相検出器の制御信号を受け、VCON のラインに対するチャージの供給を制御する。VCON はこのようにして制御され、32 段の遅延セルで 25nsec の遅延が与えられるような電圧に安定する。こうして信号が通過する遅延セルにも、32 段で 25nsec の遅延を与える VCON が供給され、入力信号に可変遅延を与えることが出来る。このとき、VCON を決めるのに使われる REF_CLK の入る遅延セルは、1 チップで 1 つのみで、16 チャンネル分の信号処理を行なう遅延セルなどには、共通して、この VCON が供給される。これは 1 つのチップの内部では、半導体の動作パラメータが全く等しいことに依っている。

遅延セル

図 6.8 に遅延セルの回路図を示す。遅延セルは主に 2 段の負論理 (図中のトランジスター P0,N0,P2,N3) と、この負論理のグランド側の電流量を制御する NMOS (図中の N1,N2) から構成される。NMOS の N1,N2 は、VCON により制御され VCON が高いときには電流が多く流れ、低い時には電流が制限され小量しか流れない。よって、この NMOS に接続された負論理は、出力が 1 から 0 に変わる時 (出力がグランドに接続されるとき) に、通常の負論理よりも時間をかけて変わることになる。また負論理を 2 つもと VCON で制御するのは、信号の上がりエッジと下がりエッジで通過するタイミングを等しくするためである。

位相検出器

図 6.9 に位相検出器の回路図を示す。位相検出器は 2 つのクロックの位相を比べて up,dn 信号を出力する。図 6.7 に示すように、REF_CLK(Reference clock) が、DLL_CLK(32 段の遅延セルを通過した REF_CLK) に対して、進み位相のときは DN 信号が Hi になり、VCON は低くなる。また REF_CLK が DLL_CLK に対して、遅れた位相のときには UP 信号が low になり、VCON は高くなる。この部分は全てデジタル回路として記述される。

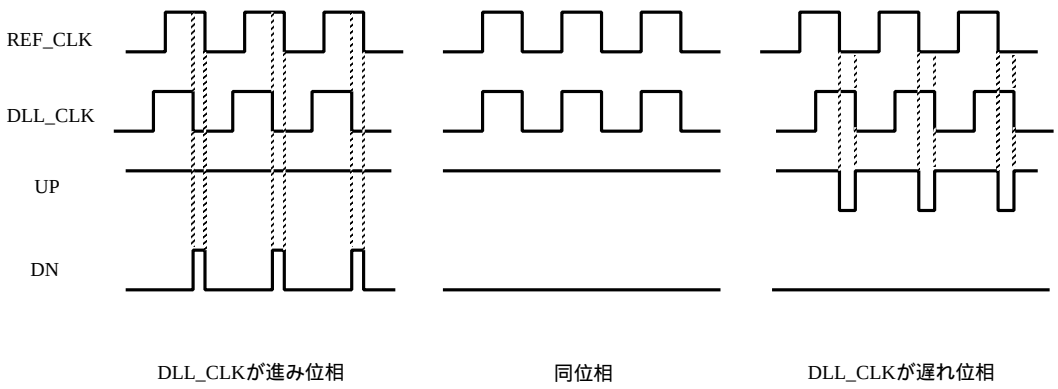


図 6.7: 位相検出器の動作

チャージポンプ

図 6.10 にチャージポンプの回路図を示す。チャージポンプは、位相検出器からの up,dn 信号を受けて VCON を出力する。DLL_CLK が REF_CLK に対して、進み位相の時には VCON を下げるように、遅れ位相のときには VCON を上げるように動作する。up 信号が ON 状態 (up 線が low の状態) では、図中のトランジスター P6 が ON 状態になり、VCON は抵抗を介して VDD 側と接続される。また dn 信号が ON 状態 (dn 線が hi の状態) では、トランジスター N260 が ON 状態になり、VCON は抵抗を介してグランドと接続される。

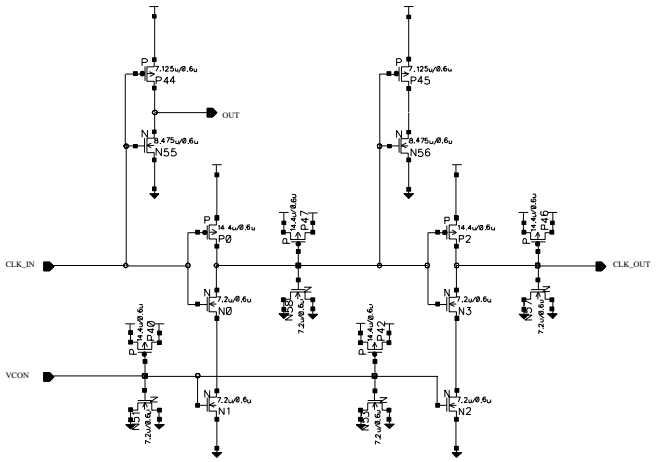


図 6.8: DELAY CELL の回路図

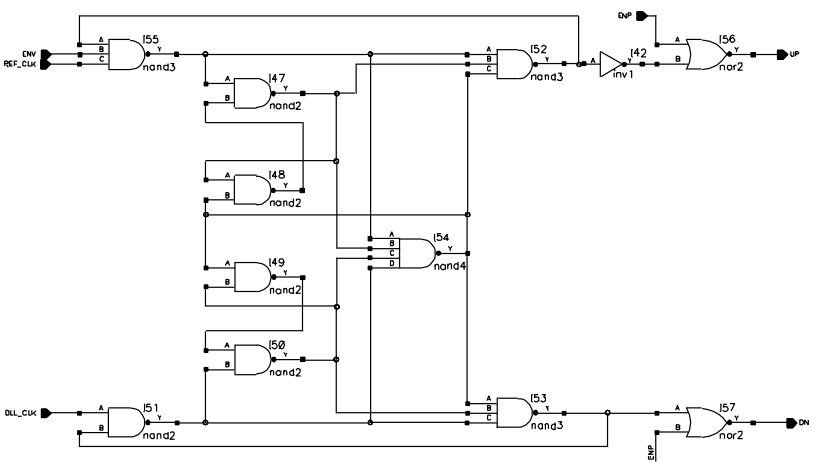


図 6.9: 位相検出器の回路図

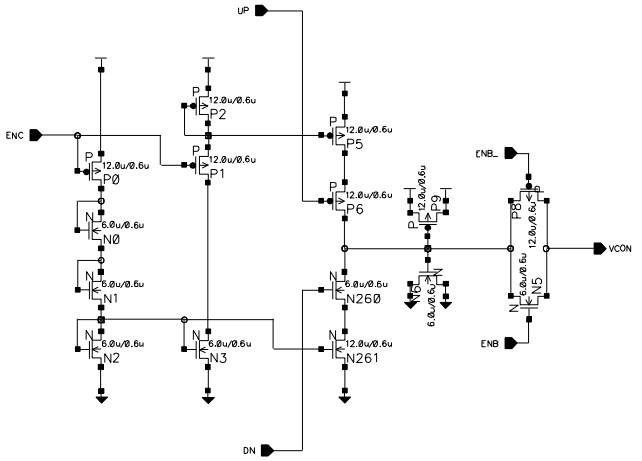


図 6.10: チャージポンプの回路図

6.2.3 BCID 回路

BCID 回路の機能

BCID(Bunch Crossing ID) 回路は、ASD から様々なタイミングで送られてくる信号を、LHC クロックに同期させ、1 クロック幅のパルスを出力する回路であり、これは入力された信号が、どのバンチ衝突により生成されたミューオンのものなのかを決定することに相当する。このバンチ識別を正しく行なうため、BCID 回路の前段にはサブナノ秒精度の遅延の調節が可能な、DLL 回路が設置される。また、この回路に送られる LHC クロックにも、同様に DLL による可変遅延がかけられる。さらに図 6.11 に示すように、信号の到着時間のばらつきが、25nsec を超える場合にも対処出来るように、ゲートの幅を 25 から 50nsec までの可変で設定できる。

回路図と動作シミュレーション

BCID 回路は図 6.12 に示す示すように、上段、下段でそれぞれ 2 つのフリップフロップと AND 回路により構成される。このうち上段が主回路で、下段の回路は、上段に入るクロックよりも遅れたクロックにより動作させることで、実効的ゲート幅を拡張するための回路である。上段、下段の出力は OR がとられ出力される。上段のクロックに対する下段のクロックの遅延時間は、拡張する分のゲートの幅に対応する。また BCID 回路の初段のフリップフロップの、イネーブルを使ってマスクの機能を持たせている。図 6.13 に、この回路の動作シュミレーションの結果を示す。図はゲート幅設定用のクロックの遅延を、25/4nsec に設定した (ゲート幅は 25+25/4nsec) 場合の動作を示すもので、図中の NDO0 から NDO4 にあるようなタイミングで、信号を入力した時の動作結果である。NDI0 から NDI4 が出力結果である。ゲートの拡張された 25/4nsec の部分に信号が入った場合、実効ゲート幅が拡張され、2 クロック幅の出力を出すことが確認できる。このとき 2 クロック分の信号の 1 つはフェイク信号にあたるが、これは Sector Logic 以降の処理によって、2 クロック目の信号を使うかどうかが、決められる。

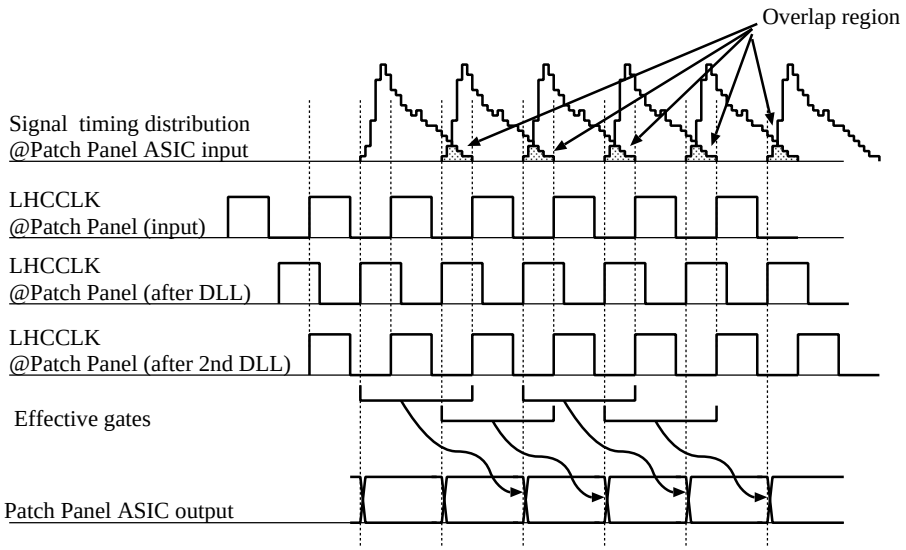


図 6.11: ゲートを拡張した場合のタイミ ング
図のように Patch Panel ASIC への入力信号のタイミングのばらつきが 25nsec よりも大きい場合、その実効的なゲート幅を拡張して、遅いタイミングでくる信号も ID できるように設定できる。またこの場合、出力信号は 2CLK 分の幅を持つことになる。

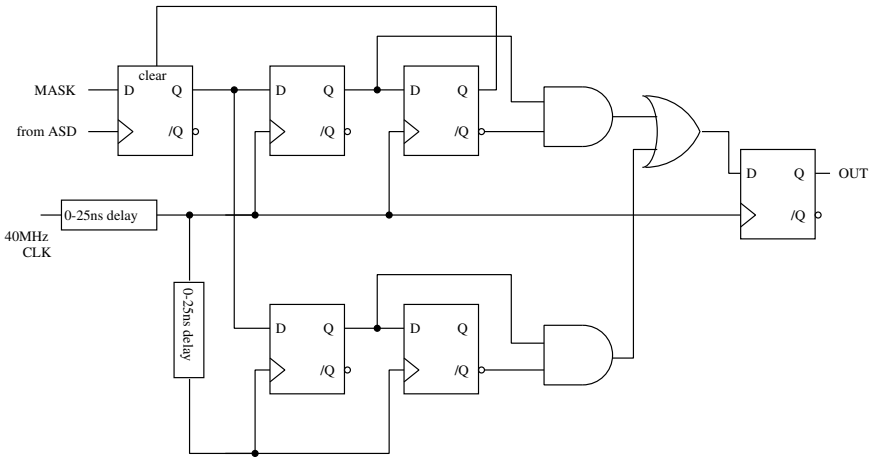


図 6.12: BCID 回路の回路図

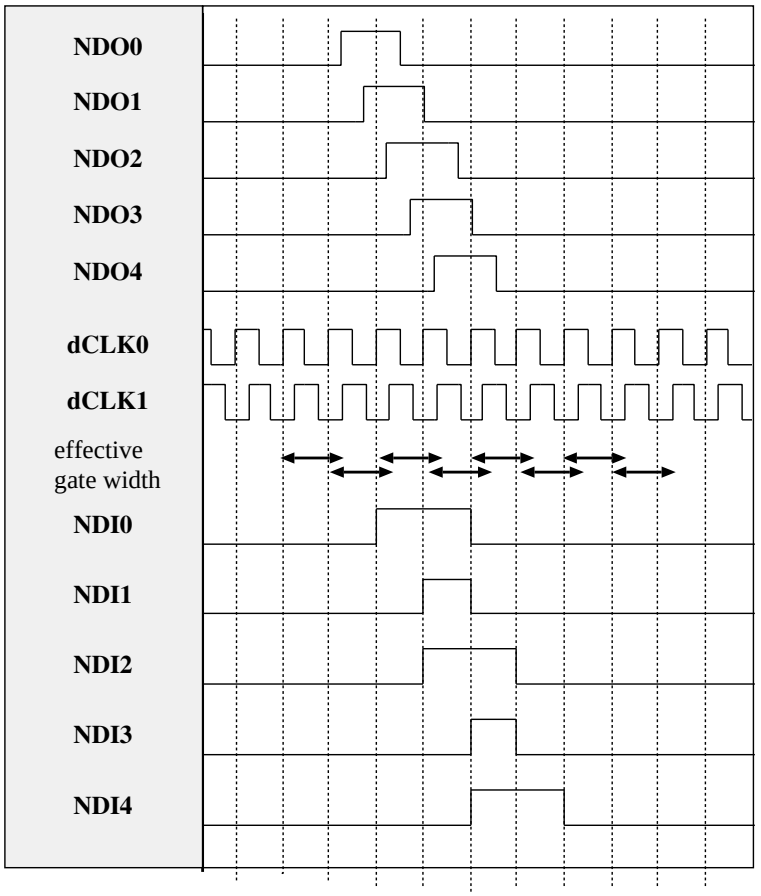


図 6.13: BCID 回路の HDL シュミレーションの結果
実効ゲート幅を 1/4CLK 拡張した場合の BCID の動作。NDO0~4 が入力信号で NDI0~4 がバンチ識別を受けた信号。隣のバンチと重複するタイミングで入ってきた信号に対しては、2CLK 幅の信号がでることが確認できる。

6.2.4 テストパルス回路

テストパルス回路は ASD に向け、差動型信号を出す回路である。この回路は TTL のトリガーを受け、ASD の閾値電圧である 50mV から 350mV 程度の可変振幅のパルスを出力する (可変は 4 段階)。またテストパルスのタイミングは、4 段階で 0 ∼ 3 クロックの幅で制御できる荒い遅延 (coarse delay) と、DLL 回路によるサブナノ精度の fine delay の 2 つで調節できる。

回路図

図 6.14, 図 6.15 にテストパルス回路の回路図とレイアウトを示す。回路の上段部は、テストパルスの振幅を決定する電流源で、10 個の PMOS により電流源の開閉を行なうことで、出力の振幅を決める。図中の INPUT 及び INPUT_には遅延回路の出力と、その逆相の信号がそれぞれ入力される。この信号を受ける PMOS の P20,P21 は、ゲート幅の長いトランジスタで、多くの電流を流すことが出来る。OUTPUT,OUTPUT_は直接 IC の外部に接続され、IC 外部で 51Ω の抵抗を介して、グランドに接続されている。

動作シミュレーション

図 6.16 に、テストパルス回路に関するシミュレーションの結果の一例を示す。これは、ON 状態のトランジスタの数に対する、テストパルスの振幅をプロットしたもので、トランジスタの各部の幅、温度、IC 製造時に生じるパラメータのばらつきなど、様々なパラメータに関してシミュレーションを行ない、最適なパラメータが選択されている。

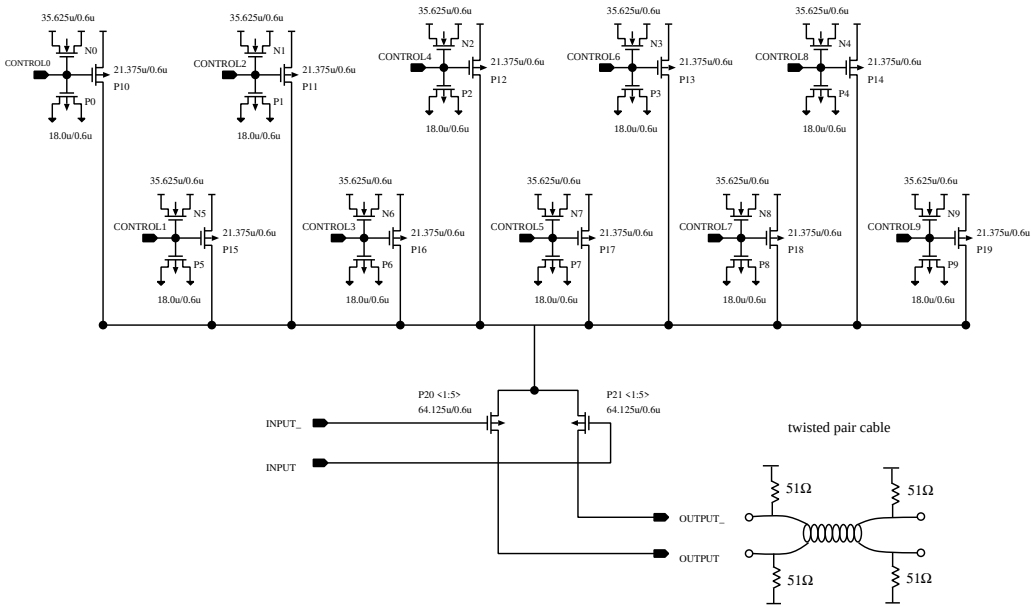


図 6.14: テストパルス回路の回路図

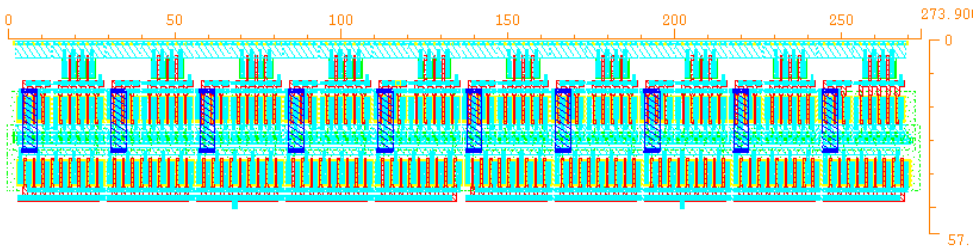


図 6.15: テストパルス回路のレイアウト

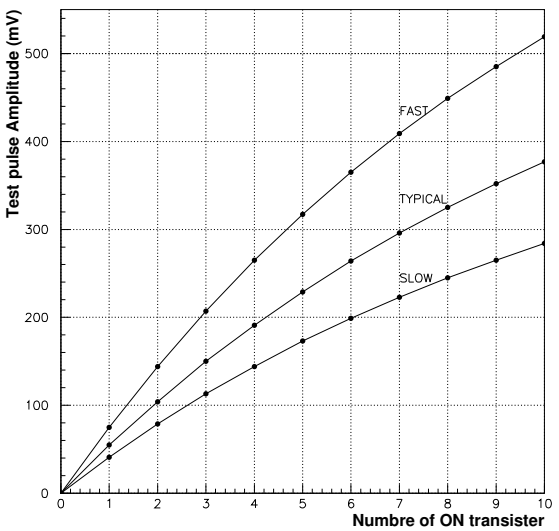


図 6.16: テストパルス回路の SPICE シュミレーションの結果

6.2.5 Patch Panel における JTAG

Patch Panel ASIC では JTAG プロトコルを用いて、IC 内部の様々なパラメータの設定や、バウンダリースキャンテストを行なう。

バウンダリースキャン

Patch Panel ASIC におけるバウンダリースキャン用のセルは、信号の入力側が LVDS レシーバーの直後、出力側が BCID 回路の直後に、それぞれ 16 チャンネルづつ設置する。また Patch Panel ASIC はタイミング調整のための回路なので、各チャンネルの動作タイミングのばらつきを最小にする必要がある。このためバウンダリースキャンセルは、完全なデジタル回路であるにもかかわらず、レイアウトレベルから設計した。

ユーザー定義レジスタ

Patch Panel ASIC に用いられる JTAG には、必須レジスタの BYPASS と INSTRUCTION の他に、表 6.1 に示すようなユーザー定義のレジスタを搭載し、主にマスク処理の設定、各部の遅延値の設定、テストパルス回路の振幅の設定などに用いられる。これらのレジスタのうち RW モードのもの (値の読み書きが可能なレジスタ) は、リセット信号により設定される初期値を持つ。また各レジスタには SEU 現象に対処するため、多数決論理回路を搭載する。また多数決論理回路上で SEU が起これば、それを検出して JTAG で読み出すことができる。各レジスタの詳しい機能に関しては、付録 B で述べる。

レジスタ名	ビット数	モード	コメント
MASK	16	RW	MASK
TESTPULSE_FINE	5	RW	Test pulse fine delay
TESTPULSE_COARSE	2	RW	Test pulse coarse delay
TESTPULSE_AMP	2	RW	Test pulse amplitude
TESTPULSE_VETO	1	RW	test pulse veto
BCID_DEL	5	RW	delay of CLK timing
BCID_GATE	5	RW	delay of gate width
SIGNAL_DEL	5	RW	delay of signal
SEU	1	R	SEU flag read out
FUSE	2	R	fuse monitor

表 6.1: ユーザー定義レジスタの一覧

6.2.6 Patch Panel ASIC の設計におけるデザインの検証

ここでは Patch Panel ASIC を設計するときに行なったデザインの検証、動作確認について述べる。

レイアウトレベルからの設計とデザインの検証

Patch Panel ASIC の場合、信号のタイミング調整という機能を持つため、1 チップで処理を行なう 16 チャンネル分の信号の動作速度は、等しくなければならない。
図 6.17 に自動配置配線によりレイアウトされた回路を示す。配置配線ツールによる自動配置配線は、まず必要な素子をランダムに配置し、その上に必要な配線を施すという方法で行なわれる。そのため図からも見てとれるように、各素子間の配線のされ方はバラバラであり、また回路の密度も低い。そのため機能的には等しい回路でも、配線経路が異なるために配線抵抗、配線容量に違いが生じて、同じタイミングで動作させることが難しくなる。

そこで Patch Panel ASIC では、信号経路の回路の全てをレイアウトレベルから設計し、各チャンネル間でのタイミングのばらつきが最小になるように設計した。また高密度な配線をすることになり、結果的には、従来使用する予定であったチップに対して、1/4 の面積のチップで Patch PanelASIC を実現できることがわかり、コストも約 1/2 になることがわかった。

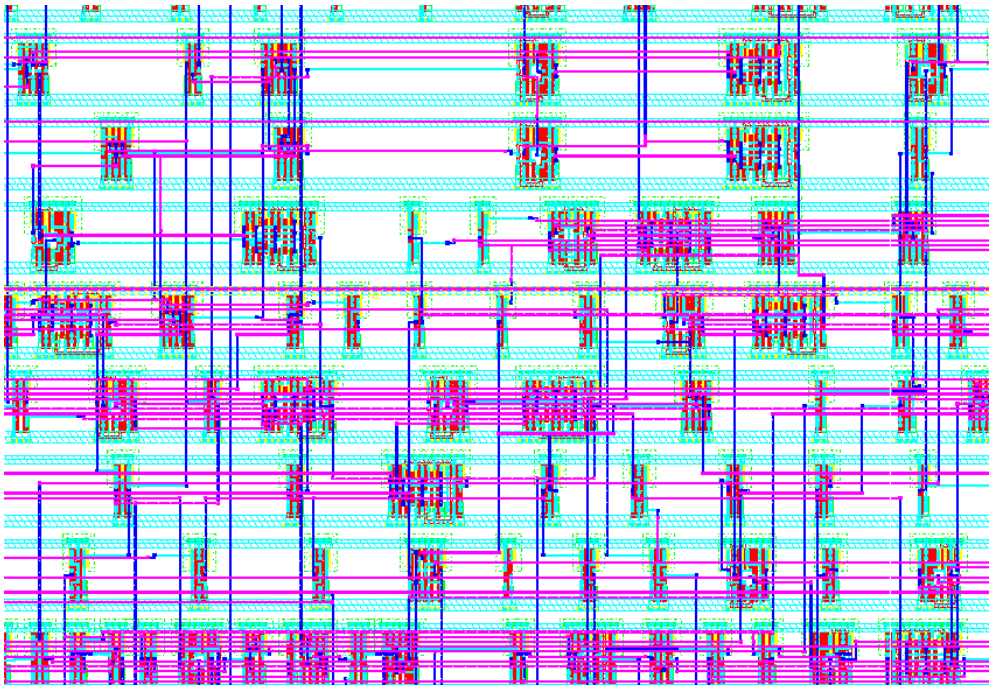


図 6.17: 自動配置配線によるレイアウト
自動配置配線では、各素子をランダムに配置し、その上に配線が行なわれる。そのため回路の密度は低く、また寄生容量、寄生抵抗が生じやすい。

Patch Panel ASIC の信号経路部分に関しては、以下に示すような回路ごとに、各種の論理素子を組み合わせて、レイアウトを行なった。これらのレイアウトに対して、DRC(Design Rule Check)

を行ない、ローム社から与えられているデザインルールを満たすことを確認した。また、これらの回路と同等の回路図を記述し、LVS(Layout vs Schematic) により、レイアウトと、この回路図が一致することを確認した。また、これらの回路の構成要素である全ての論理素子に関しても、同じように、DRC と LVS を行ないデザインが正しいことを確認している。

レイアウト、回路の例として、図 6.18、図 6.19 に BCID 回路のレイアウト、回路図をそれぞれ示す。

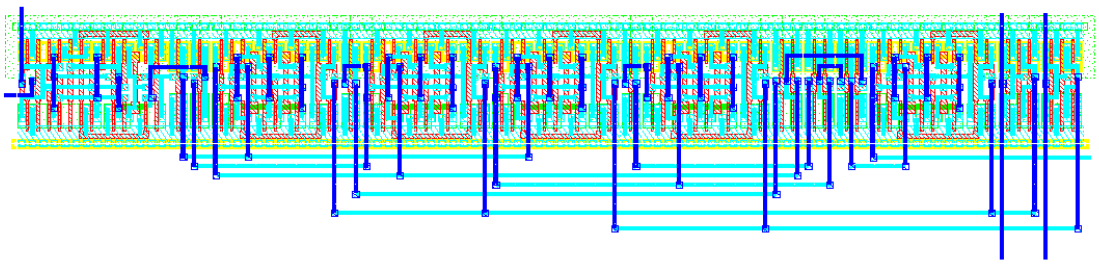


図 6.18: BCID 回路のレイアウト

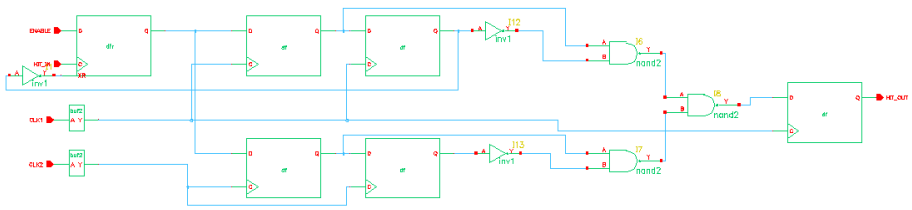


図 6.19: BCID 回路の回路図

試作 IC の結果を踏まえたアナログ回路部の検証

今回の Patch Patch ASIC の製作の前に、各部の機能ごとに動作確認を行なうために、幾つかの ASIC が試作されている。アナログ回路のうち、LVDS レシーバー、テストパルス回路に関しては、これまでの試作で正しく動作することが確認されていて、今回の Patch Panel ASIC の製作では、これらの動作確認のされた回路と同等のものを搭載した。次の節では、これらの試作 ASIC の動作検証に関しても述べる。またアナログ回路のうち、DLL 回路はこれまでの試作 ASIC で、動作タイミングが遅いことが分かっている。今回の Patch Panel ASIC では、DLL 回路を全てレイアウトで記述し、早く動作するようにした。

デジタル回路部分の動作の検証

主に JTAG プロトコルからなるデジタル部分に関しては、verlogHDL の記述に基づいて、動作シミュレーションを行ない、その動作を確認した。verilogHDL によるシュミレーション結果の例として、図 6.20 に、TAP コントローラの動作確認の結果と、図 6.21 に、EXTEST の動作確認の結果を示す。またその他の、JTAG に関するテスト結果は付録 B に示す。

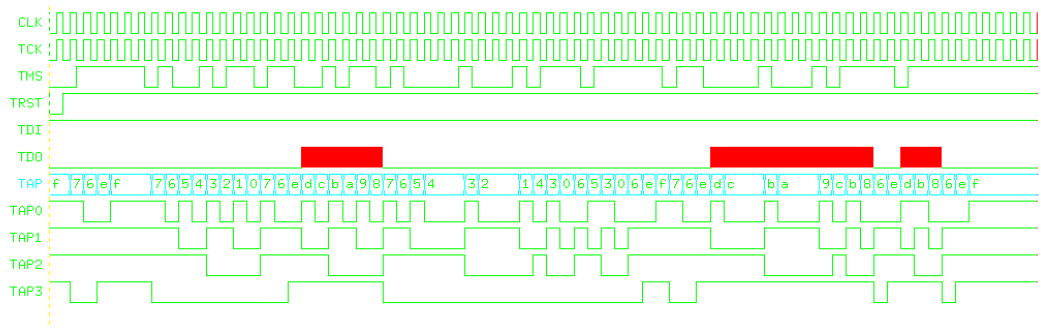


図 6.20: JTAG のテスト波形 tap state の遷移
TAP コントローラに関して、TMS,TCK からパターンを入力して Test-Logic-Reset, Run-Test/Idle,Slect-DR(IR)-Scan,Capture-DR(IR),Shift-DR(IR),Exit-DR(IR), Pause-DR(IR),Exit2-DR(IR),Update-DR(IR) と TAP コントローラの全ての状態とその間の全ての遷移経路を通り、正しく動作していることを確認した。

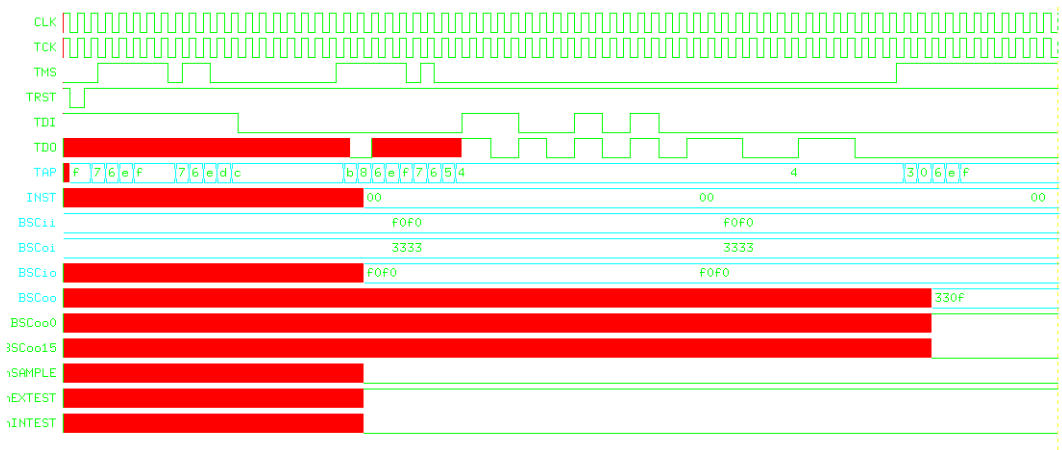


図 6.21: JTAG のテスト波形,EXTEST
BSC の入力側にパターン (f0f0)、出力側にパターン (3333) を設定しておきインストラクションレジスタに”11:1111_1111” を書き込み、EXTEST 命令にして、TDO よりこれらの値を読み出されることを確認した。

6.2.7 Patch Panel ASIC 全体のレイアウト

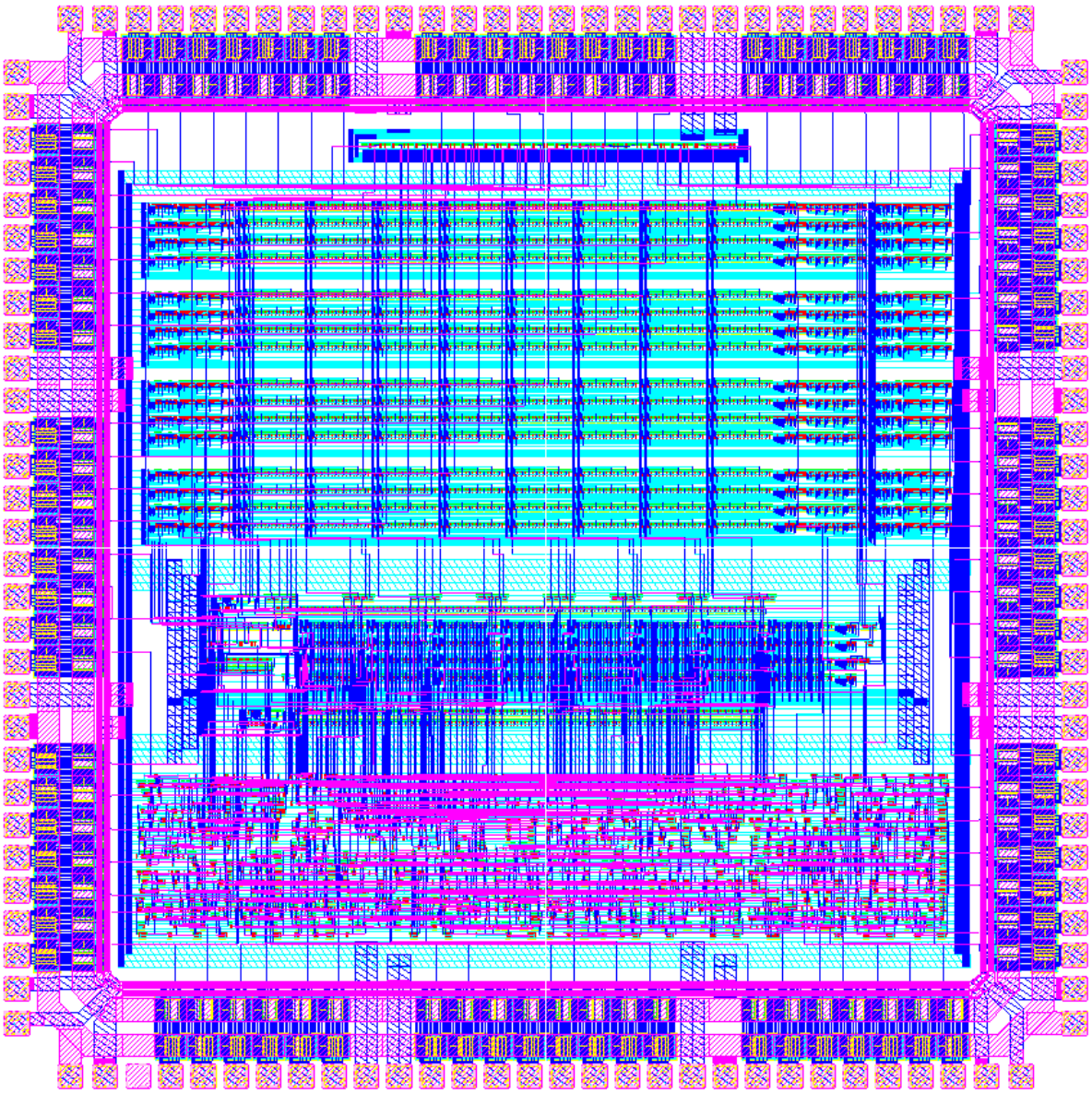


図 6.22: Patch Panel ASIC ver0 のレイアウト

上で述べた動作検証、また試作 ASIC の動作結果を踏まえて、フルスペックの Patch Panel ASIC を開発した。図 6.22 に Patch Panel ASIC(ver0) のレイアウトを、図 6.23 に、これに対応するブロック図を示す。図 6.23 にあるようにこの ASIC は主に以下に示すような 4 つブロックからなる。

- LVDS レシーバーブロック
  図の最上部。16 チャンネル分の、LVDS レシーバーが配置されている。また LVDS レシーバーは、他のデジタル回路による電源の揺らぎ (ノイズ) の影響を受けないよう、電源が他の部分の電源と独立になっている。

- 信号処理部

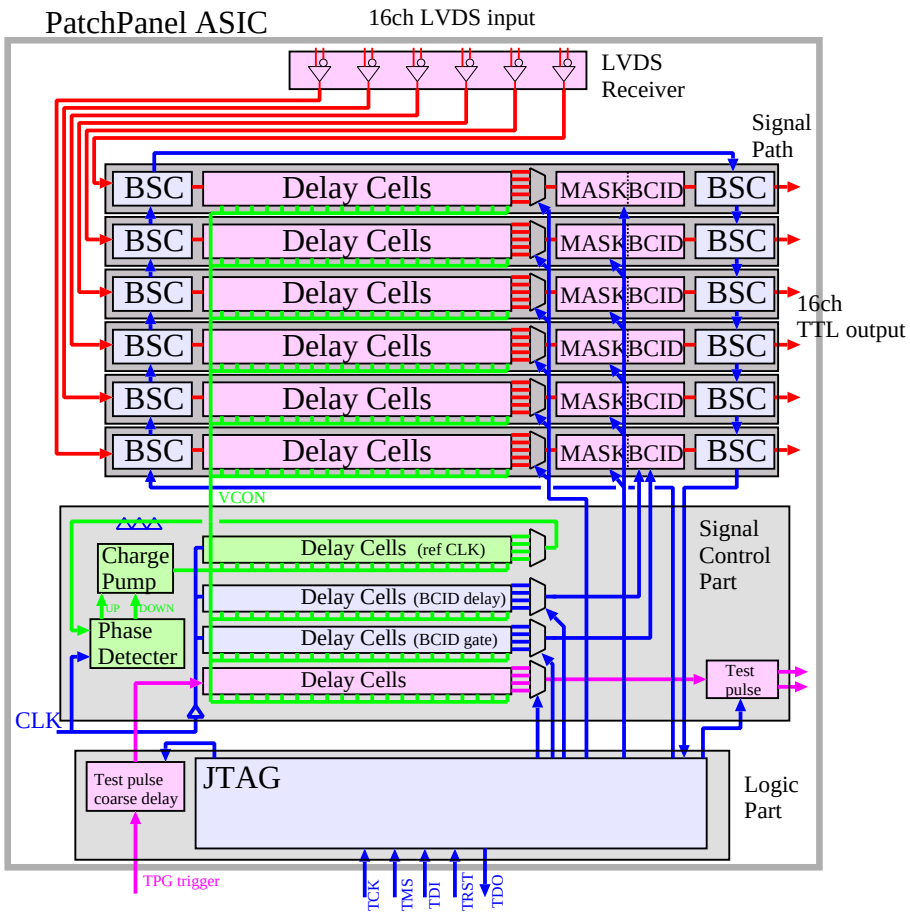


図 6.23: Patch Panel ASIC のブロック図

バウンダリースキャンセル、遅延回路、BCID 回路 (MASK 回路も含む) から構成され、16 チャンネル分の信号処理を行なう部分。信号が通過するのは LVDS レシーバーと、この信号処理部であるが、これらの部分の回路は全てレイアウトレベルにより記述され、寄生素子によって生じるチャンネル毎のタイミングの差を最小にしている。

- 信号制御部
  DLL 回路のリファレンスクロックが通る 32 段の遅延セル、位相検出器、チャージポンプ、また BCID 回路に入る LHC クロック、ゲート幅設定用の遅延回路、テストパルストリガーのための DLL 回路、及びテストパルス回路、また信号制御部に入る各種制御信号のバッファーが設置される。

- 論理回路部
  JTAG プロトコルのための回路と、テストパルストリガーのクロック単位の遅延回路からなる。TAP コントローラ、インストラクションレジスタ、全てのユーザー定義レジスタなどがここに設置される。

この IC はローム社のゲート間隔 0.6μm のフルカスタムチップのプロセスを使って製作した。サイズは 4.5mm×4.5mm である。これは 2000 年 6 月に VDEC に提出し、2001 年 1 月に納品されたもので、現在も動作確認を行なっている途中であるが、次節では、この IC の動作結果も述べる。

6.3 Patch Panel ASIC 試作 IC の動作検証

ここでは Patch Panel ASIC のために、開発されてきた 3 種類の試作 ASIC と、前節で述べたフルスペック ASIC の動作結果に関して述べる。各 ASIC の内容は以下のようになる。

- DLL テスト ASIC
  DLL 回路のみを搭載したテスト ASIC で、回路は自動配置配線によって設計されているので、多くの寄生容量、寄生抵抗を含むことになる。使用プロセスは、ローム社のゲート間隔 0.6μm のフルカスタムチップで、サイズは 4.5mm×4.5mm である。(図 6.24)

- アナログ部テスト ASIC
  Patch Panel ASIC のうちのアナログ回路の部分、DLL 回路、テストパルス回路、LVDS レシーバーを含むテスト ASIC である。使用プロセスは上と同じローム社のゲート間隔 0.6μm のフルカスタムチップでサイズは 4.5mm×4.5mm である。(図 6.25)

- Patch Panel ASIC(version (-1))
  上記のアナログ部テスト ASIC の内容に加えて、JTAG プロトコルを含めたテスト ASIC である。使用プロセスはローム社のゲート間隔 0.6μm のフルカスタムチップでサイズは 9mm×9mm である。(図 6.26)

- Patch Panel ASIC(version 0)
  前節で述べた ASIC である。Patch Panel ASIC の全機能を含む。

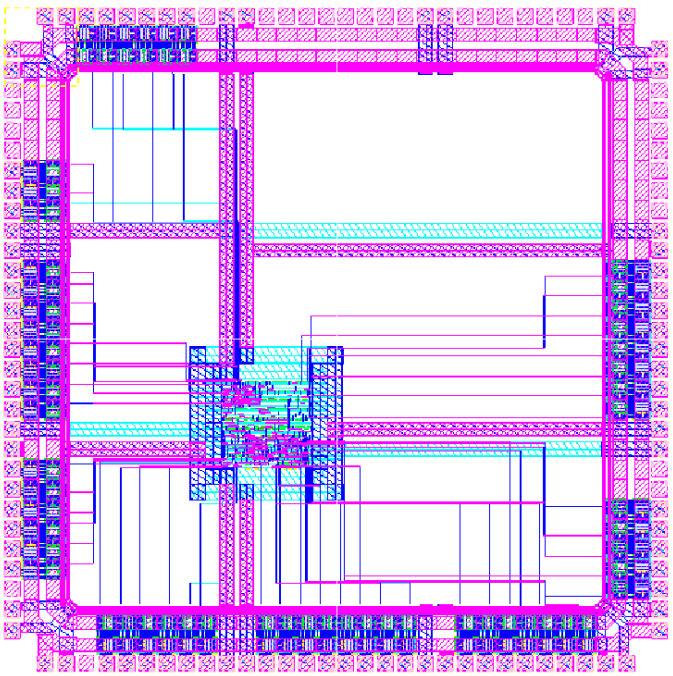


図 6.24: DLL 回路テストチップのレイアウト

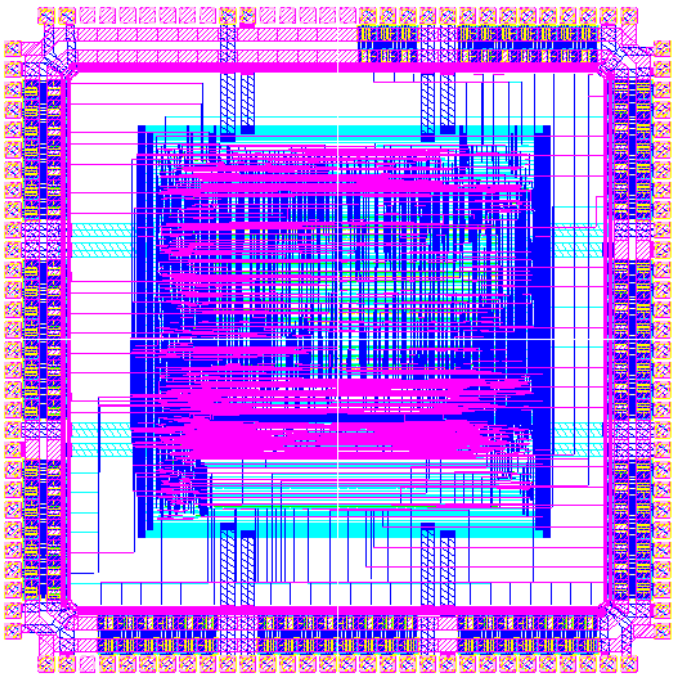


図 6.25: Patch Panel ASIC アナログ回路テスト ASIC

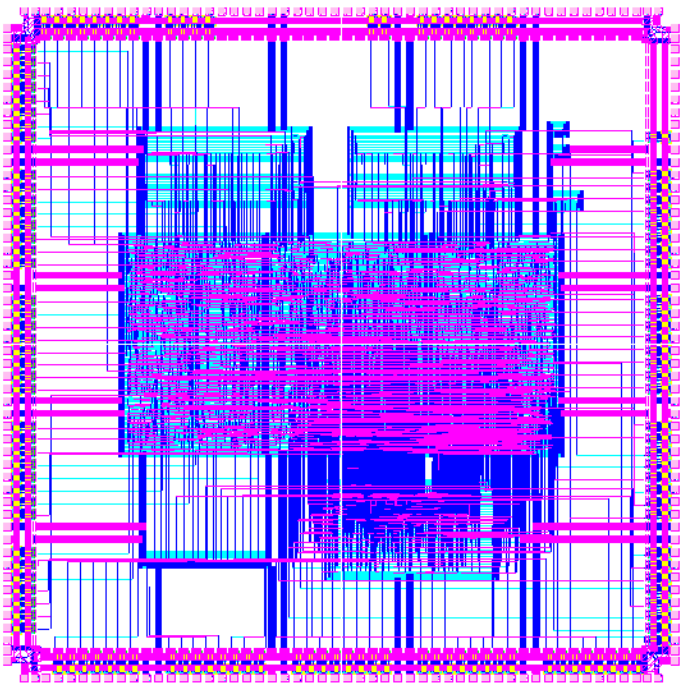


図 6.26: Patch Panel ASIC(version (-1))

6.3.1 LVDS レシーバーの動作検証

LVDS レシーバーは、その動作特性が、LVDS のオフセット電圧に依存することから、入力信号に対する出力信号の遅延を測定した。図 6.27 に、LVDS レシーバーに関する測定を行なうためのセットアップを示す。パルスジェネレーターから様々なオフセットと振幅の差動信号を入力し、IC の入力の直前と、出力の直後の信号のタイミングの差を計測した。差動信号の入力部には、ターミネーションとして、100Ω の抵抗を取り付けいている。IC の入出力の間には LVDS レシーバーの他に、数個のゲート素子が入ることになるため、LVDS レシーバー自身の動作速度を正確に計ることは出来ないが、入力の信号の設定の仕方により、動作速度の相対的な違いを計測することができる。IC の動作電圧は 3.3V である。

ここでは Patch Panel ASIC(ver0) に搭載した、LVDS レシーバーの測定結果を図 6.28 に示す。図 6.29 に示すように、Amplitude は LVDS 信号の振幅 (差動信号の上限と下限の電圧値の差) であり、offset は LVDS 信号の中心電圧である。

ASD からの出力は Offset=1.2V,Amplitude=400mV であり、測定結果によると、ASD 出力の周りの電圧領域では、Offset 電圧の電圧の違いにより、タイミングに差が生じることはない。一般的な LVDS の規格では Offset に関しては、0∼2[V] の範囲での動作が求められるが、この LVDS レシーバーの場合、Offset=2[V] の領域では、Offset=1.2[V] の領域に比べ、1nsec の遅延があるのみである。また振幅 (amplitude) に関しては、0.05nsec/10mV の遅延時間の傾きがある。しかし、ASD は Patch Panel ASIC の設置される PS-Board から、電源を供給されるため、ASD-Patch Panel ASIC 間でグランドレベルに差が生じたり、振幅が有意に変化することは無いと考えられる。よって、この LVDS レシーバーは実験で使用するための十分な性能を持つと結論できる。

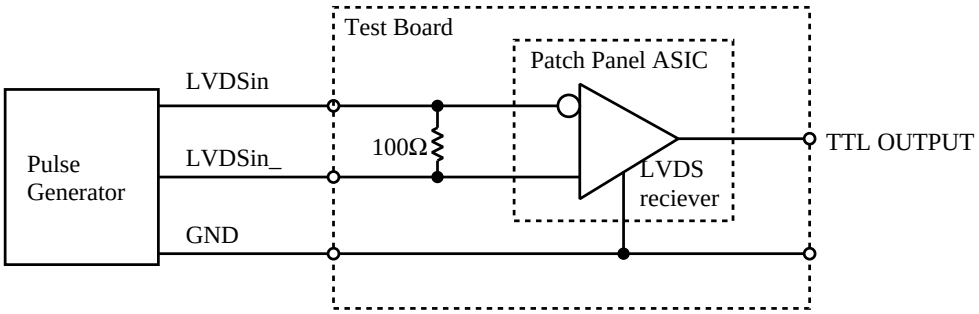


図 6.27: LVDS レシーバーの計測のセットアップ
パルスジェネレーターから、様々なオフセット、振幅の差動信号を入力し、IC の入力の直前と、出力の直後の信号のタイミングの差を計測した。またこのときの IC の動作電圧は 3.3V である。

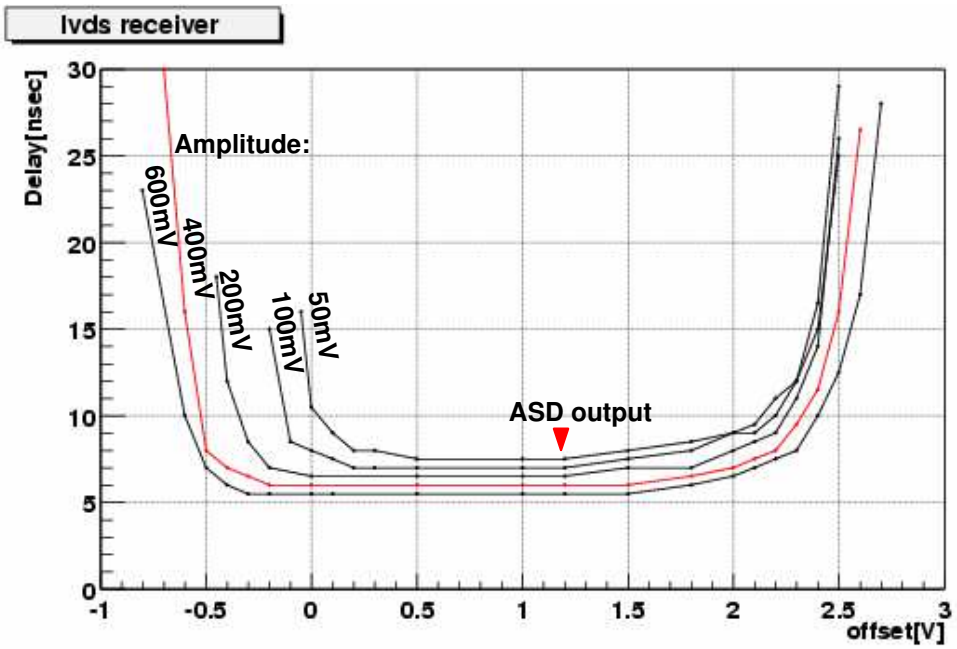


図 6.28: LVDS レシーバー オフセット電圧に対する出力信号の遅延
ASD からの出力は Offset=1.2V,amplitude=400mV であり、この領域では Offset の変化に対して有意なタイミングのずれは無いことが分かる。

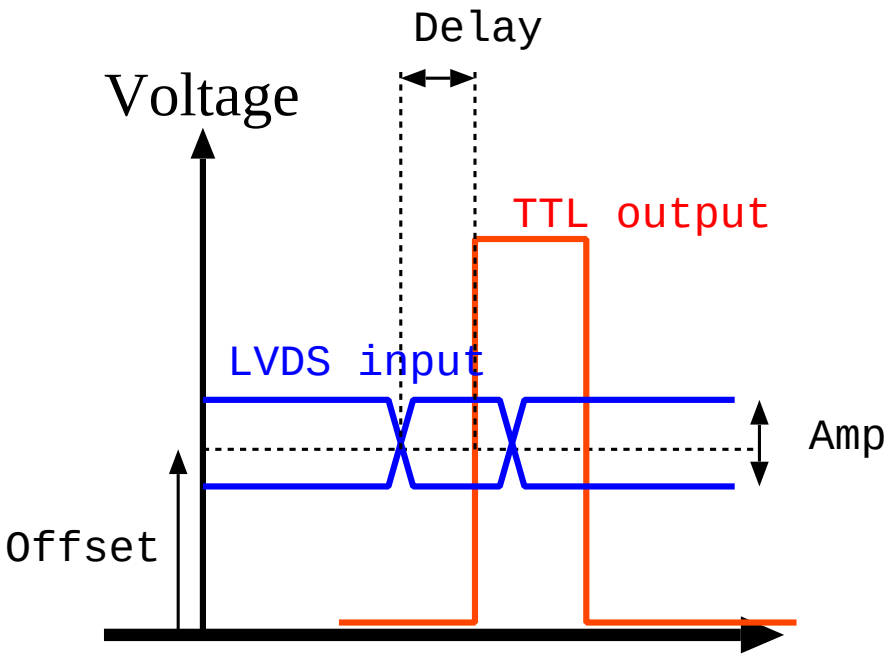


図 6.29: LVDS レシーバー オフセットと振幅の定義
Offset とは LVDS 信号の中心電圧を示し、amplitude はその振幅を示す。また ASD からの出力は Offset=1.2V,amplitude=400mV である。

6.3.2 遅延回路 (DLL) の動作検証

DLL 回路の特性 (VCON に対する、32 段の遅延セルの与える遅延時間) を図 6.30 のようなセットアップで測定した。32 段の遅延セルには、図のように外部から VCON を設定し、遅延を与える対象となる信号としてクロックを入力した。測定は、IC に入る直前のクロックと、32 段の遅延セルの直後の出力クロックのタイミングを測定した。

今回は、DLL テスト ASIC と、前節で述べた Patch Panel ASIC(version 0) に関して測定を行なった。その結果を図 6.31 に示す。DLL は、IC の動作電圧である 3.3[V] 以下の VCON で、25nsec の遅延を与える回路である。そのため DLL の特性曲線は、VCON が 3.3[V] 以下の領域で、25nsec の遅延値と交差しなければならない。図のように、DLL テスト ASIC の場合、必要な動作速度よりも遅く、VCON を 5[V] まで上げても、25nsec の遅延値に交差することは無かった。

Patch Panel ASIC(ver0) では、この DLL テスト ASIC の結果を受けて、DLL 回路をレイアウトレベルから設計することで、無駄な配線をなくし、徹底的に寄生容量の除去を行なった。そのため Patch Panel ASIC(ver0) では、動作が十分に早くなり、VCON が 3.3[V] 以下の領域で、25nsec の遅延値と交差する結果が得られた。この結果は、今回の Patch Panel ASIC(ver0) で初めて得られたものである。

また DLL は、位相検出器、チャージポンプを使った VCON をロックする機能もあるが、今回の Patch Panel ASIC(ver0) では、位相検出器の一部に配線ミスがあり、その機能を確認することが出来なかった。DLL テスト ASIC は、25nsec 周期のクロックでは動作しないが、これを 30nsec 以上の周期にすれば、ロック機能が正しく動作するので、位相検出器、チャージポンプの機能は確認されていることになる。またこれらは、全てデジタル回路なので、Patch Panel ASIC(ver0) の配線ミスは簡単に修正することが出来る。よって次回の ASIC 製作では、完全な DLL 回路の完成が期待できる。

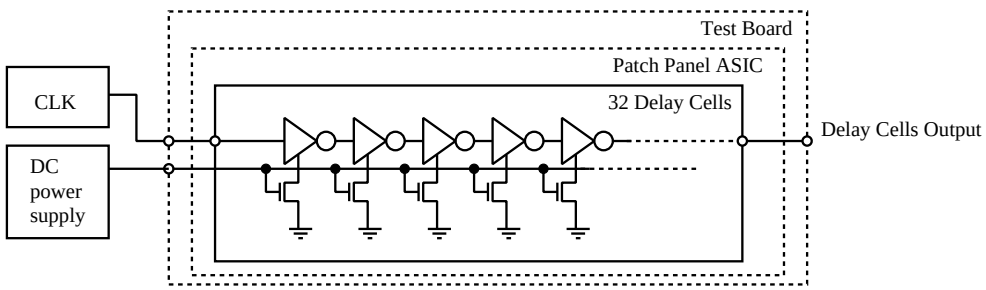


図 6.30: DLL 回路の特性を測定するためのセットアップ
外部から VCON の電圧を設定して、遅延セルにクロックを入力し、出力の遅延時間の測定を行なった。

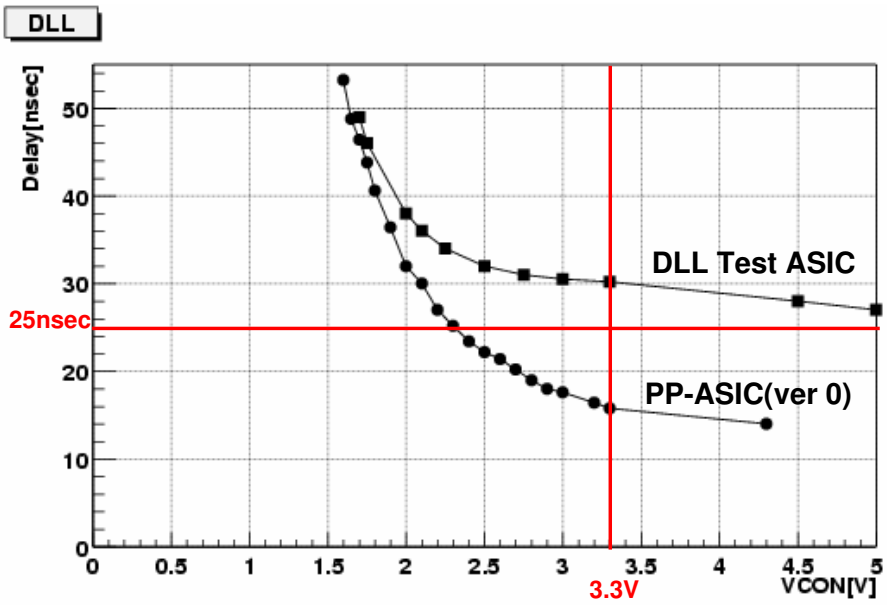


図 6.31: 制御電圧 VCON に対する遅延時間
IC の動作電圧である 3.3V 以下でクロックの周期 25nsec よりも、小さな遅延にならなければならないが、DLL Test ASIC では若干動作が遅い。Patch Panel ASIC(ver0) では、これが改善され、必要な速度で動作することがわかる。

6.3.3 テストパルス回路の動作検証

図 6.32 にテストパルス回路の動作を検証するためのセットアップを示す。図に示すようにパルスジェネレーターを通して、ASIC に TTL のパルスを入力し、出力の差動信号を測定を行なう。また出力の端子は 50Ω の抵抗を介して、グランドに落している。テストパルス回路の使用時には、テストパルスの受信側にも同様に 50Ω の抵抗を付けるが、今回は片側の抵抗のみを使って測定を行ない、振幅をこの測定値の 1/2 の値とした。

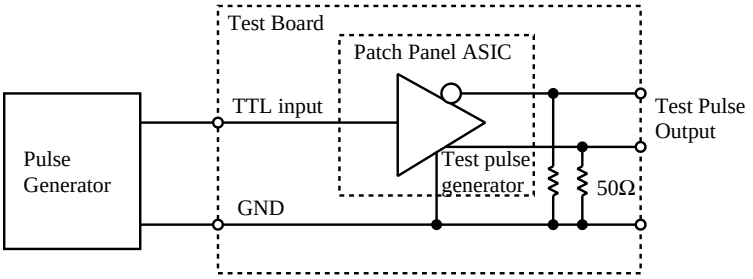


図 6.32: テストパルス回路の計測のセットアップ
パルスジェネレーターを通して ASIC に TTL のパルスを入力し、出力の差動信号を測定した。また出力の端子は 50Ω の抵抗を介して、グランドに落している。テストパルス回路の使用時には、テストパルスの受信側にも同様に 50Ω の抵抗を付けるが、今回は片側の抵抗のみを使って測定を行ない、振幅をこの測定値の 1/2 の値とした。

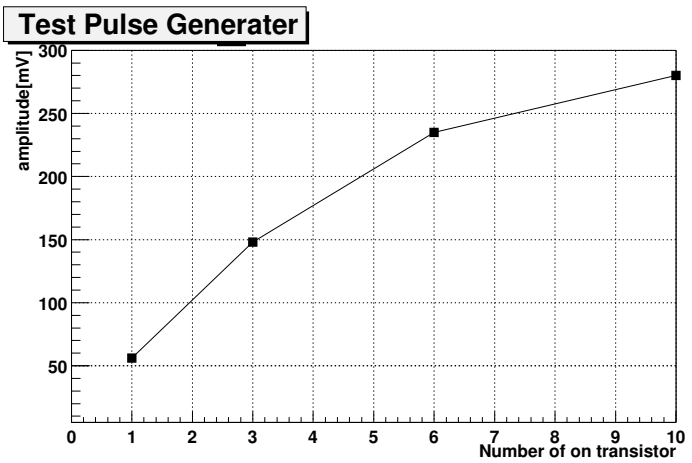


図 6.33: テストパルス回路 閉じた CMOS の個数に対する電圧の値

測定結果 (テストパルス回路の ON 状態のトランジスタの数に対する振幅) を図 6.33 に示す。測定点が 4 つのみなのは、この回路が 2bit のレジスタによって制御されるため、4 つの状態しか測定できないからである。
テストパルス回路には、ASD の閾値以上の電圧で 3 段階の振幅で出力を出すという、要求がある。ASD の閾値電圧は約 50mV であるので、この測定の結果では、その閾値電圧から、その 5 倍程度の振幅が得られることがわかる。これは十分にこの要求を満たすものである。また振幅による動作タイミングの変化は十分に小さく、最大振幅と最小振幅の場合で、テストパルスの出力のタイミングの差は 0.2nsec 以下であることを確認した。

6.3.4 JTAG プロトコルの動作検証

図 6.34 に JTAG プロトコルの動作確認をするためのセットアップを示す。図のように PPG(Pulse Pattern Generator) を使い TCK、TMS、TDI にテストパターンを入力し、ASIC の出力をロジックアナライザーで測定した。またこのテストパターンは verilogHDL 記述に基づくシミュレーションで、その動作波形を得ているもので、動作の確認は、シミュレーションの結果と、ASIC の動作を比べることで行なった。

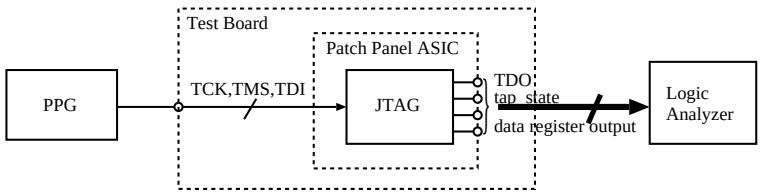


図 6.34: JTAG の動作確認のセットアップ

今回、動作を確認したのは以下に示すような、主に 3 種のテストパターンである。これらのパターンは JTAG プロトコルのうち最も重要なもので、これらのテストにより、JTAG プロトコルの機能のほとんどを確認することになる。

- TAP コントローラの全ての状態と全ての遷移経路を通過させた。これは前節のシュミレーションの結果の図 B.1 に示すパターンを使って行ない、図のパターンと全く等しい出力を得ることが出来た。
- BSC を使った EXTEST 命令を行なった。これは前節のシュミレーションの結果の図 B.3 に示すパターンと、その他の幾つかのパターンを使って行ない、シミュレーションのパターンと全く等しい出力を得ることが出来た。
- データレジスタの読み書きを行なった。これもシュミレーションの結果を用いてパターンを入力し、シミュレーションのパターンと全く等しい出力を得ることが出来た。

またこれらの動作テストは、TCK の周波数を変えながら行ない、40MHz まで正確に動作することを確認した。JTAG の動作周波数は 10MHz 以下に設定される予定なので、今回の ASIC で動作速度は十分であることが確認できた。

6.4 Patch Panel ASIC に関するまとめ

TGC トリガーシステムにおいてタイミング調整の要となる Patch Panel ASIC の設計、及び ASIC のテストを行なった。Patch Panel ASIC の信号処理系は ASD からの LVDS 信号を受ける LVDS レシーバー、サブナノ秒精度で調節可能な遅延回路 DLL、信号をクロックに同期させるバンチ識別回路、また信号のバウンダリースキャンテストを行なうための BSC からなり、これらの回路を全てレイアウトレベルから記述することにより、チャンネル間の寄生容量、寄生抵抗の違いにより生じるタイミングのばらつきを最小に押え、また回路を高密度に配置し、IC の面積を小さくすることで大幅なコストの削減につなげた。また信号処理の他に ASD に対して、タイミングと振幅が可変なテストパルスを出力するテストパルスジェネレーターを搭載している。さらに、これらの可変遅延や振幅などの値を設定するためのプロトコルとして JTAG を搭載しており、離れた場所からの IC のパラメータの設定が可能になっている。

著者はこの ASIC を様々なシュミレーションやデザインのチェックを行ないながら開発した。今回、開発した Patch Panel ASIC(ver0) のテストでは、DLL の動作速度の特性に関して、初めて正しい結果が得られた。また LVDS レシーバーの動作も、正しいことが確認でき、これまでの試作 IC の結果と合わせると、これで Patch Panel ASIC のアナログ回路部分の動作が、全て正しいことが確認できたことになる。次に製作する ASIC では、デジタル部分の若干のバグを修正することで、完全な Patch Panel ASIC が完成すると考えられる。

第7章　Slave Board ASIC, High-pT ASICの開発

ここではTGCトリガーシステムにおいて、コインシデンス処理、p_Tの判定を行なうSlave Board ASICと、High-pTの判定を行なうHigh-pT ASICの開発について述べる。主にその機能と構成、及び試作ICの開発に関して述べる。

7.1　Slave board ASIC,High-pT ASICの機能と構成

Slave Board ASIC,High-pT ASICはPatch Panel ASICによってバンチ識別がされた信号に対して、コインシデンス処理、p_T判定などを行なう部分であるが、Slave Board ASICは表7.1に示す5種類、High-pT ASICは表7.2に示す2種類の機能がある。またこれらは開発コスト、開発期間を短縮するために、それぞれ1種のASICで実現され、用途に応じて機能を切替えて用いられる。またSlave Board ASICはPS-Board上に設置され、High-pT ASICはHigh-pT Board上に設置される。以下ではこれらのASICの機能に関して説明していく。

種類	入力信号	出力信号	処理
WDSB	(2+32+2)ch×2(pivot) (6+32+6)ch×2(middle)	(4(δR)+5(R))×2(A,B)=18bit	3/4(4/4)conincidence low-$p_T$trg ($\delta R = \pm 7$)
SDSB	32ch×2(pivot) 32ch×2(middle)	(3($\delta\phi$)+5(ϕ))×2(A,B)=16bit	3/4(4/4)conincidence low-$p_T$trg ($\delta\phi = \pm 3$)
WTSB	(2+32+2)ch×3(triplet)	(1+5(R))×3(hit)	2/3(3/3) conincidence
STSB	32ch×2(strip)×2(TGC)	(1+4(R))×4(hit)×2(TGC)	1/2(2/2) conincidence
EFSB	32ch×(2(wire)+2(strip))	6bit:EI(8bit:FI)	1/2(2/2) conincidence

表 7.1: Slave Board ASICの種類

種類	入力信号	出力信号	処理
wire	triplet:18bit(32ch×3hit)×4(SB) doublet:18bit(32ch×2hit)×3(SB)	(2+5(δR))×2(trk)	high-$p_T$trg $\delta R = \pm 15$ 2/6 truck select
strip	triplet:20bit(32ch×4hit)×3(SB) doublet:18bit(32ch×2hit)×3(SB)	(2+4($\delta\phi$))×2(trk)	high-$p_T$trg $\delta\phi = \pm 7$ 2/6 truck select

表 7.2: high-pT ASIC機能の種類

7.1.1 Doublet Slave Board(DSB)

図 7.1 に DSB のブロック図を示す。DSB は middle と pivot の 2 ステーションの、各 doublet チェンバーから、それぞれ 32 チャンネルづつ (計 128 チャンネル) 信号を受ける。信号はまず、位相補正回路に入り 0.5 クロック精度で 0 ∼ 1.5 クロックの可変の遅延を受ける。位相補正の後はマスク回路を通過し、レベル 1 バッファー及びコインシデンスマトリックスに送られる。

図 7.2 にコインシデンスマトリックスのブロック図を示す。middle,pivot から、それぞれ 32 チャンネルの信号が入力されるが、隣接するチャンネルからの入力を得るために、これに加えて pivot で両隣接部から、それぞれ 2 チャンネル、計 8 チャンネル (8=2(layer)×2(side)×2(ch)) 入力され、middle 側では両隣接部から、それぞれ 6 チャンネル、計 24 チャンネル (24=2(layer)×2(side)×6(ch) 入力される。マトリックスは上下 (A,B) に分かれ、またそれぞれは図 7.3 に示すようなサブブロックから構成される。

このサブブロックには、pivot から 4(channel)×2(layer)、middle が 12(channel)×2(layer) の信号が入ることになる。それぞれの doublet チェンバーはチャンネルの半分の距離だけずらして設置されるので、4(channel)×2(layer) で (pivot の場合) 実効的な分解能は 2 倍になり、8 段階の位置情報が得られる。またこのとき 3-out-of-4 のコインシデンスがとられるが、これをオプション機能として 4-out-of-4 に設定することも可能である。また pivot の信号に対する middle の信号のずれ δR($\delta\phi$) が $\pm$7($\pm$3) の範囲で求められ、これらは対角成分毎に OR がとられる。

δR($\delta\phi$) は A ブロック、B ブロックでそれぞれ p_T が最大のもの (δR ($\delta\phi$) が最小のもの) が選ばれて、後で述べるデクラスタリング処理がなされる。以上の処理を終えた信号は、エンコードされ IC 外へ、また読み出し系に出力される。エンコードは A ブロック、B ブロックそれぞれに関して、32bit のヒット情報は 5bit に変換され、15bit(7bit) の δR ($\delta\phi$) 情報、及びトリガー信号の有無を表す 1bit を加えた 16bit(8bit) は 4bit(3bit) に変換される。

7.1.2 Triplet Slave Board

Wire Triplet Slave Board(WTSB)

図 7.4 に WTSB のブロック図を示す。WTSB は Triplet チェンバーの 3 層から、それぞれ 36 チャンネルづつ (隣接するチャンネルからの 4 チャンネルも含む)、計 108 チャンネルの信号を受ける。信号は DSB と同様に位相補正回路、マスク回路を通過して読み出し系とコインシデンスマトリックスに送られる。

コインシデンスマトリックスでは図 7.5 に示すような 2-out-of-3 のコインシデンス処理を行なう。また、このコインシデンス条件はオプション機能で 3-out-of-3 にすることも出来る。コインシデンス処理の後はデクラスタリングされ、32 チャンネルのヒット情報が 5bit にエンコードされ、さらにヒットの有無を表す 1bit を加え、1 ヒットあたり 6bit に変換され、さらに R の高い方から 3 ヒットが選ばれて合計 18bit が出力される。

Strip Triplet Slave Board(STSB)

図 7.6 に STSB のブロック図を示す。Triplet チェンバーのストリップは 2 層だけで、また隣接チャンネルからの入力が無いので、1TGC からの信号チャンネル数は 64(=32×2(strip)) になるが、1STSB は 2TGC 分の信号を受けて、合計で 128 チャンネルの入力を受ける。信号は位相補正回

路、マスク回路を通過して、読み出し系とコインシデンスマトリックスに送られる。

コインシデンスマトリックスでは図 7.7 に示すような 1-out-of-2 のコインシデンス処理を行なう。またこのコインシデンス条件はオプション機能として 2-out-of-2 にすることも出来る。

1TGC 分の 32 チャンネルは、4 つの断片に分けられ各断片でデクラスタリングが行なわれる。16bit の位置情報は 4bit にエンコードされ、さらに 1bit の信号の有無情報を加えた 5bit になる。したがって 4 つの断片では 20bit、2TGC 分で 40bit を出力することになる。

7.1.3 EI/FI Slave Board(EFSB)

EI/FI 用 doublet チェンバーからの入力は 1 層につき、EI のワイヤ側の信号が 16 チャンネル、FI のワイヤ側の信号が 32 チャンネル、ストリップ側は EI、FI 共に 32 チャンネルである。
図 7.8 に EFSB のブロック図を示す。入力はワイヤ信号とストリップ信号があり、それぞれ 32 チャンネルが 2 層 (doublet) 分で計 128 チャンネルある。入力された信号は、他の SB と同様に、信号は位相補正回路、マスク回路を通過して読み出し系とコインシデンスマトリックスに送られる。コインシデンスマトリックスは図 7.9 に示すようなもので、TSB に良く似た構造をしている。出力は EI の場合が 6bit,FI の場合が 8bit でこの信号は Sector Logic に送られる。

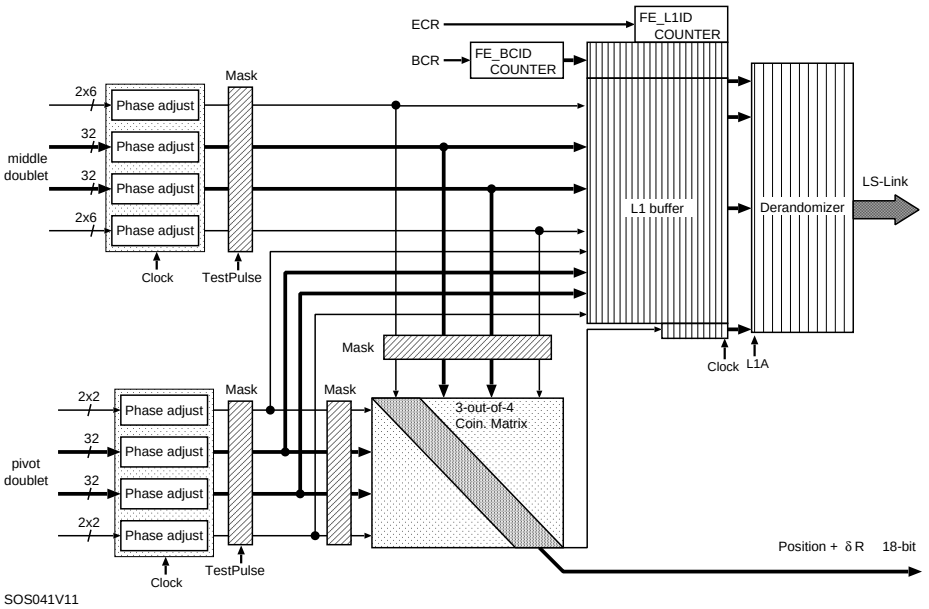


図 7.1: Doublet Slave Board のブロック図

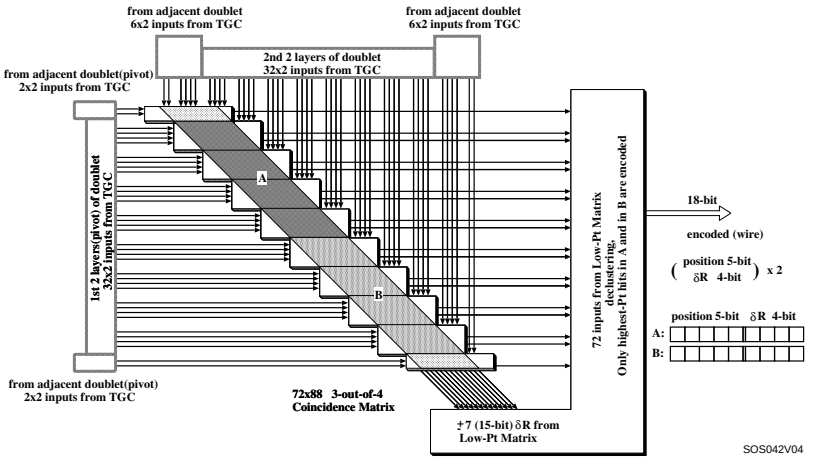


図 7.2: Doublet Slave Board の $low - p_T matrix$ の構造

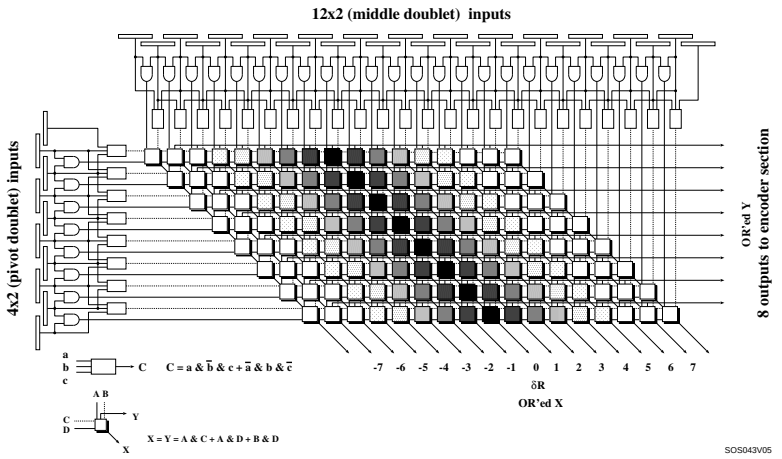


図 7.3: Doublet Slave Board の $low - p_T matrix$ の詳細な構造

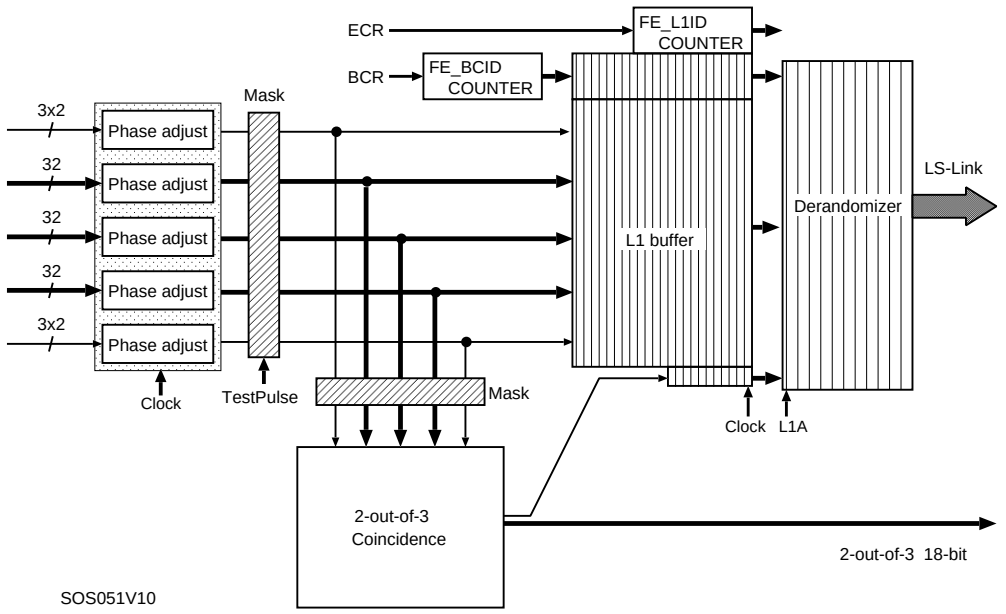


図 7.4: Triplet Slave Board(wire) のブロック図

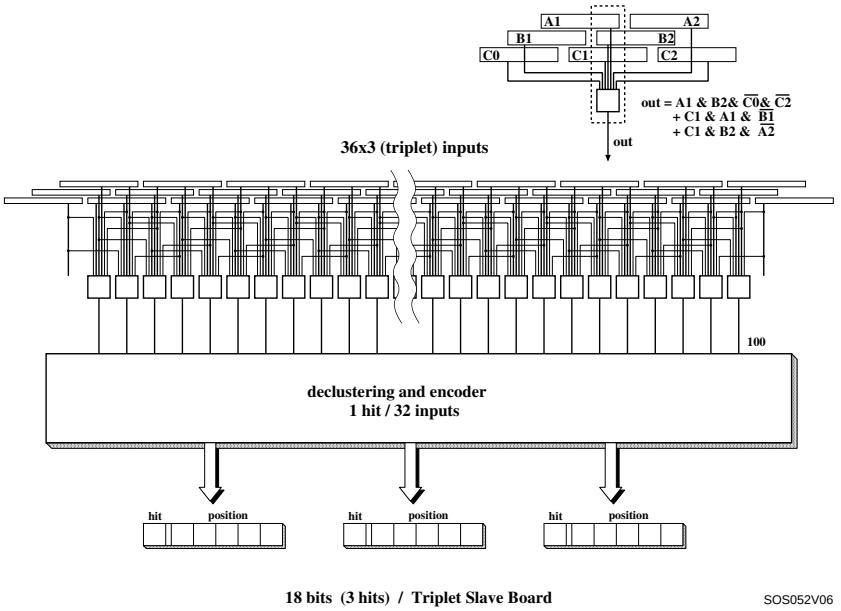


図 7.5: Triplet Slave Board(wire) の論理

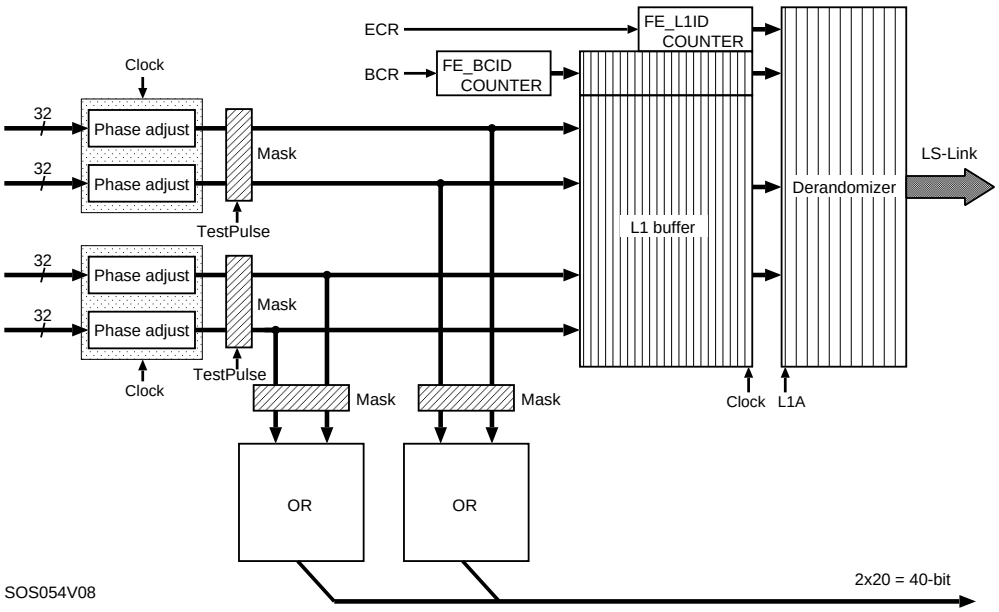


図 7.6: Triplet Slave Board(strip) のブロック図

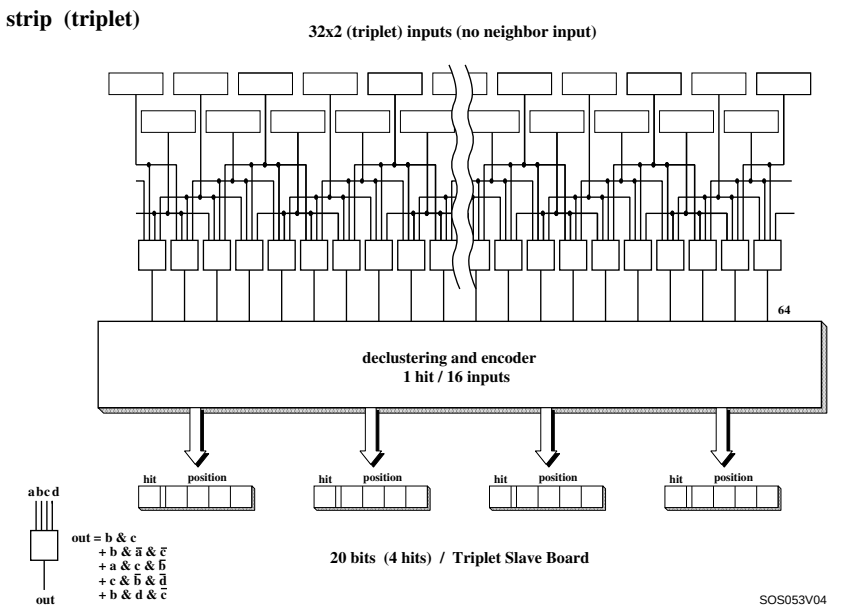


図 7.7: Triplet Slave Board(strip) の論理

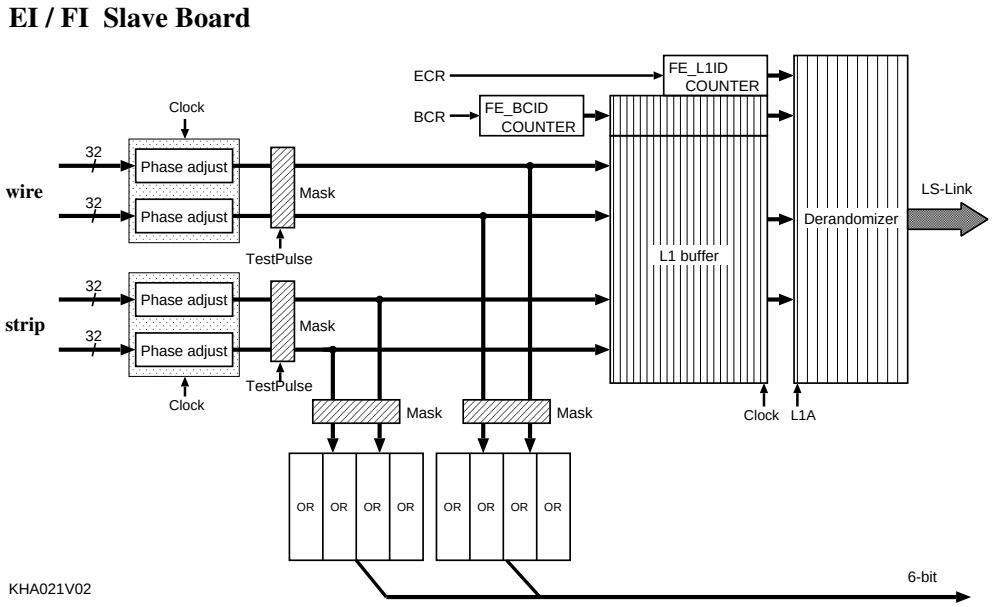


図 7.8: Triplet Slave Board(EI/FI) のブロック図

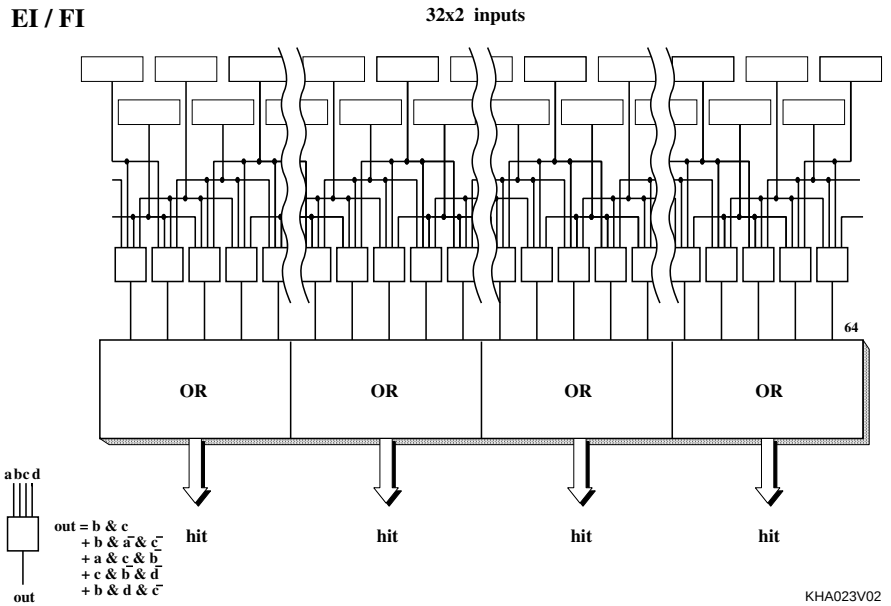


図 7.9: Triplet Slave Board(EI/FI) の論理

7.1.4 Slave Board ASIC に共通の機能

またこの 5 種の Slave Board 機能に共通した処理として以下のようなものがある。

- レベル 1 バッファーと信号読み出し系
  Patch Panel ASIC から受けた信号は、上述の処理が行なわれると共に、レベル 1 トリガーが確定するまでの間、レベル 1 バッファーに貯えられる。レベル 1 バッファーの深さは可変で、またレベル 1 判定を受けた信号はデランダマイザー (derandomizer) に送られる。デランダマイザーは、様々なタイミングで来るレベル 1 判定を受けた信号を整列させる。デタンダマイザーをでた信号は、シリアル信号に変換されて、IC 外に読み出される。

- マスク
  TGC 検出器の一部のチャンネルが故障し、信号を出さなくなるか、粒子が来ていないのに信号を出すといったことが起こった場合、そのチャンネルの処理を無効にするために、マスクをかけることが出来る。マスクは各チャンネル毎に、またコインシデンス処理へ入る信号と、読み出し系へ向かう信号で独立に設定可能で、またマスクの状態も各チャンネル毎に 1,0 で選択可能である。

  Patch Panel ASIC のマスク回路の場合、マスク時の出力は 0 に固定されていたが、Slave Board ASIC 上のマスクは 1,0 の設定が出来るなど柔軟である。これは Slave Board ASIC はコインシデンスやトリガー判定の機能を持つため、1 つのチャンネルの故障が周りのチャンネルの処理に影響を与えるためである。

- 遅延回路 (位相補正回路)
  ASD から Slave Board ASIC の間の配線遅延などを補正するため、また信号とクロックのエッジが重なる場合に対処するため、0 ∼ 1.5 クロックまで 1/2 クロック精度で遅延を与える回路である。この遅延の値は JTAG のレジスタによって制御される。

- テストパルス回路
  テストパルス回路は TTC からの信号を受け、テストパルスを出す回路で、入力部の直後に任意のテストパターンを設定することができる。またテストパルスのトリガー信号は、1 クロック精度の 0∼15 クロックの可変の遅延回路を通して送られる。

- デクラスタリング

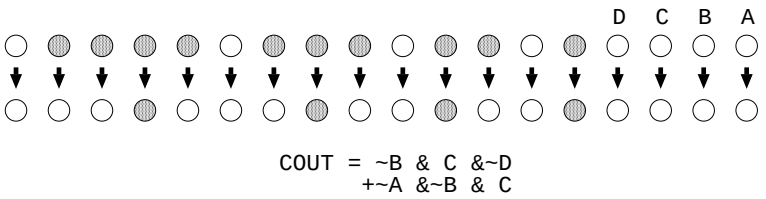


図 7.10: デクラスタリングのアルゴリズム

デクラスタリングは図 7.10 に示すように、連続する複数のヒットがあった場合に、その中から 1 つの信号だけを出力する論理である。

- JTAG プロトコル
  IC 内部の様々なパラメータの設定、バウンダリースキャンを行なうためのプロトコルである。Slave board ASIC では、信号の入力、トリガー及び読みだしの出力にバウンダリースキャンを設置し、またマスクの設定、信号の遅延の設定、トリガーモード (コインシデンスの条件) の設定、レベル 1 バッファーの深さなどの設定をするためにユーザー定義レジスタを搭載する。またこのレジスタには SEU 現象に対処するため、多数決論理回路を用いており、またこの SEU の有無も読みだすことができる。

7.1.5 High-pT ASIC

図 7.11, 図 7.12 に High-pT ASIC のブロック図を示す。wire 用の High-pT ASIC の場合、信号は 3 つの doublet SB と、対応する位置の 4 つの triplet SB から入力を受け、位相の調整後、デコードされコインシデンスマトリッスに入る。マトリックスは doublet から 192(32ch$\times$2$\times$3(SB))、triplet からは δR の分を考慮して 232(20+192+20) 入力になる。図 7.13 がこのマトリックスの 1/4 の部分で、図 7.14 がさらにその 1/8 にあたるマトリックスである。図 7.13 は 2 つの部分にわかれ、それぞれから δR が最小のトラックが探し出される。また δR のウインドウの大きさは-20$\sim$20 まであるが δR が 10$\sim$15 と-10$\sim$-15 の領域は 2 つづつ OR をとることで出力は-15$\sim$15 になる。計 8 個得られるトラックのうち pT の大きな 2 つが選択され Sector Logic に転送される。

また、ケーブル遅延の補正と、信号がクロックのエッジと重なる場合に対処するため、位相補正回路を備えている。また遅延の調節範囲は Slave Board ASIC より大きく 1/2 クロック単位で 8 段階に調節できる。またこの遅延は各 Slave Board ASIC の入力ごとに設定可能である。またこれらの遅延は、JTAG プロトコルによって制御される。

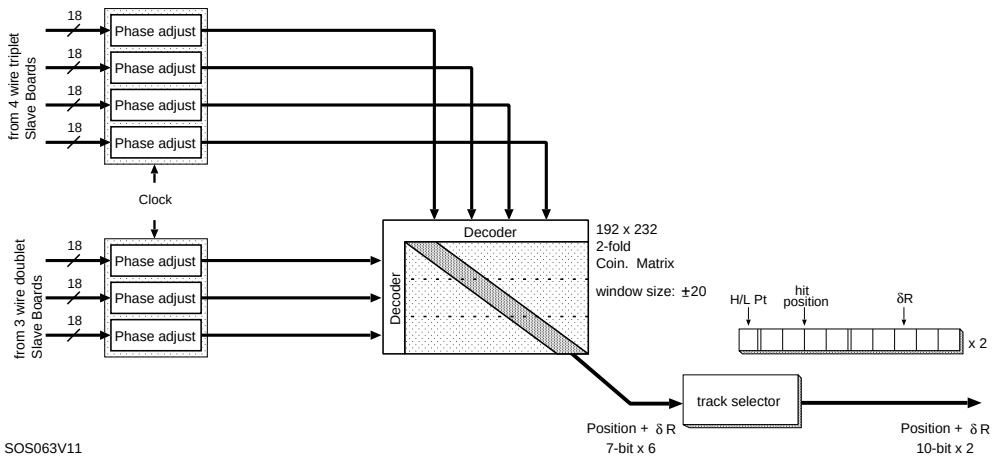


図 7.11: High-pT ASIC のブロック図 (wire)

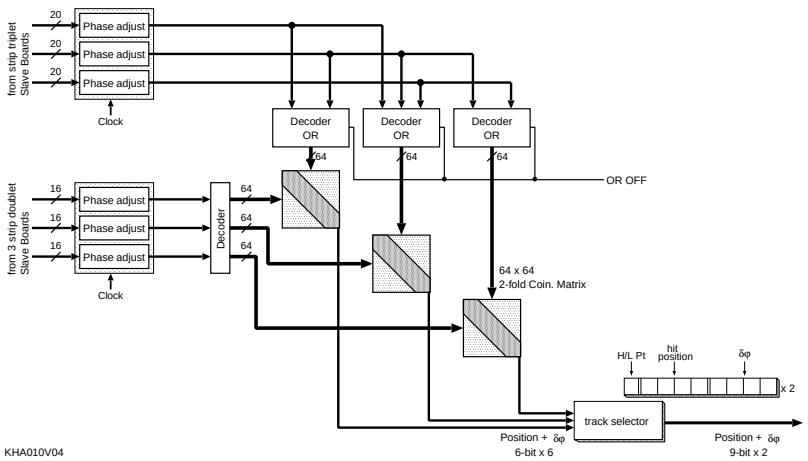


図 7.12: High-pT ASIC のブロック図 (strip)

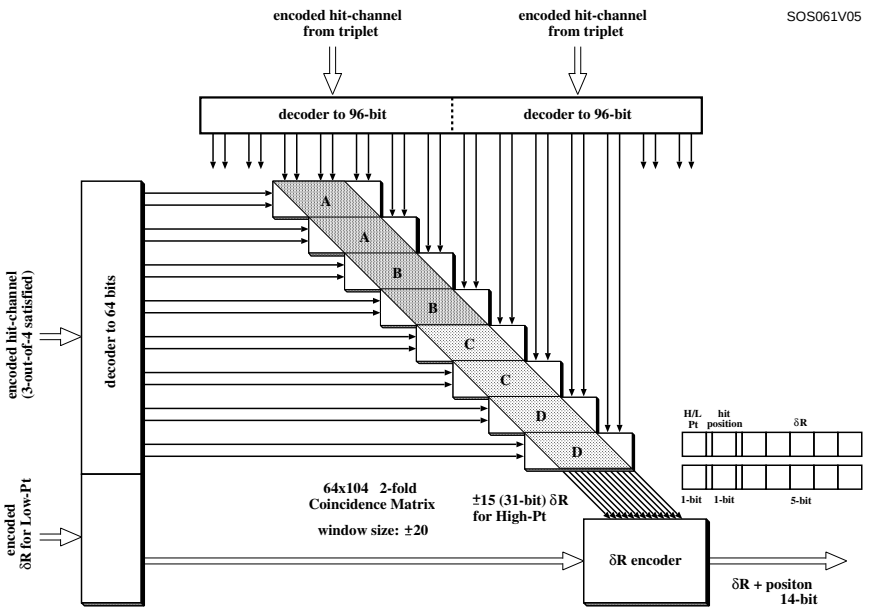


図 7.13: Doublet Slave Board のブロック図

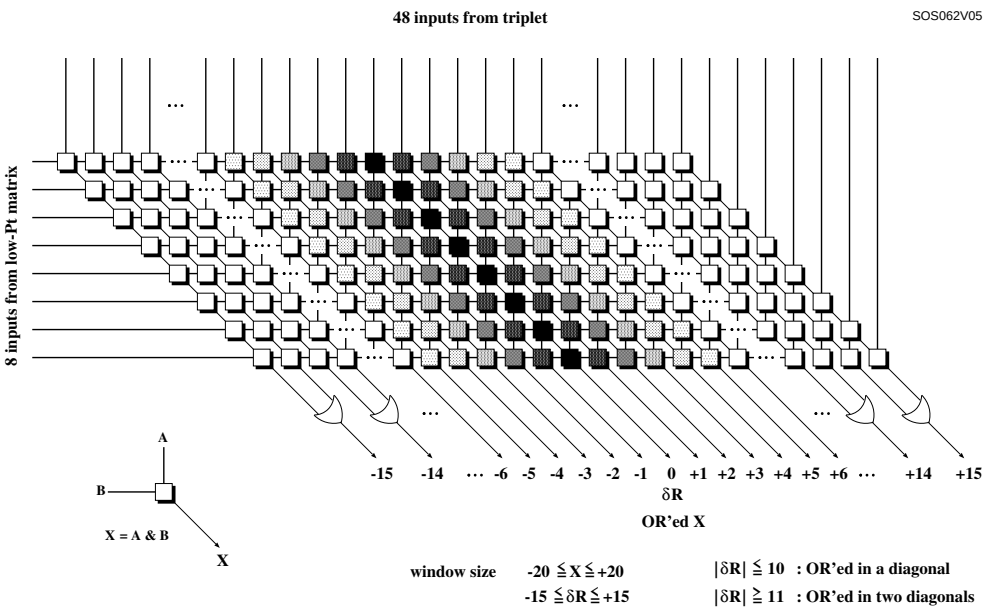


図 7.14: Doublet Slave Board の $low - p_T matrix$ の詳細な構造

7.2 Slave Board ASIC の設計

7.2.1 概要

Slave Board ASIC は、これまでにマトリックス部の回路の一部と、読みだし部に関して、テスト用の ASIC が試作されていたが、[7],[8] 今回、初めて Slave Board ASIC の全ての機能を含む ASIC の設計を行なった。

図 7.15 に Slave Board ASIC 全体のブロック図を示す。Slave Board ASIC は規模が大きくなることから、開発は複数名で分担して行なわれた。今回は主に入力部 (Input part)、マトリックス部 (Matrix part)、読み出し部 (readout part)、制御部 (control part) の 4 つの部分に分けて開発が進められ、各部で動作検証などが行なわれた後に、1 つの ASIC に組み上げるという方法をとった。またここでは著者が主に開発を進めた入力部と制御部に関して重点的に述べていく。

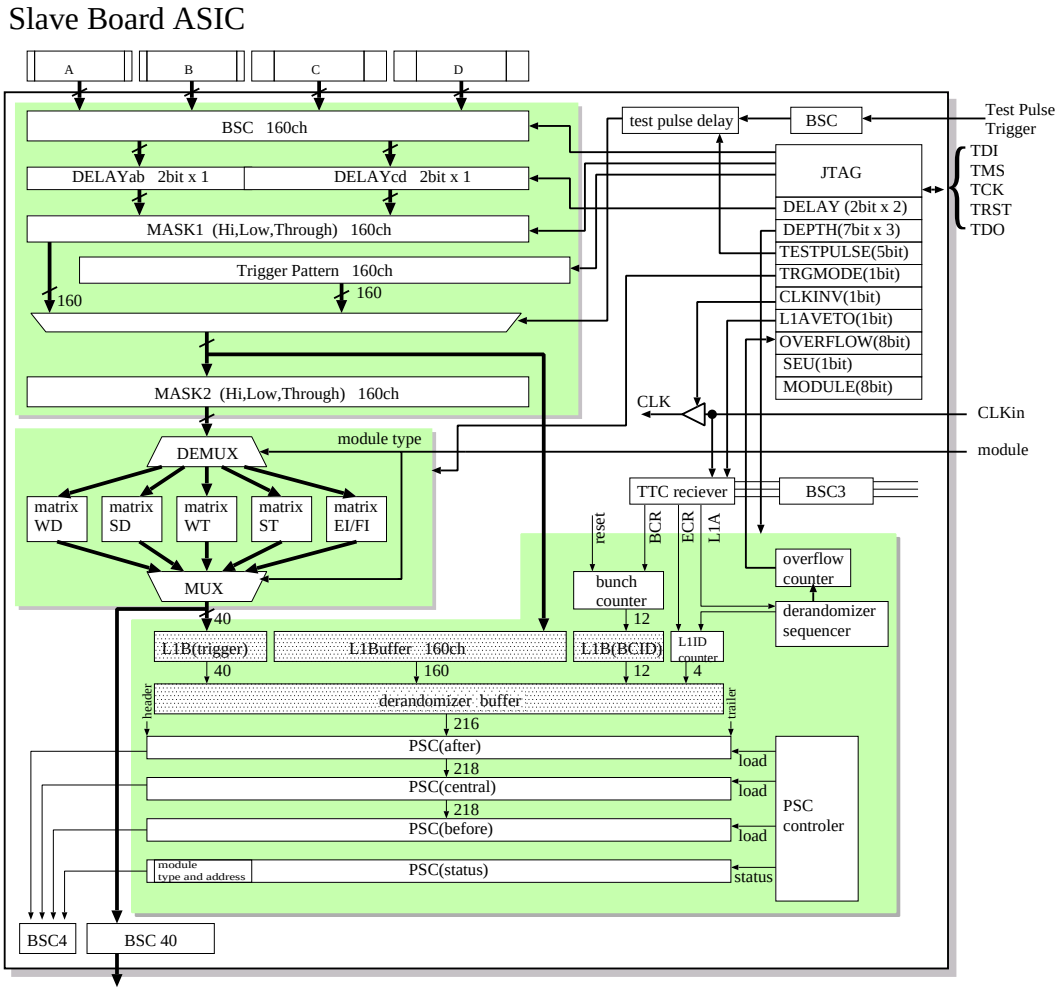


図 7.15: Slave Board ASIC のブロック図

Slave Board ASIC は完全なデジタル回路で開発は主に、HDL コードの記述と、配置配線によってなされる。使用するプロセスはローム社のゲート幅が 0.35μm のフルカスタムチップで、サイズは 9.7mm×9.7mm で、使用可能なピンの数は 256 である。また使用するライブラリはローム社提供のパスポートライブラリで、既成のメモリセルが使用可能なライブラリである。

プロセスによる動作タイミングの違い

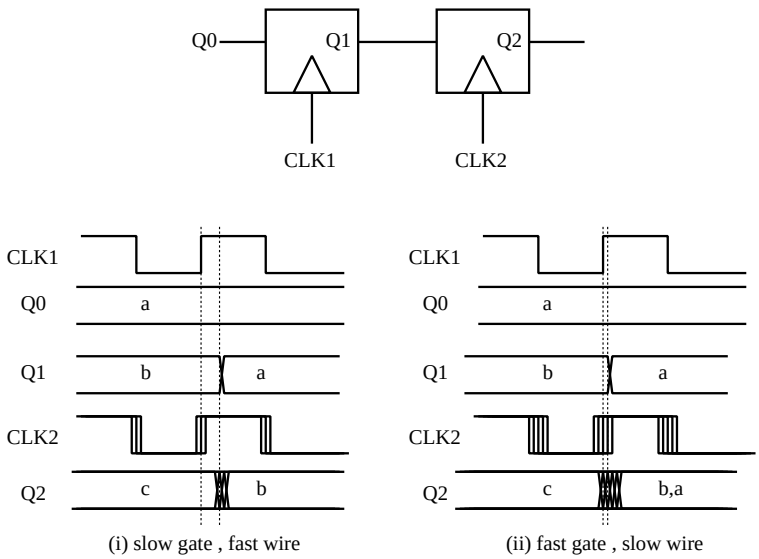


図 7.16: プロセスによるシフトレジスタの動作の違い

近年の半導体プロセスの微細化により、トランジスタ素子の動作速度は早くなったが、配線遅延が大きくなるという問題が発生している。このため回路設計もこの変化を考慮して行なう必要がある。考慮する必要がある典型的な場合として、シフトレジスタの動作タイミングの問題がある (図 7.16)。図は 2 つの D フリップフロップで構成するシフトレジスタで、従来型のプロセス (左) と近年の早いゲート (右) のプロセスによる、タイミングチャートを示したものである。(CLK2 は CLK1 に対するタイミングのずれを示す。) 従来型の場合 D フリップフロップが CLK を受けてから、ラッチしたものを出力するのに 1nsec 程度の遅延があるため、図の回路でシフトレジスタが構成できたが早いプロセスの場合、ゲートの動作が早く、また配線遅延が大きくなるため、CLK が到着する時間にも、ばらつきが生じ 1 段目の D フリップフロップが出力を出すよりも前に、2 段目の D フリップフロップがラッチをかけるということが起こる可能性がある。

Slave Board ASIC の 0.35μm のプロセスもこれを考慮しなければならないプロセスにあたるので以下に述べる回路設計ではこれを考慮して行なった。

7.2.2 入力部

Slave Board ASIC の全信号入力 160 チャンネル分を受ける入力部のうち、1 チャンネル分の回路のブロック図を図 7.17 に示す。入力された信号はまず BSC(Boundary Scan Cell) を通り、図 7.18 の位相補正回路 (遅延回路) に入る。またこの回路の入出力に関するタイミングチャートを図 7.19 に示す。この回路は基本的には、入力信号がクロックのエッジに重なるタイミングで入る場合に対処する回路である。例えばデフォルト設定値の最小遅延 (クロックの上がりエッジでラッチするのみ) のときに、信号の立ち上がりエッジがクロックの上がりエッジと同じタイミングで来る場合、そのクロックでラッチさられるかどうかが不確定になり、同一 IC 内部でもあるチャンネルではラッチされ、別のチャンネルではラッチされず、次のクロックでラッチされるということが起こり得る。このような場合、設定値を変えて一度クロックの逆エッジでラッチをしてから、次の上がりエッジで出力すれば良い。またこの回路は pivot チェンバーと、middle チェンバーの配置

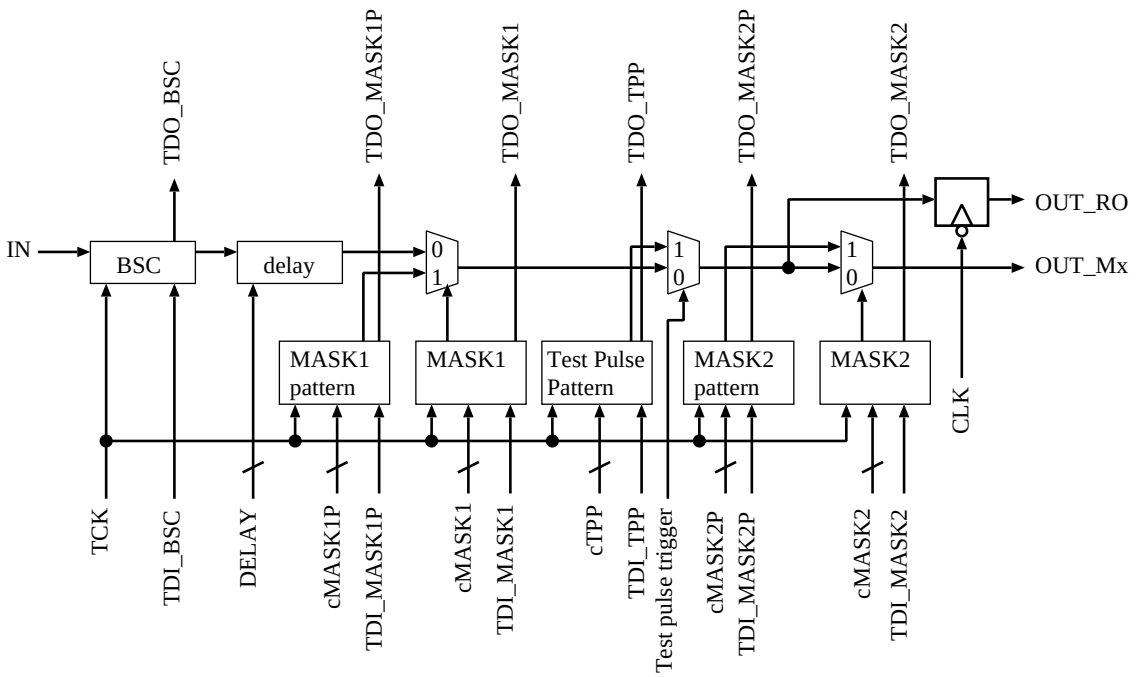


図 7.17: マスク部 1 チャンネル分の回路図

位置の差から生じる可能性のある遅延を補正するために 1.5 クロックまで遅延をかけられるようにもなっている。したがってこの回路の設定は、A 列 B 列と C 列 D 列[1]でそれぞれ共通に設定する。

位相補正回路を出た信号はマスク回路に入る。マスク回路は 2 つの JTAG のレジスタにより値を設定され、マスクの有無 (MASK1) と、マスクのパターン (1 or 0) (MASK1P) が設定できる。またこれは 160 チャンネル各々に設定が可能で、読みだし部とマトリックス部に共通してかかるマスクにした。

信号は次にテストパルス回路を通過する。テストパルス回路もパターン (1 or 0) を設定可能で、160 チャネル各々に設定できるように設計した。テストパルスのトリガーは、1 クロック単位で調節できる 0∼15 クロックの遅延回路を通して、160 チャンネル共通にやってくる。またテストパルスは読みだし部と、マトリックス部に共通して入ることになる。

次に信号は読みだし部とマトリックス部に分かれる。読みだし部へはクロックの下がりエッジでのラッチを通して送られる。この信号は読みだし部では、クロック上がりエッジで受けとられるわけだが、この前に下がりエッジのラッチを入れるのは上述したシフトレジスタのタイミングの問題に対処するためである。一方マトリックス部へはさらにマスク回路を通して送る。これも先のマスクと同様に、マスクパターンとマスクの有無が設定可能で 160 チャンネル各々に設定できるように設計した。

[1] A 列 B 列は pivot チェンバーからの入力チャンネル、C 列 D 列は middle チェンバーからの入力チャンネルである。

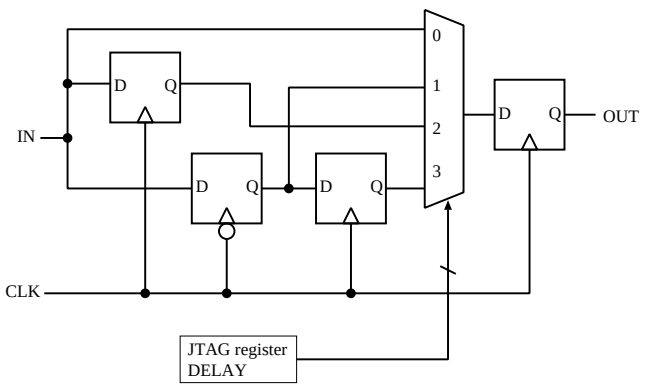


図 7.18: 位相補正回路

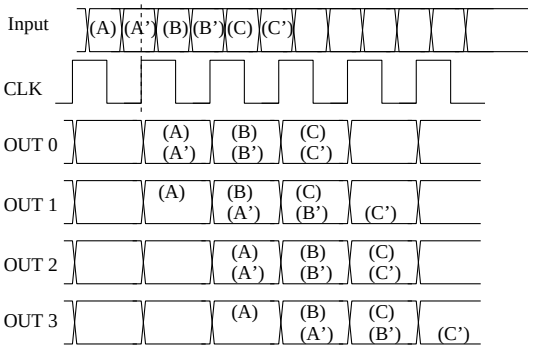


図 7.19: 位相補正回路のタイミングチャート

7.2.3 制御部

制御部は主に、ASIC の各部で使用されるパラメータの設定と読みだしを行なう JTAG プロトコルと、TTC 信号を受信する部分からなる。

Slave Board ASIC における JTAG プロトコル

Slave Board ASIC には表 7.3 に示すような JTAG のユーザー定義レジスタがある。また、パラメータ設定用の RW モードのレジスタには、SEU 現象に対処するための多数決論理回路を搭載した。レジスタには主に、入力部に対して、信号のマスク、遅延を設定するもの、マトリックス部に、トリガー条件の設定を行なうもの、読みだし部に、バッファーの深さを設定するものがある。レジスタの機能に関しては付録 C で詳細を述べる。

BSC(Boundary Scan Cell)

入力側では 160bit の信号入力と、TTC からの信号 (テストパルス、BCR,ECR,L1A) に入力用のバウンダリースキャンセルを搭載した。出力側ではトリガー信号出力 (40bit) と、シリアルの読みだし信号 (4bit) に出力用のバウンダリースキャンセルを付けた。Patch Panel ASIC のバウンダリースキャン機能と合わせて、EXTEST を行なえば、Patch panel ASIC と Slave Board ASIC 間の相互接続テストが可能になる。

TTC 信号の受信

TTC からの LHCCLK を含むタイミング情報は、Slave Board ASIC において、図 7.20 に示すような回路図で受けとられる。図の”CLKin” は TTC から送られてくるクロックで、これは TTC 信号を受ける最初のラッチに使われる。このクロックは EXOR で反転の設定を受け”CLK” として Slave Board ASIC 内部の各処理に使われる。

また図中の 4 つの負論理は前述したシフトレジスタのタイミングの問題に対処するために遅延を意図的に付けたものである。TTC 信号には次のようなものがある。

- L1A

レジスタ名	ビット数	モード	コメント
MASK1	160	RW	Mask after BSC
MASK1P	160	RW	Mask pattern for MASK1
MASK2	160	RW	Mask before DEMUX
MASK2P	160	RW	Mask pattern for MASK2
TPP	160	RW	Test pulse pattern
DELAY	4	RW	delay for signals
DEPTH	21	RW	L1 buffer depth
TRGMODE	1	RW	Trigger Condition
TESTPULSE	5	RW	testpulse delay & veto
L1AVETO	1	RW	veto for L1A
CLKINV	1	RW	CLK invert
RESET	1	RW	RESET
SEU	1	R	SEU flag
MODULE	8	R	Module address & type
OVERFLOW	8	R	derandomizer over flow count
ID	32	R	chip ID(version,factory,,,etc)

表 7.3: ユーザー定義レジスタの一覧

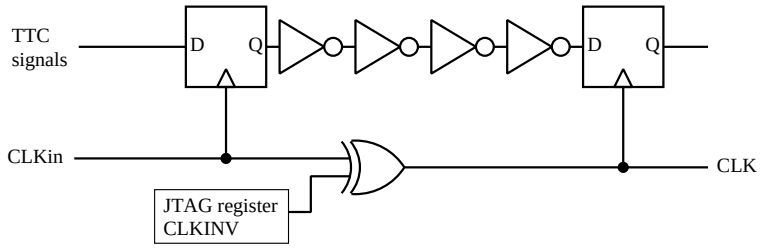


図 7.20: TTC 信号の受信用回路

Level 1 Accept 信号は 図 7.20 の回路の他に、veto をかける回路が入力部分に付けられる。この VETO も JTAG によって設定できる。

- ECR,BCR
  ECR(Event Counter Reset),BCR(Bunch Counter Reset) は図 7.20 の回路で受けられる。
- テストパルストリガーテストパルストリガーは図 7.20 の回路の他に、veto と、1 クロック単位で 0∼15 クロックの遅延を調節する回路を経て出力される。この VETO と遅延も JTAG によって設定される。

クロックの分配

Slave Board ASIC は処理するチャンネルが多いことに伴い、それを制御するクロックも IC 各部で多量に使われる。このため入力されたクロックを、枝状に分岐させて各部に配る必要があるが、このときもタイミングを考慮する必要がある。

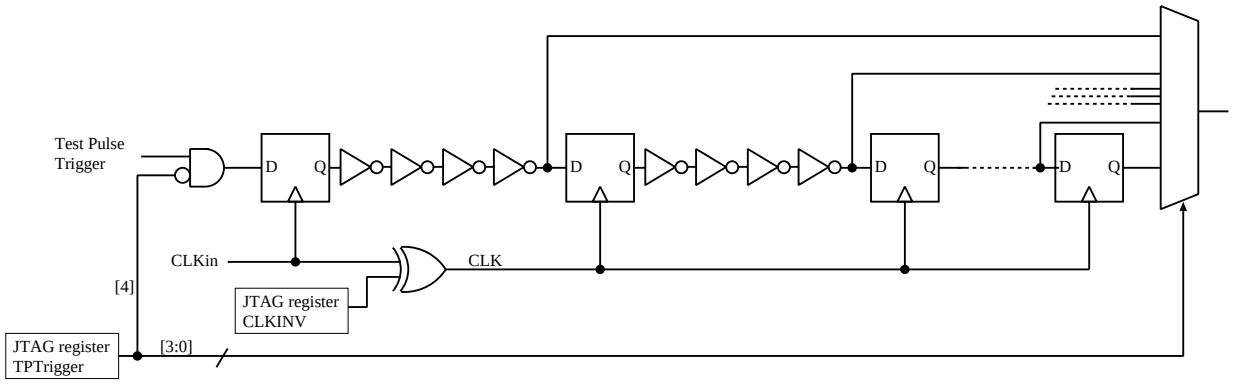


図 7.21: テストパルス信号の受信回路

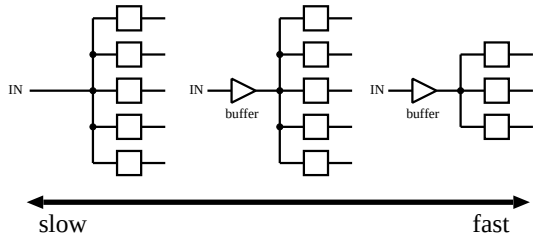


図 7.22: バッファーと負荷による動作速度の違い

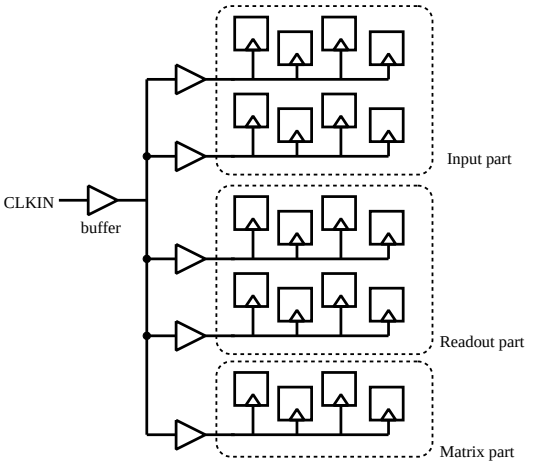


図 7.23: Slave Board ASIC でのクロックの分配の仕方

図 7.22 に示すように、1 本のワイヤから多数の素子を動作させるときは、分岐させる前にチャージのバッファーを付けると、次段の素子の動作が早くなるり、さらに動作させる素子が少なければ、より早くなる。この様にしてクロックの分配の仕方によって、末端の素子の動作のタイミングに差が生じるが、このばらつきを小さくするため、枝の各部分の負荷容量を等しくするなどの工夫をする。Slave Board ASIC の場合は、図 7.22 に示すように、2 段のバッファーを通して末端の素子を動作させ、2 段目のバッファーは 40~60 の素子を動作させるように設計した。

7.2.4 マトリックス部

5 種類の機能があるコインシデンスマトリックス部は、別々に設計され 1 つの ASIC に搭載される。また、これらの機能は基板上の 3bit のスイッチの設定に応じて切替えられる。またコインシデンス条件は 2 つ (通常とオプション) あり、これは JTAG プロトコルによって設定可能である。

7.2.5 読み出し部

レベル 1 バッファー

レベル 1 バッファーには入力信号用 (160bit)、トリガーマトリックス用 (40bit) 、バンチ衝突カウンター用 (12bit) の 3 種類あり、長さは 128bit までで 3 種類それぞれで可変である。この長さは JTAG のレジスタにより設定され、1 種あたり 7bit (128 段階) で、3 種で計 21bit のレジスタの値を受ける。またレベル 1 バッファーは動作中 40MHz のクロックで常に電力を消費することになるので、低電力で動作させるため、デュアルポートメモリーで実現する。

デランダマイザー

デランダマイザーには 1 度のレベル 1 アクセプトで、レベル 1 バッファーの内容を 3 バンチ分書き込み、パラレルシリアルコンバーターが順次読み出す。長さは 128bit で 100kHz のレベル 1 アクセプトに対してオーバーフローしないことを考慮したものである。またオーバーフローがあった場合、これをカウント (8bit のカウンターを備えている) し、これは JTAG で読み出すことが出来る。

パラレルシリアルコンバーター

パラレルシリアルコンバータ (PSC) は 4 本あり、そのうち 3 本がデランダマイザーからのデータをコピーして転送するものである。残る 1 本はモジュールのアドレスや SEU のフラグなど ASIC の状態を転送するために使われる。

7.2.6 シミュレーションによるデザインの検証

Slave Board ASIC を設計するにあたり、verilog HDL の記述もとにシミュレーションによる動作の検証を行ない、各部の動作を確認した。シミュレーションの結果に関しては付録 C で詳細を述べる。

7.2.7 ASIC 全体のレイアウト

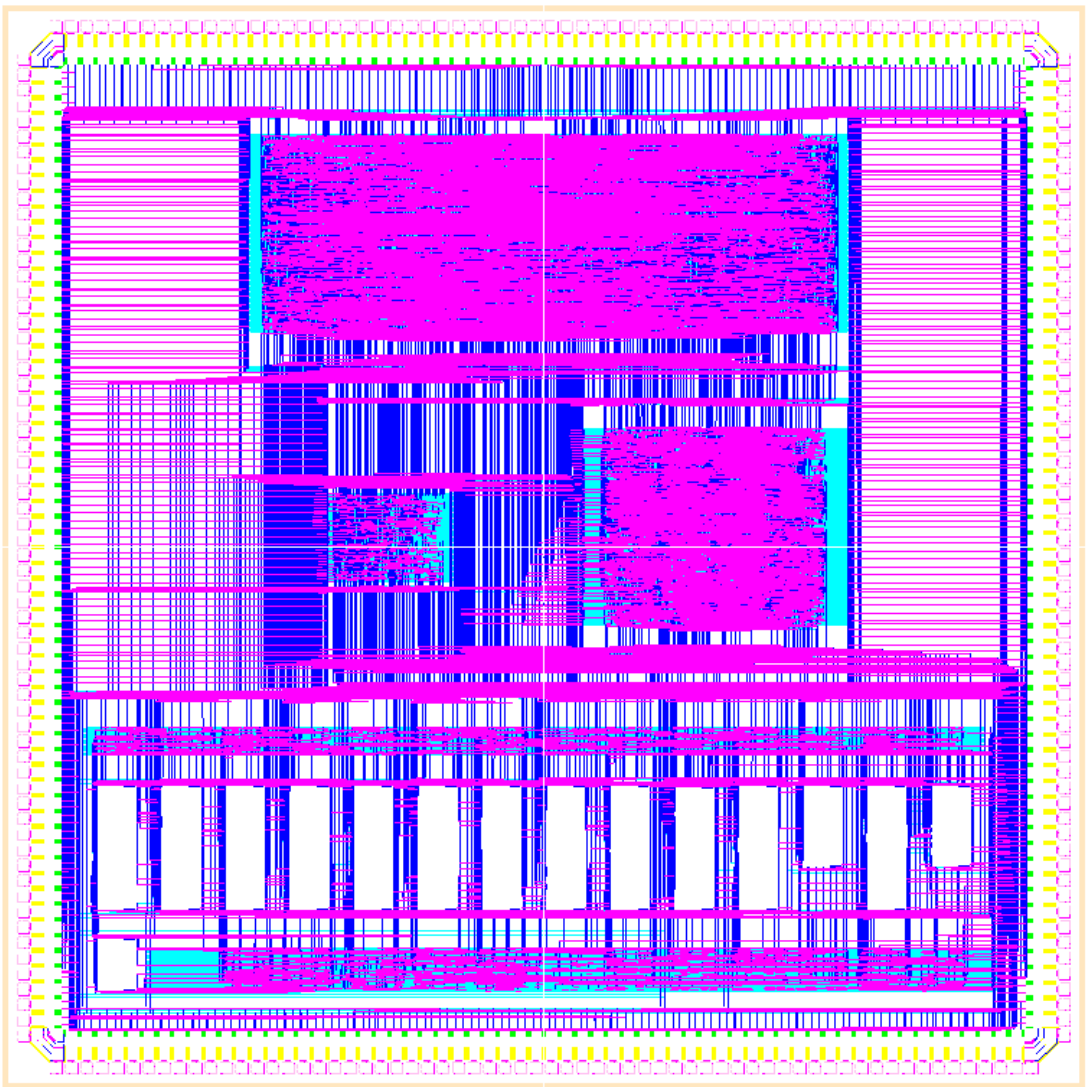


図 7.24: Slave Board ASIC のレイアウト

図 7.24 に今回試作した Slave Board ASIC のレイアウトを示す。これはローム社のゲート間隔 0.35μm のフルカスタムチップを使って試作したもので、サイズは 9.7mm×9.7mm である。

これは、主に 4 つの部分からなり上段の大きな塊が入力部、中央左が JTAG などを含む制御部、中央右が 5 種類のコインシデンスマトリックス、下段がデュアルポートメモリーを含む読みだし部である。またピン配置は上辺、右辺、左辺が主に信号の入力に当てられ ABCD の各列[2]で互い違いに入力を受ける。また下辺はトリガー出力や読みだし、制御用に当てられる。

[2] A 列 B 列は pivot チェンバーからの入力チャンネル、C 列 D 列は middle チェンバーからの入力チャンネルである。

7.3 High-pT ASIC の設計

High-pT ASIC はこれまで何度か試作が繰り返されており、完成段階にあるといえるが、これまで信号の遅延などの、制御パラメータを設定するプロトコルが搭載されていなかったため、今回著者がこの制御部分の開発を担当し JTAG プロトコルによって、パラメータの制御を可能にした。

High-pT ASIC もまた Slave Board ASIC と同様に完全なデジタル回路で開発は主に HDL コードの記述と論理合成によってなされる。使用するプロセスは日立のゲートアレイ (ゲート間隔 0.35μm) である。

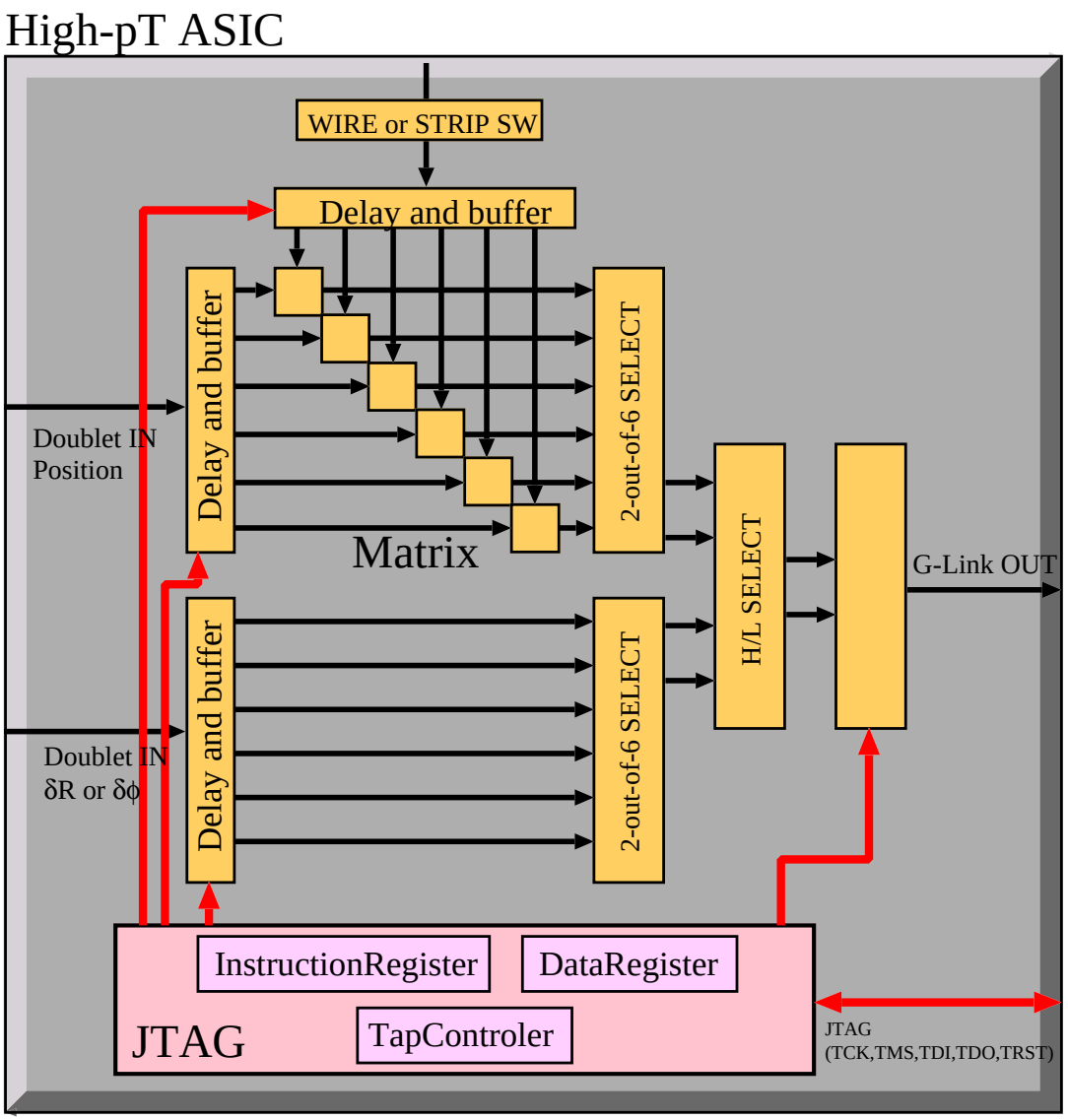


図 7.25: High-pT ASIC のブロック図

図 7.25 に High-pT ASIC のブロック図を示す。High-pT ASIC は信号処理部と制御部分 (JTAG 部分) からなりここでは著者が開発を担当した制御部に関して述べる。

7.3.1 制御部分

High-pT ASIC は表 7.4 に示すユーザー定義のレジスタによって制御される。

レジスタ名	ビット数	モード	コメント
DELAY0	3	RW	DELAY for triplet1
DELAY1	3	RW	DELAY for triplet2
DELAY2	3	RW	DELAY for triplet3
DELAY3	3	RW	DELAY for triplet4
DELAY4	3	RW	DELAY for doublet1
DELAY5	3	RW	DELAY for doublet2
DELAY6	3	RW	DELAY for doublet3
GLINK	3	RW	G-LINK control
SEU	1	R	SEU flag read out

表 7.4: ユーザー定義レジスタの一覧

High-pT ASIC もまた放射線環境下に設置されることから、これらのうち RW モードのレジスタには SEU 対策として多数決論理回路を搭載している。また今回 High-pT ASIC に搭載した JTAG は後述する JTAG コードを自動生成するスクリプトを用いて設計されたものである。

7.3.2 シミュレーションによる動作検証

High-pT ASIC の制御部分を設計するにあたり、全てのレジスタへのデータの書き込み、読み込みのテストを行ない、正しく動作することを、確認した。詳細は付録 C で述べる。

7.3.3 High-pT ASIC のレイアウト

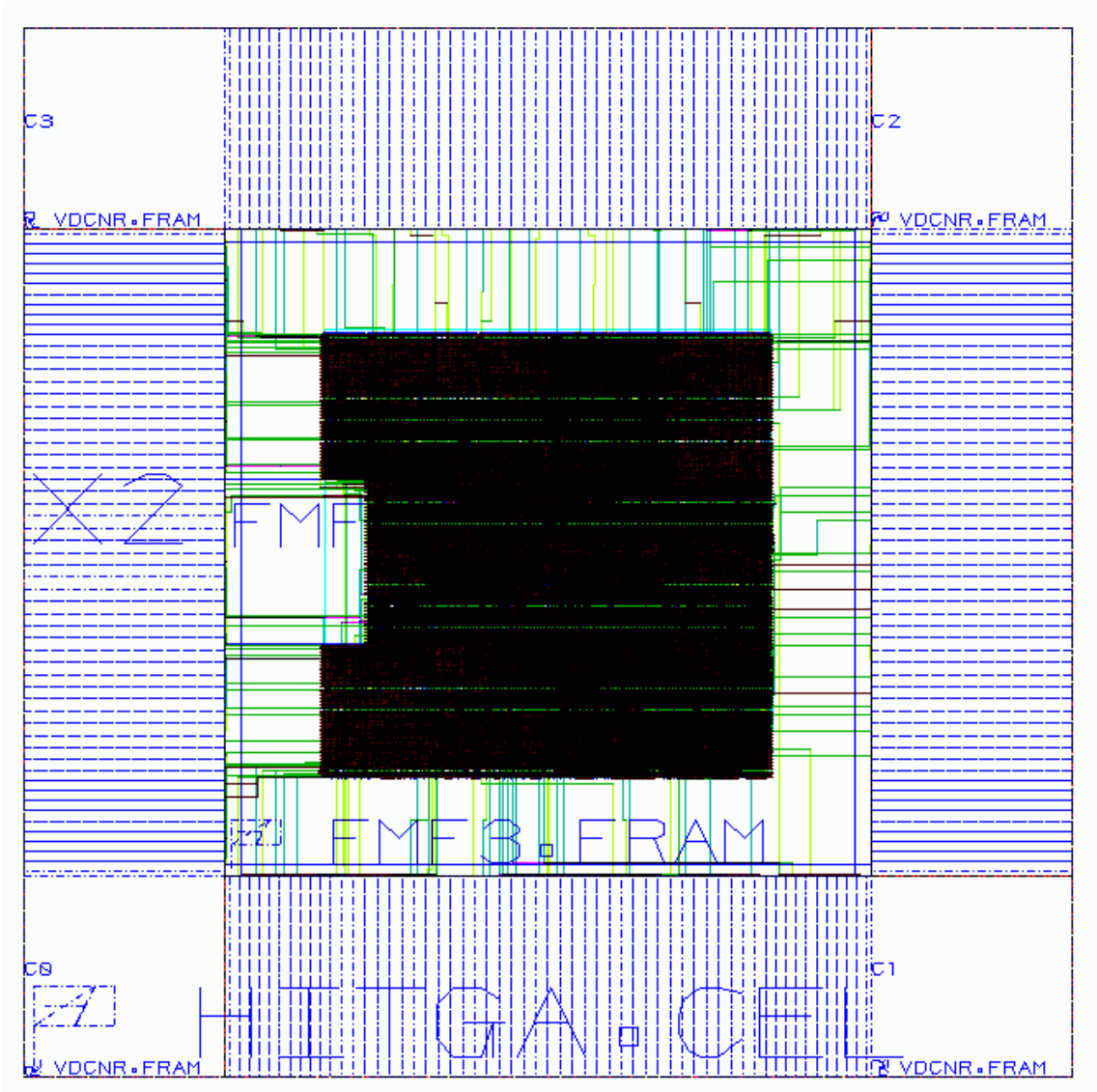


図 7.26: high-pT ASIC のレイアウト (日立ゲートアレイ 0.35μm)

これまで TGC エレクトロニクスで開発する ASIC は、フルカスタムチップを用いていたが、今回の High-pT ASIC の開発で始めて、ゲートアレイを使って ASIC 開発を行なった。ゲートアレイは論理素子が予め配置されているチップに、配線を施すことによって作る IC で、半導体メーカーは予め配線前までのチップを大量に用意しておき、ユーザーの要求にしたがって配線を行なうことでチップをつくる。このため短期間、低コストで IC をつくることができる。またその反面、アナログ回路や Slave Board ASIC で使ったようなメモリを含む回路には向かない。High-pT ASIC の場合は全てデジタル回路でメモリなどの特殊な回路を含まないのでゲートアレイでの開発が向いている。

図 7.26 にゲートアレイで実現される High-pT ASIC のレイアウトを示す。

7.4 Slave Board ASIC, High-pT ASIC に関するまとめ

Slave Board ASIC は Patch Panel ASIC からの信号を受けて、コインシデンス処理、low-pT の判定を行ない、これを High-pT Board に向けて出力する ASIC である。またレベル 1 のトリガー判定が出るまでの間、信号を貯えるレベル 1 バッファーも搭載しており、レベル 1 アクセプト信号を受け次第バッファーの値を読み出し、スタースイッチに転送する。Slave Board ASIC は主に信号のマスクや遅延の設定やテストパルスパターンを設定する入力部、low-pT 判定を行なう機能ごとの 5 種類のコインシデンスマトリックスを持つマトリックス部、レベル 1 バッファーと読み出し用のパラレル-シリアルコンバーターを備えた読み出し部、また IC 各部のパラメータの設定、TTC からのタイミング信号を受信するための制御部からなる。今回、著者は入力部と制御部に関して、様々な動作確認のためのシミュレーションを行なうなどして開発を行なった。

この ASIC はローム社のサイズが 9.7mm×9.7mm でゲート間隔が 0.35μm のフルカスタムチップを使って実現され、2001 年 1 月に VDEC に提出され、春頃に納品される予定である。また今回の Slave Board ASIC の試作は全機能を含むものとしては始めてのもので、今後は IC が納品されしだい徹底的な動作検証を行ない修正点を洗い出していく必要がある。

High-pT ASIC は Slave Board ASIC の次段の処理を行なう部分で、doublet チェンバーと triplet チェンバーの信号を比べて High-pT の判定を行なう部分である。High-pT ASIC はこれまでに何度か試作が繰り返されており完成段階にあると言えるが、これまで信号の遅延などの制御パラメータを設定するプロトコルが搭載されていなかったため、今回著者がこの制御部分の開発を行ない、JTAG でこれらのパラメータを制御できるようにした。また High-pT ASIC は、日立のゲート間隔 0.35μm のゲートアレイで試作を行なった。最終的に High-pT ASIC はコスト面で有利なゲートアレイで実現される予定である。またこの ASIC も 2001 年春以降に納品される予定で、その後も徹底的な動作検証が求められる。

第8章　放射線耐性試験用ICと放射線照射試験

先に述べたようにTGCの信号処理用のASICは放射線環境下に設置される。ここではその放射線のうちγ線の影響を調べるためのASICとγ線照射試験に関して述べる。

8.1　概要

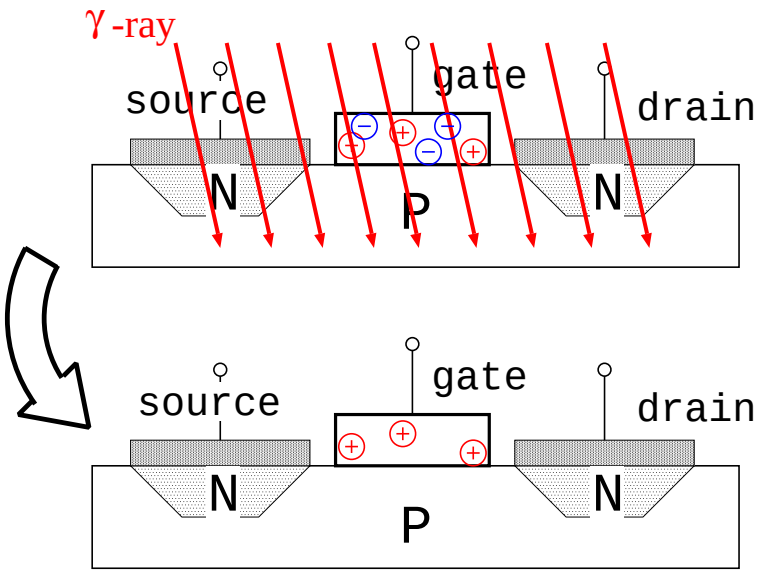


図 8.1: γ線が半導体に影響を与える仕組み

γ線は半導体プロセス中のゲート部分に用いられるポリシリコンに影響を与える。ICにγ線が入るとポリシリコン中に電子と正孔のペアが生成され、移動度の違いからそのうち正孔のみが残りこれがその半導体のパラメータに影響を与える。

今回のテストではこのパラメータの変化を見るために単体のCMOSとリングオシレーターを搭載したICを設計した。テスト対象のプロセスはPatch Panel ASIC,JRC ASICに用いられるローム社のゲート幅0.6μmのフルカスタムチップである。

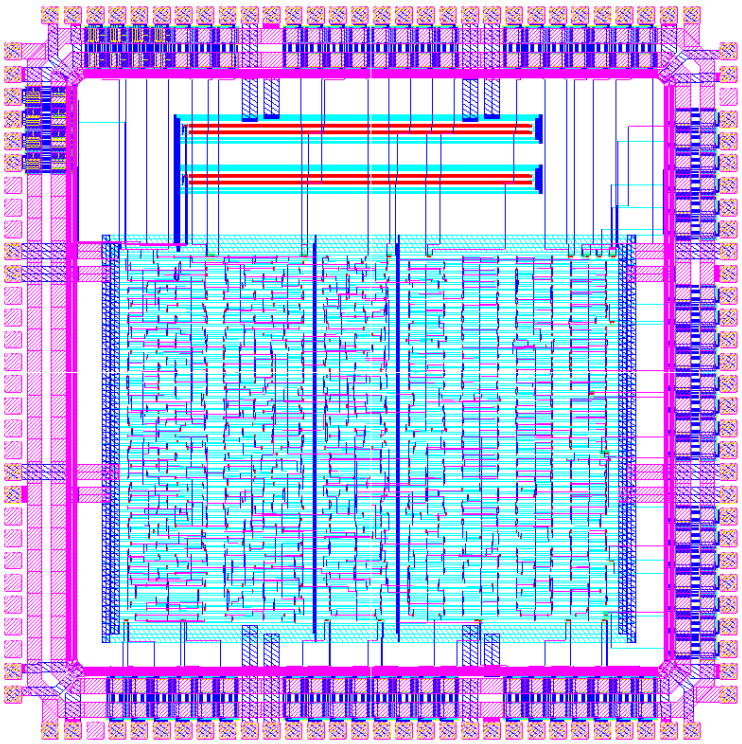


図 8.2: 放射線耐性試験用 IC のレイアウト

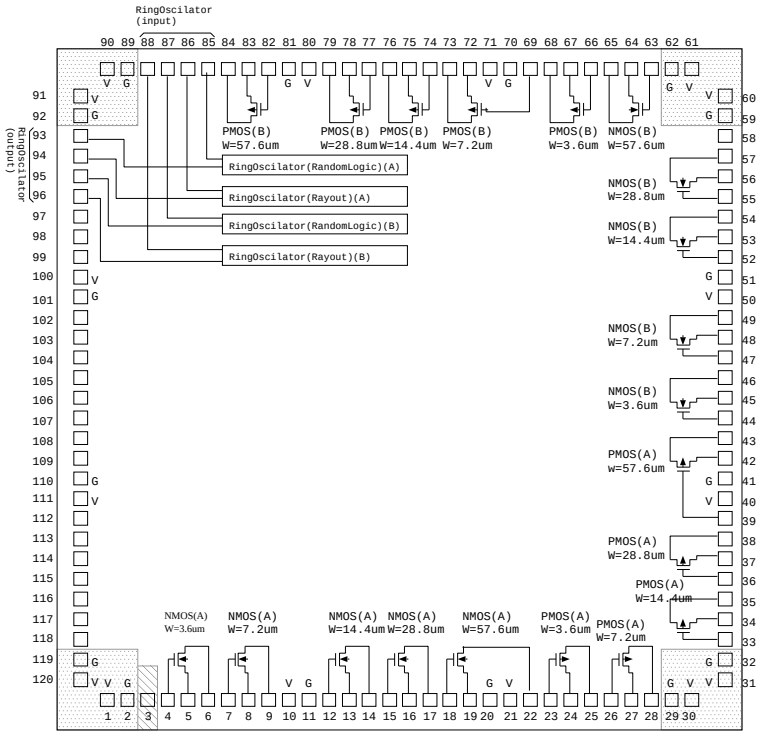


図 8.3: 放射線耐性試験用 IC のブロック図

8.2 放射線試験用 IC

放射線耐性試験用のチップのレイアウト、ブロック図を図 8.2, 図 8.3 にそれぞれ示す。チップの内容は以下に示すとおりゲート幅 (W) の異なる NMOS,PMOS それぞれ 5 種類で計 20 個の CMOS と、リングオシレータのレイアウトで記述したものと自動配置配線で記述したものである。

- 単体の CMOS
  - PMOS(W=3.6 μm)
  - PMOS(W=7.2 μm)
  - PMOS(W=14.4 μm)
  - PMOS(W=28.8 μm)
  - PMOS(W=57.6 μm)
  - NMOS(W=3.6 μm)
  - NMOS(W=7.2 μm)
  - NMOS(W=14.4 μm)
  - NMOS(W=28.8 μm)
  - NMOS(W=57.6 μm)

  数量：各 2 個/1chip

- リングオシレーター
  - レイアウトで書いたもの
  - ランダムロジックで書いたもの

  数量：各 2 個/1chip

8.2.1 単体 CMOS

単体の CMOS のレイアウトの例を図 8.4 に示す。(図は PMOS の W が 57.6μm のものである。) この CMOS から外部に測定用のワイヤとして source, drain, gate がそれぞれ取り出される。また source と drain はデザインの上では対称で区別がない。またゲート電極には静電気の影響でトランジスタが壊れることを防止しすための保護回路として抵抗が付けられている。この抵抗はポリシリコンをワイヤ状に配置したものでこのプロセスの場合 1μm 四方のポリシリコンは 8Ω の抵抗になるのでこれを利用して約 100Ω の抵抗にしている。またバックゲートは PMOS,NMOS でそれぞれ電源電極、グランド電極に設置されている。電源、グランドは IC 内部で共通である。
またこの IC の場合、トランジスタのパラメータを正確に計測する必要があるので通常は IO パットに設置内部に設置されている保護回路は取り除いていいる。図 8.5 にこの IO パットのレイアウトを示す。図の左が通常の保護回路付きの IO パットで、右が保護回路を取り除いた IO パットである。

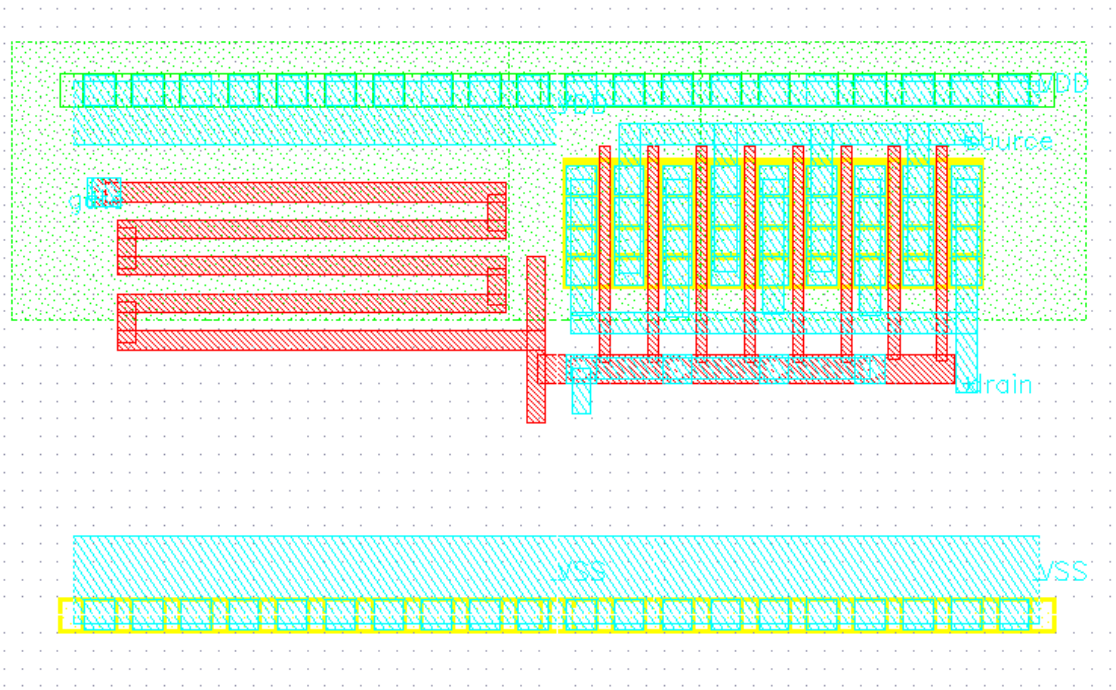


図 8.4: 単体 CMOS のレイアウト。図はゲートの幅 (W) が 57.6μm の PMOS

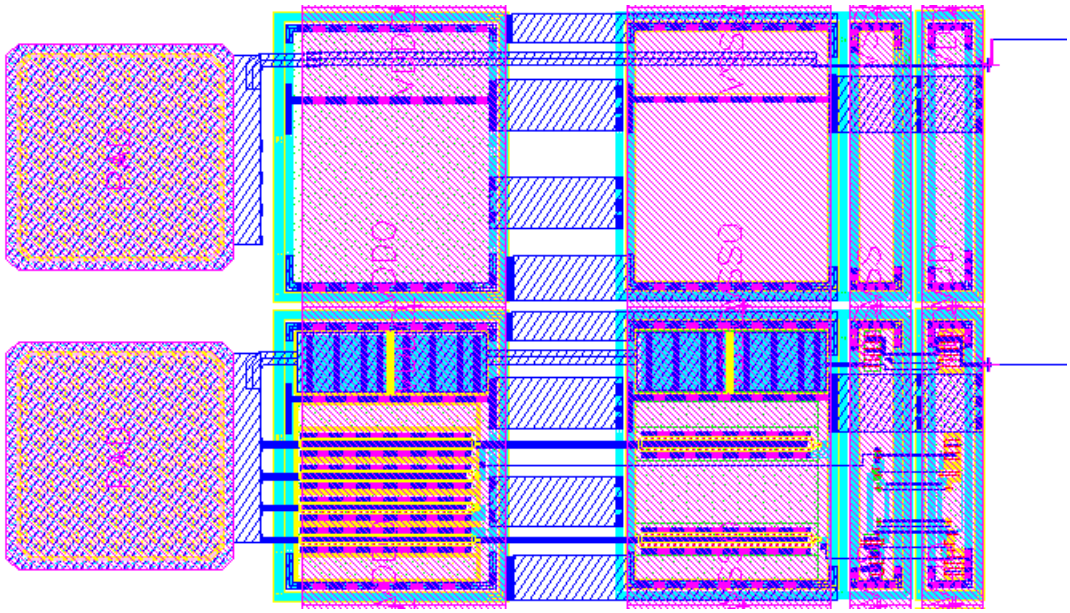


図 8.5: 電源パット、下が通常のパット、上が単体 CMOS の IO になるパット

8.2.2 リングオシレーター

リングオシレーターは負論理を奇数個リング状に配置した発振回路でここでは放射線の照射によりこの発振周波数の変化を見るために設置する。図 8.6 にリングオシレータの回路図を示す。この回路の場合は負論理を片側を HI にした NAND 論理によりリングオシレーターを構成している。単なる負論理回路の場合、リングの中に 1 つ付けなければならない読みだし部分で容量のバランスが異なるために発振のタイミングのバランスも異なることが考えられる。このためリングを NAND 論理で構成し各 NAND の出力部にはもう 1 つ負荷を与えるための NAND 論理を取り付けそのうちの 1 つを読みだしに使っている。またリング上の NAND の常に HI にするワイヤを LOW にすると発振は止まるのでこれは発振回路のスイッチになる。

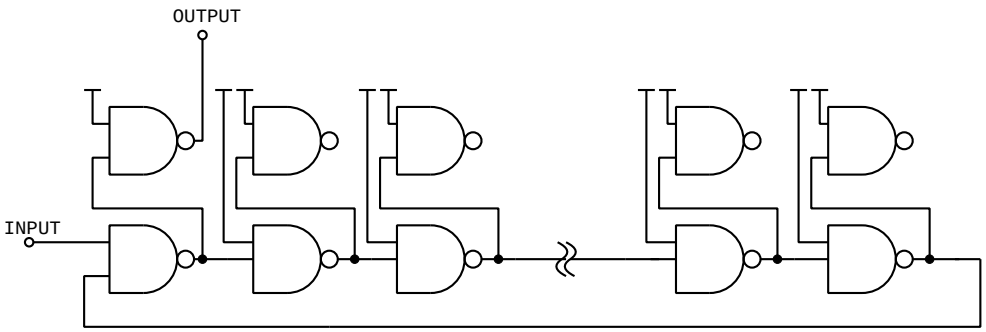


図 8.6: リングオシレーターの回路図
リングオシレーターは負論理を奇数個リング上に配置したものであるが、読みだし部分での負荷の違いを無くすため NAND 論理の素子を配置し負論理とした。また今回はこの負論理を 255 個配置した。

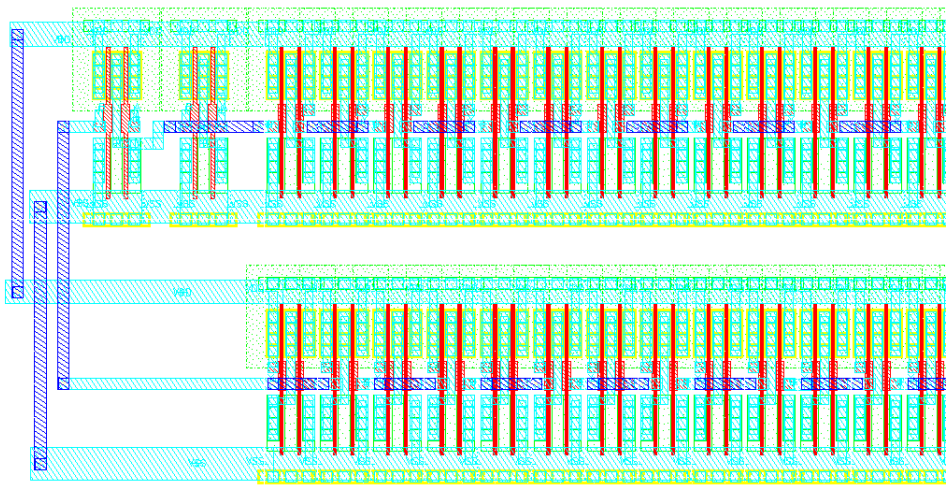


図 8.7: リングオシレーターのレイアウトの一部

8.2.3 照射試験用ボード

図 8.8 にテスト用のボードの写真を示す。図のように 4 つの IC を配置している。この IC の場合、通常は搭載している保護回路を外しているため静電気の影響を考慮してボード上で単体の CMOS のピンを短絡させるようなスイッチを付けている。

図 8.8: 照射試験用のボード
4 つの ASIC を搭載する各 ASIC の両サイドに取り付けられているのが、スイッチボードである。これによってバイアスの有無を決め、またトランジスタの各端子をグランドと短絡させ静電気による破損を防ぐ。

8.3 放射線照射試験

3 章でも述べたように TGC の M1 レイヤーでは 10 年間に 0.23[krad](=2.27[Gy]) の吸収線量が見積もられており安全係数をかけて 15.9[krad](=159[Gy]) までの照射後の動作確認が求められている。そこで今回は前節で述べた IC に対して東京大学原子力研究総合タンターにおいて 100[krad] までの γ 線の照射試験を行なった。

照射対象となるのは前で述べた IC4 つで、以下の回路を含むものである。

- 単体の CMOS
  それぞれ幅の異なる PMOS40 個、NMOS40 個で、合計 80 個の CMOS
- リングオシレータ
  自動配置配線によるもの 8 個と、レイアウトレベルから記述したもの 8 個で、合計 16 個のリングオシレーター

8.3.1 γ 線照射時のセットアップ

照射は東京大学原子力研究総合タンターの生物照射装置を用いて行なった。線源は 69.5[TBq] の ^{137}Cs で、線源からの距離で照射量の調節を行なうものである。照射のセットアップを図 8.9、図 8.10 に示す。写真右の円筒型のものが、線源の容器である。線源はセットアップを行なう時など、人が照射室内にいるときは、地下に収められており、照射時には室外からそれを制御して図の容器に収められる。今回、テストボードは線源容器から 23cm の位置 (容器の半径は 7cm なので中心からの距離は 30cm である。) に設置し、5krad/h で照射を行なった。この距離を短くすればより高い照射率で照射できるが、照射にムラができるためこの距離で行った。

図 8.11 に照射時のテストボードの回路図を示す。リングオシレータは ON の状態で発振させながら照射した。また CMOS の内の半数のもの (PMOS20 個と NMOS20 個) にはバイアスとして、40MHz のクロックを入力しながら照射を行なった。またこのバイアスの有無は、テストボード上のスイッチボードによって切替えられる。スイッチボードは図中の太線部分で、各 CMOS のソース、ドレイン、ゲートの接続を制御するもので、バイアス用とグランド用がある。また NMOS のバックゲートは VDD に接続され、PMOS のバックゲートはグランドに接続した状態で照射を行なった。またボードには $\pm$5V の電源を供給している。

今回の照射試験では合計で 100krad までの照射を行なったが、0, 0.5, 1, 5, 10, 50, 100[krad] の各照射量に対して、IC のパラメータの測定を行なった。

通常の IC は静電気などの影響で、IC 内のトランジスタの破損を防ぐための保護回路が搭載されているが、今回の放射線照射試験用の ASIC は、トランジスタのパラメータを測定するためのものなので、この保護回路を取り除いて設計している。そのため、このためテストボードの取り扱いには注意が必要になる。今回は照射施設とパラメータ測定を行なう場所との間など、テストボードを運搬する際に図 8.12 に示すような回路でトランジスタの保護を行なった。保護用の回路では図のように各 CMOS のソース、ドレイン、ゲートを短絡させ、それらの間に電位差が生じないようにしている。この設定もまたスイッチボードで行なっている。

図 8.9: 照射時のセットアップ (写真)
円筒型のものが線源の容器。これに向かい合わせるようにテストボードを設置している。線源容器までの距離は 23cm で 5krad/h で照射される。

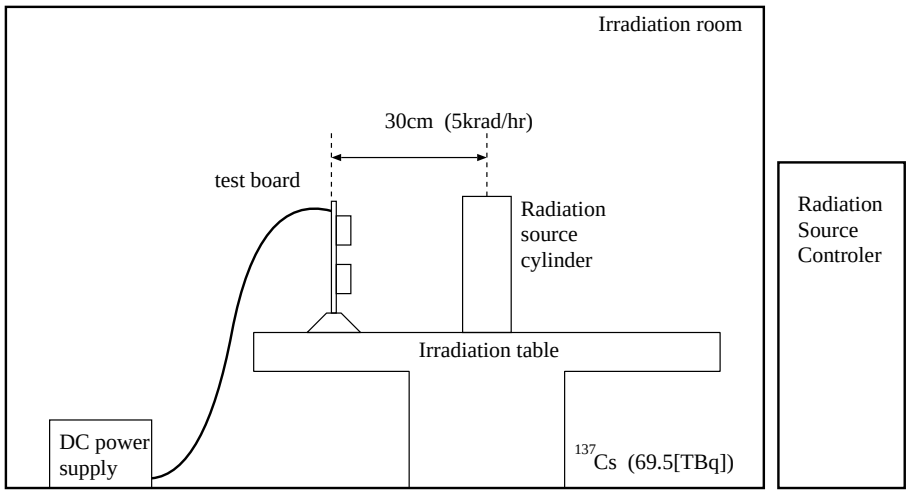


図 8.10: 照射時のセットアップ
線源容器の中心から 30cm の位置に、容器に向かい合わせるようにテストボードを設置する。照射室内で人が作業をするときなどは、線源は地下に格納されており、照射時には、照射室外の制御装置で、線源を照射台上の容器に収める。

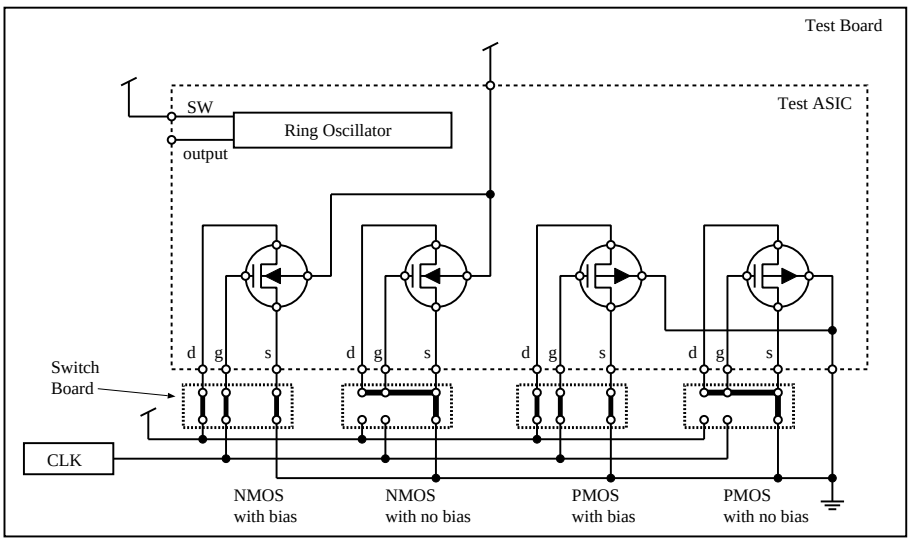


図 8.11: 照射時のセットアップ (回路図)
半数の CMOS にバイアスとしてクロックを入力しがなら照射を行なう。これはスイッチボードで設定される。

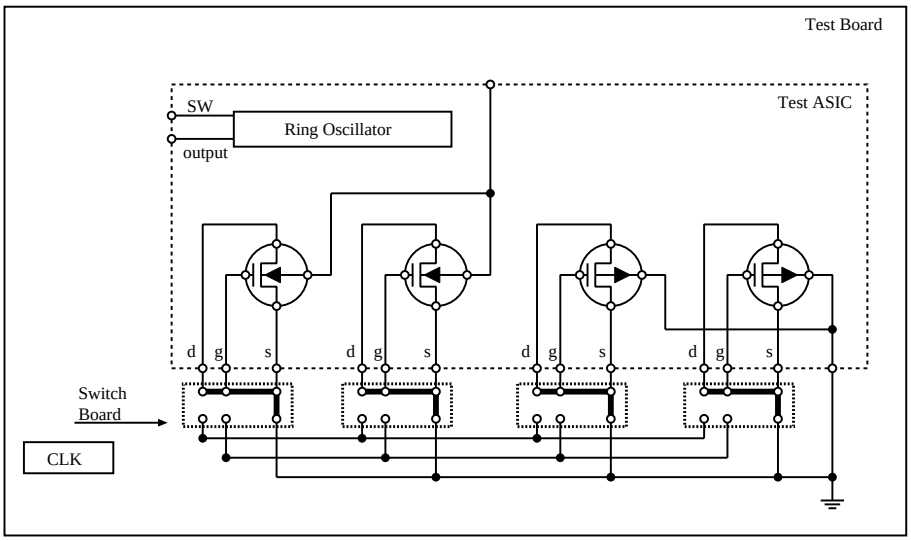


図 8.12: テストボード運搬時の配線
テストボードの運搬時には静電気などによりトランジスタの破損を防ぐため CMOS の各端子を短絡させる。この設定はスイッチボードによってなされる。

8.3.2 測定系のセットアップ

今回の照射試験では 100krad までの照射を行ない 0, 0.5, 1, 5, 10, 50, 100 [krad] の各照射量に対して以下の測定を行なった。

- 単体 CMOS
  測定は半導体パラメータアナライザーを用いて図 8.13 に示すようなセットアップで行なった。ソース、バックゲートをグランドへ、ゲート電極、ドレイン電極を可変電圧の電源に接続した。
  NMOS の場合、V_{DS}（ドレイン、ソース間電圧）を 5[V] にし V_G（ゲート電圧）を-1∼5[V] の領域で測定した。PMOS の場合は V_{DS} を-5[V] で V_G を 1∼-5[V] の領域で測定した。

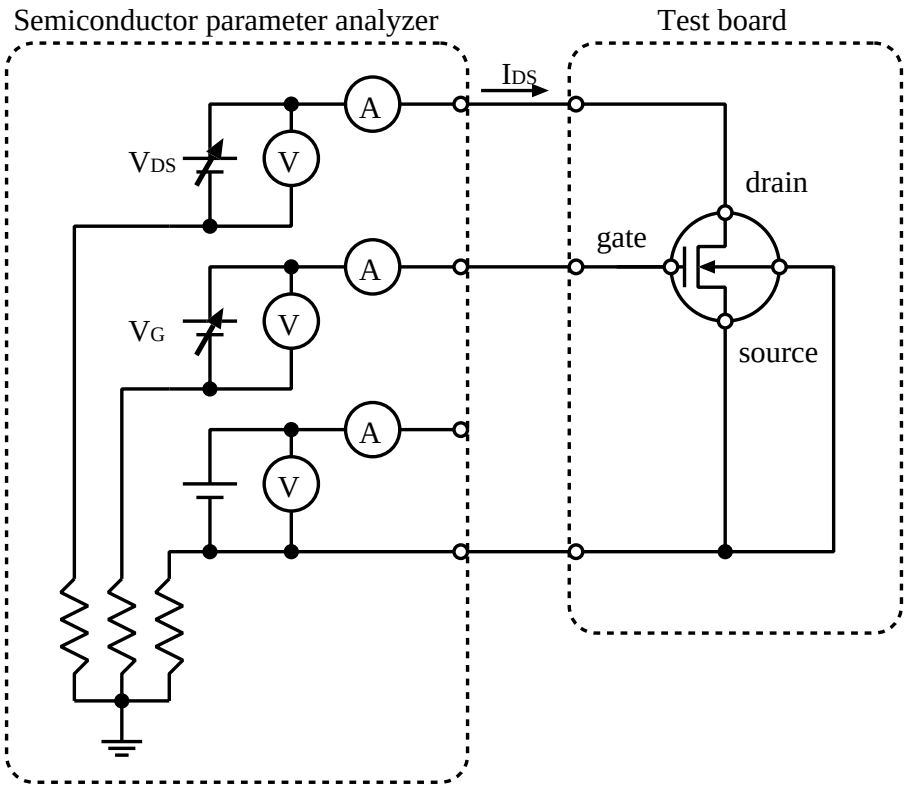


図 8.13: 半導体パラメータアナライザーを使った測定系のセットアップ

- リングオシレーター
  発振周波数を測定した。

8.3.3 測定結果

単体 CMOS

図 8.15 図 8.16 に照射前の V_G の変化に対する I_{DS} をプロットした（上段が線形プロット、下段が対数プロット）。
100[krad] の照射後、線形プロットでは有意な変化が見られなかったが、対数プロットの V_G が 0[V] の周辺で I_{DS} に変化が見られた。そこで V_G が 0[V] での I_{DS}（リーク電流）を照射量に対してプ

ロットした (図 8.17)。また 50[krad] と 100[krad] の照射の後には、約 24 時間のアニーリングの後にも測定を行なっている。また図中の赤線 (濃線) がバイアスありで照射した CMOS、緑線 (淡線) がバイアスなしで照射した CMOS である。

NMOS の場合、照射量 10[krad](100[Gy]) まででは有意な変化が無く、50[krad] でも 10^{-7}[A] 程度しかリーク電流が無いことが確認出来る。またプロット中の 4 つの NMOS に関して、電流値が 1mA 程度まで大きくなっているものがあるが、これらはテストボードの運搬中、あるいは測定中に静電気の影響で破損したものと考えられる。また PMOS の場合に関しては 100[krad] の照射に対してもリーク電流が 10[nA] 以下に押えられているのがわかる。

また PMOS,NMOS いずれの場合に関しても、バイアスがある方が、リーク電流が大きくなる傾向にあることが確認できるが、これはゲートポリシリコンには、バイアスがかかる方が正孔が残りやすいことを示している。また 50[krad] と 100[krad] の照射後には、24 時間の経過後にも測定を行なったが、リーク電流の減少が確認できる。

リングオシレーター

照射量に対する発振周波数の変化を図 8.14 に示す。50[krad] までの照射では有意な変化は見られず、100[krad] の照射で 2%程度の発振周波数の低下が認められるのみである。リングオシレータの発振周波数は各素子の動作速度を反映しており、アナログ回路を含む Patch Panel ASIC などでは、重要なパラメータであるが、この結果から γ 線の照射が、動作速度に影響を与えることが無いことが確認出来る。(発振周波数の 2%の変化は 1 つの素子の動作速度にすると 10psec 程度の変化に対応するので、これは無視できる。)

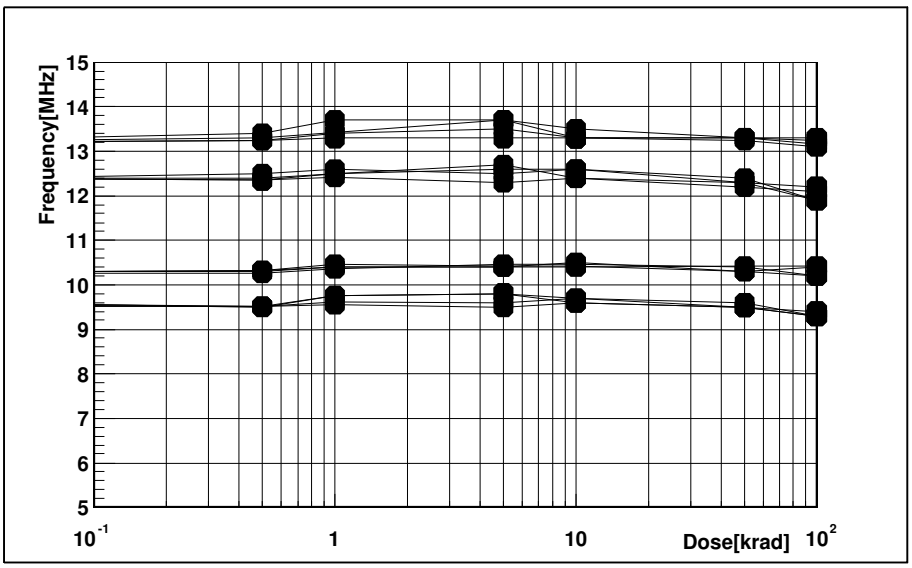


図 8.14: γ 線照射量に対するリングオシレーターの周波数発振周波数が 4 段階に分かれているが、上 2 つが自動配置配線によるもので、下 2 つがレイアウトレベルから設計したものである。それぞれの 2 つは基本的には等しい回路であるが、寄生容量などのために生じたものと考えられる。

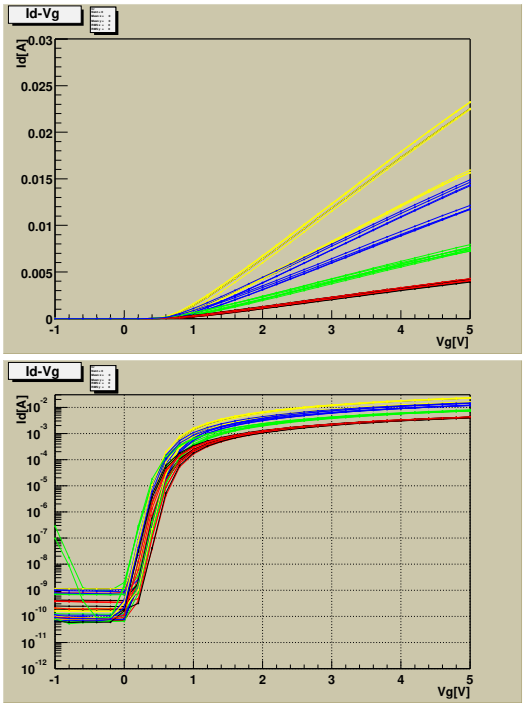


図 8.15: PMOS の動作特性 (Id-Vg) 色分けは W (ゲートの幅) に対応する。

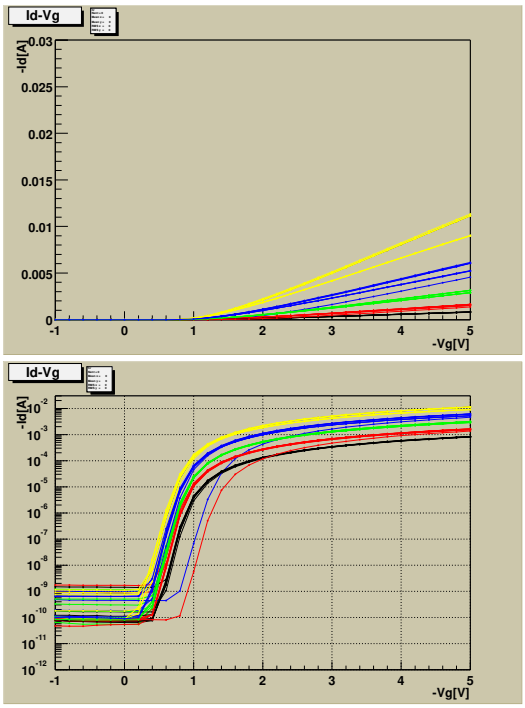


図 8.16: PMOS の動作特性 (Id-Vg:対数プロット) 色分けは W に対応する。

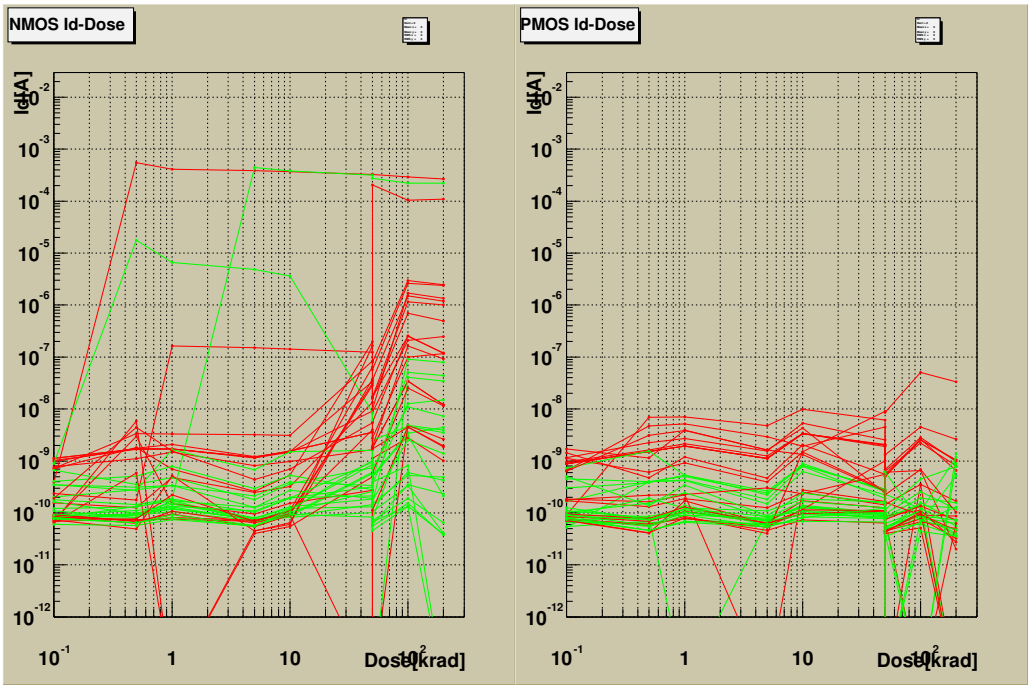


図 8.17: γ 線照射量に対するリーク電流の変化
図中の赤線 (濃線) がバイアスありで照射した CMOS、緑線 (淡線) がバイアスなしで照射した CMOS である。また 50[krad],100[krad] の照射の後には 24 時間のアニーリングを行なっている。

8.4 放射線耐性試験用 IC と放射線照射試験に関するまとめ

ATLAS 検出器は、強い放射線環境下に設置されることになるため、検出器の各構成要素は各々の環境下で、10 年間の実験中安定した動作をすること確認する必要がある。TGC エレクトロニクスを構成する ASIC も、放射線環境下での影響を確認する必要があり、今回は幾つかある放射線の影響のうち TID(Total Ionising Dose) の影響を測定するため、Patch Panel ASIC と JRC ASIC に用いられる予定である、ローム社のゲート間隔 0.6μm のプロセスを使って、トランジスタのパラメータの変化を見るための IC をつくり、γ 線の照射試験を行なった。

IC にはゲート幅の異なる NMOS,PMOS と、リングオシレータを配置し 4 つのチップ (CMOS は計 80 個、リングオシレーターは 16 個) に対して照射テストを行なった。CMOS は半導体のパラメータを計測するために、リングオシレーターはその発振周波数が、論理素子の動作速度を反映しているので、この動作速度を計るために搭載した。TGC のエレクトロニクスが設置される場所では 10 年間で 2.27[Gy](=2.27×10^2[rad]) の TID があることが見積もられており、安全係数をかけて 1.59×10^2[Gy] の照射後の、動作の確認が求められている。そのため今回は東京大学原子力科学研究センターにおいて、10^3[Gy] までの照射を行ない半導体のパラメータの変化を測定した。

測定結果では 10^2[Gy] までの照射では半導体のパラメータ、動作速度に有意な変化は見られず、10^3[Gy] までの照射で、わずかなリーク電流と速度の低下が見られたのみであった。また照射後のアニーリングにより、変化したパラメータも、もとの値に戻る傾向にあることを確認した。これによって ATLAS 実験 10∼20 年間中に照射される γ 線量に対して、これらのプロセスが安定して動作することを確認したことになる。

今後はさらに実際に実験に使用する ASIC を使って、同様の照射試験が必要である。また TID のみでは無く SEE(Single Event Effect)、NIEL(Non Ionising Energy Loss) の影響をテストする必要がある。

第9章　全体のまとめ

ATLAS 実験では 40MHz という高頻度でバンチの衝突が起こり、膨大な数のイベントが生成される。本論文ではこの膨大なイベント中から、重要なイベントを迅速に且つ正確に選択し、データ収集を行なうために重要な、TGC ミューオントリガーシステムと、その 30 万以上のチャンネルからなる信号を処理するための IC の開発に関して述べた。

信号処理の初段にあたり、タイミング調整を行ない、信号をクロックに同期させる Patch Panel ASIC の開発では、様々な動作シミュレーション、デザインの検証を行ないデジタルアナログ混在の IC を開発した。この Patch Panel ASIC の動作テストでは、これまでで初めて、正しく且つ十分な余裕を持って動作する DLL 回路の特性を得ることができた。また LVDS レシーバーの動作も確認し、本実験でも十分に使える回路をつくることが出来た。またこれまでの試作 ASIC の結果と合わせると、これで Patch Panel ASIC 上の全てのアナログ回路の動作を確認したことになる。

また次段の運動量の判定を行なう Slave Board ASIC では信号入力部、また IC 全体の制御を行なう制御部の開発を行ない、その最初のプロトタイプ ASIC において、シミュレーションでそれが、十分に動作することを確認した。また high-pT 判定を行なう high-pT ASIC では、主にその制御部分の開発を行ない、ほぼ完成段階にあたる ASIC の動作をシミュレーションで確認した。

さらに放射線環境下での、IC の安定した動作を確認するための ASIC をつくり、放射線照射テストを行い、10∼20 年間の ATLAS 実験での放射線に対応する照射に対し、IC が安定して動作することを確認した。

付 録A　JTAGプロトコルに関する補足

ここでは、やや専門的な事項である JTAG プロトコルの詳しい動作の説明を行ない、JTAG の verilogHDL コードのサンプルを示す。

A.1　JTAG プロトコルの動作例

ここでは HDL シュミレーションの結果に基づいた波形を用いて、JTAG の動作を説明する。またサンプル回路として図 A.1 に示すような、入力側の BSC が 4bit, 出力側の BSC が 4bit, また RW モードの 4bit のユーザー定義レジスタ regA,regB を含む回路を用いる。また説明に使う波形は実際に verilog HDL のシュミレーションにより生成されたものある。ユーザーレジスタへのデータの書き込み読み込み、また BSC を使った EXTEST,SAMPLE/PRELOAD の例に関して説明する。

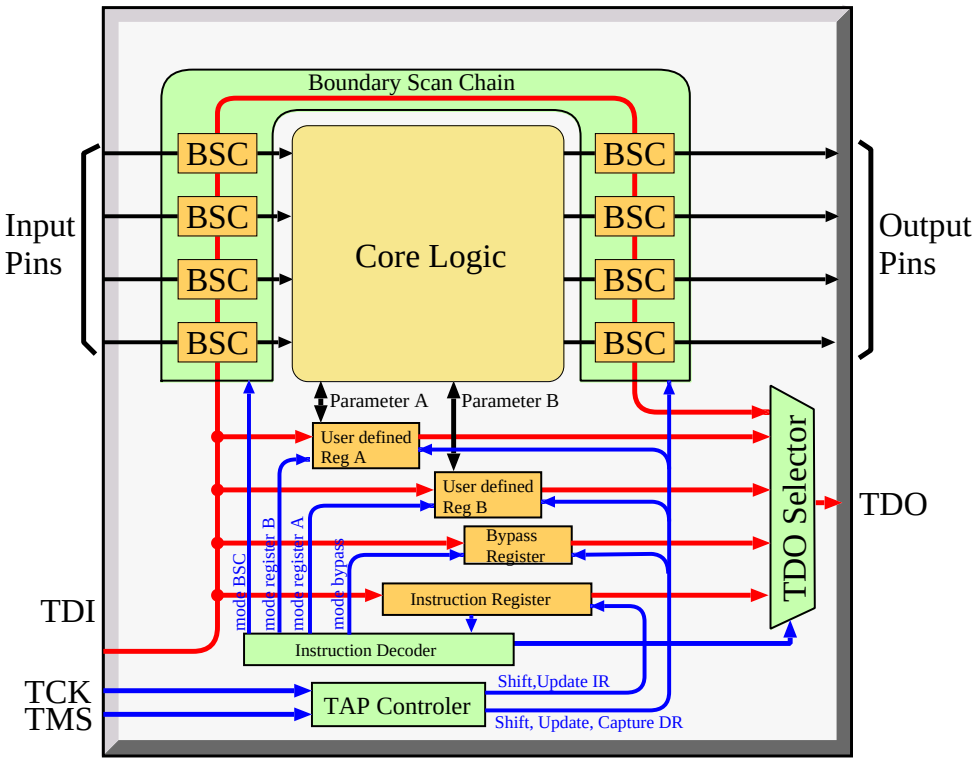


図 A.1: サンプルの構成
JTAG の動作を説明するためのサンプル回路として BSC(出力側) を 4bit,BSC(入力側) を 4bit、RW モードのユーザー定義レジスタ、regA(4bit),regB(4bit) を含む回路を用いる。

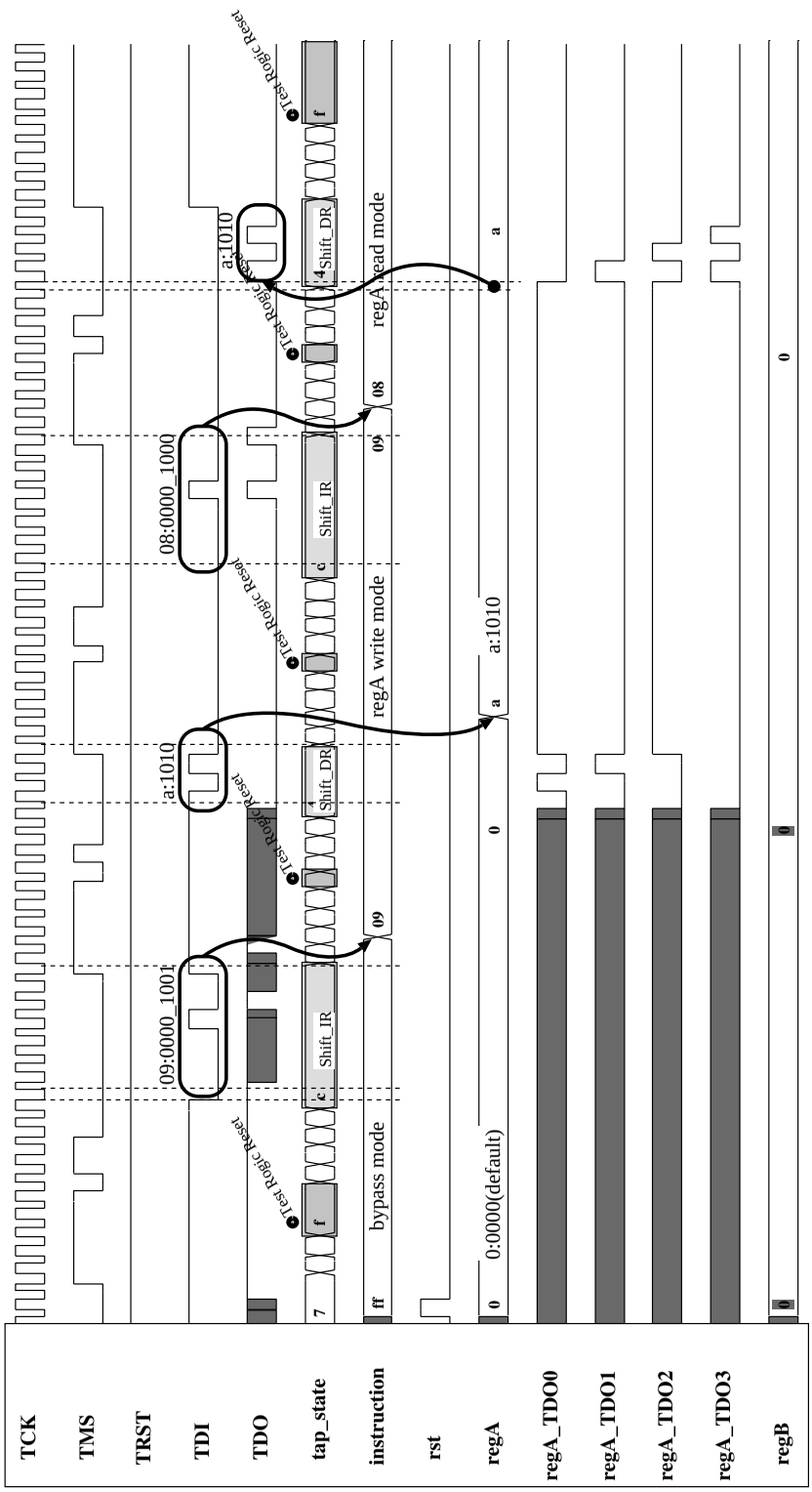


図 A.2: JTAG プロトコルの動作波形 (データレジスタの読み書き)

A.1.1 データレジスタの読み書き

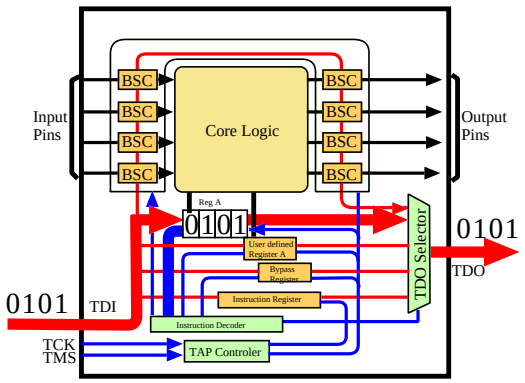


図 A.3: ユーザーレジスタの読み書き

まず実際の ASIC の使用に際して最も重要なユーザーレジスタへのデータの書き込み、読みだしに関して述べる。ここでは 4bit のユーザーレジスタ regA に 4bit のデータ 1010 を設定し、それを再び読み出すという動作に関して説明する。波形は図 A.2 に示す。

リセット　ユーザー定義レジスタにはデフォルト値が設定されており RST 信号 (TRST とは違う。) を入力することでこの値にリセットできる。この JTAG 回路は正論理の RST 信号で、これによって regA、regB がともにデフォルト値の 0000 になり、インストラクションレジスタは 1111_1111(BYPASS) になる。

書き込み命令の設定　ユーザーレジスタ regA にデータを書き込むにはまずインストラクションレジスタに regA の書き込み命令 (Write mode) を設定する。Test Logic Reset の状態にある tap state を TMS に 01100 とパターンを入れることで Shift IR の状態にして、シフトの間 TMS を 0 にしておく。データは tap state が Shift IR になった次の TCK の上がりエッジから取り込まれる。またこのとき TDI からのデータは TCK の下がりエッジで IC 内に入ってくる。regA の Write mode のインストラクションコードである 0000_1001 を順次 TDI から入力していく。このコードの最終 bit をシフトさせるときは TMS を 1 にして次の TCK では最終 bit の読み込みと同時に Shift IR 状態を抜ける必要がある。
ここで TMS を 1 にして 5TCK 分いれて tap state を Test Rogic Reset に戻す。またこの過程で tap state は Update IR を通過するのでこのときにインストラクションコードが更新される。

ユーザーレジスタへのデータの書き込み　書き込み命令の設定によりレジスタ regA が動作可能になった。レジスタ書き込みの動作も命令の設定と同じ要領でできる。Test Logic Reset の状態にある tap state を TMS に 0100 と入れることで Shift DR の状態にして、シフトの間 TMS を 0 にしておく。命令の設定のときと同じタイミングでパターン 1010 を入力し、tap state を Test Rogic Reset に戻す。この場合でも途中で Update DR を通過しているのでこのときにレジスタの値が更新される。

読み込み命令の設定　ユーザーレジスタ regA のデータを読み出すためインストラクションレジスタに regA の読み込み命令 (Read mode) を設定する。書き込み命令の設定と同じようにて、インストラクションコード 00001000 を書き込む。

ユーザーレジスタからのデータの読みだし　データ書き込み時と同じように tap state を Shift DR にして 4TCK で読み出す。データがシフトレジスタに読み込まれるのは Shift DR 状態の前の状態である Capture DR のときの TCK 上がりエッジである。TDO はこの上がりエッジの直後の下がりエッジでデータを出力する。

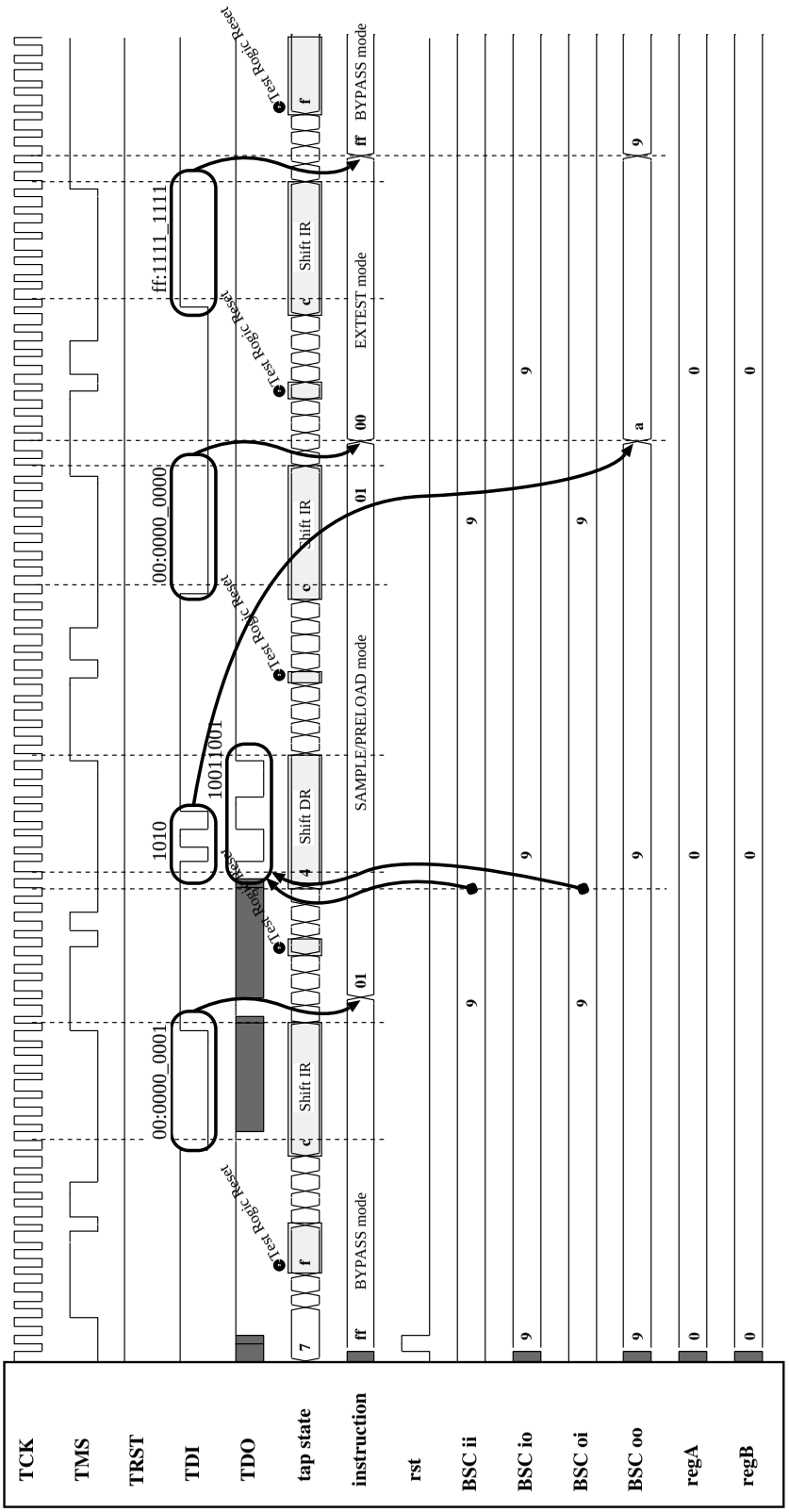


図 A.4: JTAG プロトコルの動作波形 (EXTEST、SAMPLE/PRELOAD)

A.1.2 BSC(SAMPLE/PRELOAD,EXTEST)

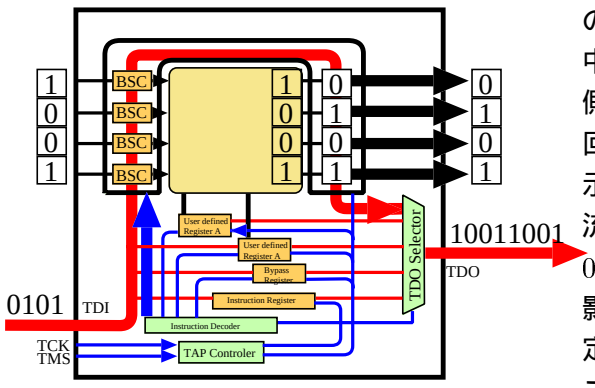


図 A.5: ユーザーレジスタの読み書き

ここではBSCのSAMPLE/PRELOAD,EXTESTの機能を説明する。波形は図 A.5 に示す。波形の中の BSCii が入力側 BSC のピン側、BSCio が入力側 BSC の内部回路側,BSCoi が出力側 BSC の内部回路側、BSCoo が出力側 BSC のピン側をそれぞれ示す。また BSC の bit の数え方はテストデータの流れに沿って入力側 BSC で 0,1,2,3 出力側 BSC で 0,1,2,3 とする。(これは 16 進表示の bit を読む時に影響する。)。この例では IC の入力は 9:1001 に固定する。また内部回路の出力も 9:1001 に固定し、これらのパターンを JTAG で読み出す手順を説明する (SAMPLE)。また EXTEST を使って IC の出力側のピンにパターン 1010 を設定する。

SAMPLE/PRELOAD 命令の設定　ユーザーレジスタの読み書きのところで説明したのと同じ要領で SAMPLE/PRELOAD 命令のインストラクションコードである 0000_0001 を書き込み、Test Rogic Reset に戻る。

パターンの読みだしとパターンの設定　この例の場合は IC の入力ピンの状態、内部回路の出力の読みだしと後で行なう EXTEST のためのパターンの設定を同時に行なっている。またこの間に内部回路が通常の動作をしていたとしても JTAG のテスト動作はそれに全く影響を与えない。まずパターンの読みだし (SAMPLE) に関して説明する。読みだしは tap state を Shift DR にして BSC の数である 8 回分の TCK で読み出す。BSC への値のラッチは Shift DR の前の状態である Capture DR が Shift DR に移る時の TCK 上がりエッジで行なわれる。この例の場合、読みだしの結果 10011001 を得ているがこの内上位 4bit が内部回路の出力に対応し、下位 4bit が IC の入力側のピンの状態に対応する。
パターンの設定 (PRELOAD) もユーザーレジスタの場合と同じように、EXTEST のときに出力ピンに出したいパターンを入れていく。この例の場合ではパターンは a:1010 でこの 4bit の入力の後さらに 4bit シフトさせることでこのパターンが出力側の BSC にまで押し出されたことになる。この段階では BSC の Update reg にパターンは入っているが BSC の出力部のセレクタが Update reg の値を外部にロードしていない状態である。これをロードするために次に EXTEST 命令を設定する。

EXTEST 命令の設定　EXTEST 命令のインストラクションコードである 0000_0000 を入力する。PRELOAD で設定したパターンはインストラクションコードをシフトさせたあとの Update IR 状態のときの TCK の下がりエッジから有効になる。
この状態に設定し外部でテストを行なう時はここで Test Rogic Reset 状態にしておけば良い。

BYPASS 命令に戻す　EXTEST が完了し BYPASS モードに戻す。これで通常の動作状態に戻せる。

A.2 JTAG のサンプルコード

ここでは前節の JTAG の動作の説明に使った JTAG のコードを示す。このコードは JTAG コードの自動生成スクリプト「jtagmaker」によって生成されたもので、4bit の RW モードのレジスタを 2 種、入出力で各々4bit の BSC を搭載するものである。jtagmaker に関する詳細は http://www.icepp.s.u-tokyo.ac.jp/ katori/jm.html で述べている。

```
//‘timescale 10ps/10ps
//this verilog code is made by jtagmaker.pl (version 0.2, last update:2dec2000)
//by Yuichi Katori
//*********************************************************************************
//******************************************************************** JTAG_MAIN
//*********************************************************************************
module jtag_main(TCK,TMS,TDI,TRST,TDO,rstin,DR4regA,DR4regB,BSCii,BSCio,BSCoi,BSCoo);
input TCK, TMS, TDI, TRST;
output TDO;
input  rstin;
output [3:0]DR4regA;
output [3:0]DR4regB;
input [3:0]BSCii;
output [3:0]BSCio;
input [3:0]BSCoi;
output [3:0]BSCoo;
wire mEXTEST,mSAMPLE,mINTEST,mRUNBIST,mID,mUSERCODE,mCLAMP,mHIGHZ,mBYPASS,mwINST,rst;
wire TDO_INST,TDO_BYPASS;
wire NCBYi,NCBYo;
wire [4:0]OPBY;
wire SEU_TOTAL;
wire [3:0]tap_state;
wire [7:0]INSTRUCTION,SEU_INST,s8INST;
wire [4:0]OPinst;
wire [1:0]mregA;
wire [1:0]mregB;
wire mrregA,mwregA;
wire mrregB,mwregB;
wire TDO_regA;
wire TDO_regB;
wire [4:0]OPBSCI;
wire [4:0]OPBSCO;
wire mBSC;
wire BSCTDIO;
wire TDO_BSC;
wire [3:0]s4BSCI;
wire [3:0]s4BSCO;
```

```
wire [4:0]OP_regA;
wire [4:0]OP_regB;
wire [3:0]s4regA,seu4regA;
wire [3:0]s4regB,seu4regB;
//********************************************* declaration of tap state
parameter Test_Logic_Reset      = 4'hf;
parameter Run_Test_Idle         = 4'h7;
parameter Select_DR_Scan        = 4'h6;
parameter Capture_DR            = 4'h5;
parameter Shift_DR              = 4'h4;
parameter Exit1_DR              = 4'h3;
parameter Pause_DR              = 4'h2;
parameter Exit2_DR              = 4'h1;
parameter Update_DR             = 4'h0;
parameter Select_IR_Scan        = 4'he;
parameter Capture_IR            = 4'hd;
parameter Shift_IR              = 4'hc;
parameter Exit1_IR              = 4'hb;
parameter Pause_IR              = 4'ha;
parameter Exit2_IR              = 4'h9;
parameter Update_IR             = 4'h8;
//*********************************************
parameter def4regA  = 4'b0000;
parameter def4regB  = 4'b0000;
//*********************************************

parameter wHI = 1'b1;
parameter wLOW = 1'b0;
//*********************************************
assign rst = ~rstin;
tap_controler tap(TCK, TMS, TRST, tap_state);
assign SEU_TOTAL = |SEU_INST|seu4regA|seu4regB;
//************************ function for operation of data register (write & read mode)
//where OPRW[] are operation wire for (mux (before shift reg & for ouput)) &
//enable condition for shift reg & tdo reg & update reg
function [4:0] OPRW;
input [3:0]tap_state;
input [1:0]mORW;
begin
OPRW[0] = (tap_state == Shift_DR); //mux before TDI reg
OPRW[1] = ((tap_state == Shift_DR)|(tap_state == Capture_DR))
&& (mORW[1] | mORW[0]);  //enable TDI reg
OPRW[2] = (tap_state == Shift_DR); //enable TDO reg
OPRW[3] = (tap_state == Update_DR) && mORW[0]; //enable UPDATE reg
```

```
OPRW[4] = 1'b1; //mux of output
end
endfunction
//************************ write mode only
function [4:0] OPW;
input [3:0]tap_state;
input [1:0]mORW;
begin
OPW[0] = 1'b1; //mux before TDI reg
OPW[1] = (tap_state == Shift_DR) && (mORW[1] | mORW[0]);  //enable TDI reg
OPW[2] = (tap_state == Shift_DR); //enable TDO reg
OPW[3] = (tap_state == Update_DR) && mORW[0]; //enable UPDATE reg
OPW[4] = 1'b1; //mux of output
end
endfunction
//************************ read mode
function [4:0] OPR;
input [3:0]tap_state;
input [1:0]mORW;
begin
OPR[0] = (tap_state == Shift_DR) && mORW[1];
OPR[1] = ((tap_state == Shift_DR)|(tap_state == Capture_DR)) && (mORW[1] | mORW[0]);
OPR[2] = (tap_state == Shift_DR) && (mORW[1] | mORW[0]);
OPR[3] = 1'b0;
OPR[4] = 1'b1;
end
endfunction
//******************************************************************** bypass reg
assign OPBY[0] = 1'b1;
assign OPBY[1] = mBYPASS;
assign OPBY[2] = mBYPASS;
assign OPBY[3] = 1'b0;
assign OPBY[4] = 1'b0;
DR DR1bypass(TCK,TDI,TDO_BYPASS,NCBYi,NCBYo,OPBY);
//****************************************************************** regA (4bit)
assign OP_regA = OPRW(tap_state,mregA);
DRV DR4regA0(TCK,TDI      ,s4regA[0],DR4regA[0],DR4regA[0],OP_regA,
seu4regA[0],def4regA[0],rst);
DRV DR4regA1(TCK,s4regA[0],s4regA[1],DR4regA[1],DR4regA[1],OP_regA,
seu4regA[1],def4regA[1],rst);
DRV DR4regA2(TCK,s4regA[1],s4regA[2],DR4regA[2],DR4regA[2],OP_regA,
seu4regA[2],def4regA[2],rst);
DRV DR4regA3(TCK,s4regA[2],TDO_regA ,DR4regA[3],DR4regA[3],OP_regA,
seu4regA[3],def4regA[3],rst);
```

```
//************************************************************** regB (4bit)
assign OP_regB = OPRW(tap_state,mregB);
DRV DR4regB0(TCK,TDI      ,s4regB[0],DR4regB[0],DR4regB[0],OP_regB,
seu4regB[0],def4regB[0],rst);
DRV DR4regB1(TCK,s4regB[0],s4regB[1],DR4regB[1],DR4regB[1],OP_regB,
seu4regB[1],def4regB[1],rst);
DRV DR4regB2(TCK,s4regB[1],s4regB[2],DR4regB[2],DR4regB[2],OP_regB,
seu4regB[2],def4regB[2],rst);
DRV DR4regB3(TCK,s4regB[2],TDO_regB ,DR4regB[3],DR4regB[3],OP_regB,
seu4regB[3],def4regB[3],rst);
assign mBSC = mSAMPLE | mEXTEST | mINTEST;
//********************************************************** BSC input side (4bit)

assign OPBSCI[0] = mBSC && (tap_state == Shift_DR);
assign OPBSCI[1] = mBSC && ((tap_state == Shift_DR)|(tap_state == Capture_DR));
assign OPBSCI[2] = mBSC && (tap_state == Shift_DR);
assign OPBSCI[3] = mBSC && (tap_state == Update_DR);
assign OPBSCI[4] = mINTEST;
DR BSCI0(TCK,TDI       ,s4BSCI[0],BSCii[0],BSCio[0],OPBSCI);
DR BSCI1(TCK,s4BSCI[0],s4BSCI[1],BSCii[1],BSCio[1],OPBSCI);
DR BSCI2(TCK,s4BSCI[1],s4BSCI[2],BSCii[2],BSCio[2],OPBSCI);
DR BSCI3(TCK,s4BSCI[2],BSCTDIO   ,BSCii[3],BSCio[3],OPBSCI);
//********************************************************** BSC output side (4bit)

assign OPBSCO[0] = mBSC && (tap_state == Shift_DR);
assign OPBSCO[1] = mBSC && ((tap_state == Shift_DR)|(tap_state == Capture_DR));
assign OPBSCO[2] = mBSC && (tap_state == Shift_DR);
assign OPBSCO[3] = mBSC && (tap_state == Update_DR);
assign OPBSCO[4] = mEXTEST;
DR BSCO0(TCK,BSCTDIO  ,s4BSCO[0],BSCoi[0],BSCoo[0],OPBSCO);
DR BSCO1(TCK,s4BSCO[0],s4BSCO[1],BSCoi[1],BSCoo[1],OPBSCO);
DR BSCO2(TCK,s4BSCO[1],s4BSCO[2],BSCoi[2],BSCoo[2],OPBSCO);
DR BSCO3(TCK,s4BSCO[2],TDO_BSC  ,BSCoi[3],BSCoo[3],OPBSCO);
//************************************************************** TDO selector
assign TDO = (mwINST & TDO_INST)|(mBYPASS & TDO_BYPASS)
|(mEXTEST & TDO_BSC)|(mSAMPLE & TDO_BSC)|(mINTEST & TDO_BSC)
|(|mregA & TDO_regA)
|(|mregB & TDO_regB)
;
//************************************************************** INST decoder
assign mEXTEST = (INSTRUCTION == 8'b0000_0000) ? wHI : wLOW;
assign mSAMPLE = (INSTRUCTION == 8'b0000_0001) ? wHI : wLOW;
assign mINTEST = (INSTRUCTION == 8'b0000_0010) ? wHI : wLOW;
assign mRUNBIST = (INSTRUCTION == 8'b0000_0011) ? wHI : wLOW;
```

```
assign mID = (INSTRUCTION == 8'b0000_0111) ? wHI : wLOW;
assign mUSERCODE = (INSTRUCTION == 8'b0000_0101) ? wHI : wLOW;
assign mCLAMP = (INSTRUCTION == 8'b0000_0110) ? wHI : wLOW;
assign mHIGHZ = (INSTRUCTION == 8'b0000_0111) ? wHI : wLOW;
assign mBYPASS = (INSTRUCTION == 8'b1111_1111) ? wHI : wLOW;
assign mwINST = ((Capture_IR >= tap_state)
& (tap_state >= Update_IR)) ? wHI : wLOW;
assign mrregA = (INSTRUCTION == 8'b0000_1000) ? wHI : wLOW;
assign mwregA = (INSTRUCTION == 8'b0000_1001) ? wHI : wLOW;
assign mregA = {mrregA,mwregA};
assign mrregB = (INSTRUCTION == 8'b0001_0000) ? wHI : wLOW;
assign mwregB = (INSTRUCTION == 8'b0001_0001) ? wHI : wLOW;
assign mregB = {mrregB,mwregB};
//************************************************************** INST register
assign OPinst[0] = 1'b1;
assign OPinst[1] = mwINST && (tap_state == Shift_IR);
assign OPinst[2] = mwINST;
assign OPinst[3] = mwINST && (tap_state == Update_IR);
assign OPinst[4] = 1'b1;

DRV DR8inst0(TCK,TDI,s8INST[0],INSTRUCTION[0],INSTRUCTION[0],OPinst,
SEU_INST[0],wHI,rst);
DRV DR8inst1(TCK,s8INST[0],s8INST[1],INSTRUCTION[1],INSTRUCTION[1],OPinst,
SEU_INST[1],wHI,rst);
DRV DR8inst2(TCK,s8INST[1],s8INST[2],INSTRUCTION[2],INSTRUCTION[2],OPinst,
SEU_INST[2],wHI,rst);
DRV DR8inst3(TCK,s8INST[2],s8INST[3],INSTRUCTION[3],INSTRUCTION[3],OPinst,
SEU_INST[3],wHI,rst);
DRV DR8inst4(TCK,s8INST[3],s8INST[4],INSTRUCTION[4],INSTRUCTION[4],OPinst,
SEU_INST[4],wHI,rst);
DRV DR8inst5(TCK,s8INST[4],s8INST[5],INSTRUCTION[5],INSTRUCTION[5],OPinst,
SEU_INST[5],wHI,rst);
DRV DR8inst6(TCK,s8INST[5],s8INST[6],INSTRUCTION[6],INSTRUCTION[6],OPinst,
SEU_INST[6],wHI,rst);
DRV DR8inst7(TCK,s8INST[6],TDO_INST,INSTRUCTION[7],INSTRUCTION[7],OPinst,
SEU_INST[7],wHI,rst);
endmodule
//*****************************************************************************
//************************************************************** Tap controler
//*****************************************************************************
module tap_controler(TCK, TMS, TRST, tap_state);

input TCK, TMS, TRST;
output [3:0] tap_state;
```

```
reg [3:0] tap_state;

// definition of tap state
parameter Test_Logic_Reset      = 4'hf;
parameter Run_Test_Idle         = 4'h7;
parameter Select_DR_Scan        = 4'h6;
parameter Capture_DR            = 4'h5;
parameter Shift_DR              = 4'h4;
parameter Exit1_DR              = 4'h3;
parameter Pause_DR              = 4'h2;
parameter Exit2_DR              = 4'h1;
parameter Update_DR             = 4'h0;
parameter Select_IR_Scan        = 4'he;
parameter Capture_IR            = 4'hd;
parameter Shift_IR              = 4'hc;
parameter Exit1_IR              = 4'hb;
parameter Pause_IR              = 4'ha;
parameter Exit2_IR              = 4'h9;
parameter Update_IR             = 4'h8;

// TAP state machine
always @(posedge TCK or negedge TRST)
   begin
if(!TRST) tap_state <= Test_Logic_Reset;
else
begin
case (tap_state)
Test_Logic_Reset:
            tap_state <= TMS ? Test_Logic_Reset : Run_Test_Idle;
        Run_Test_Idle:
            tap_state <= TMS ? Select_DR_Scan : Run_Test_Idle;
        Select_DR_Scan:
            tap_state <= TMS ? Select_IR_Scan : Capture_DR;
        Capture_DR:
            tap_state <= TMS ? Exit1_DR : Shift_DR;
        Shift_DR:
            tap_state <= TMS ? Exit1_DR : Shift_DR;
        Exit1_DR:
            tap_state <= TMS ? Update_DR : Pause_DR;
        Pause_DR:
            tap_state <= TMS ? Exit2_DR : Pause_DR;
        Exit2_DR:
            tap_state <= TMS ? Update_DR : Shift_DR;
        Update_DR:
```

```
            tap_state <= TMS ? Select_DR_Scan : Run_Test_Idle;
        Select_IR_Scan:
            tap_state <= TMS ? Test_Logic_Reset : Capture_IR;
        Capture_IR:
            tap_state <= TMS ? Exit1_IR : Shift_IR;
        Shift_IR:
            tap_state <= TMS ? Exit1_IR : Shift_IR;
        Exit1_IR:
            tap_state <= TMS ? Update_IR : Pause_IR;
        Pause_IR:
            tap_state <= TMS ? Exit2_IR : Pause_IR;
        Exit2_IR:
            tap_state <= TMS ? Update_IR : Shift_IR;
        Update_IR:
            tap_state <= TMS ? Select_DR_Scan : Run_Test_Idle;
        endcase
end
    end

endmodule
//////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////// JTAG registers
//////////////////////////////////////////////////////////////////////////////////
module DRV(TCK,TDI,TDO,NDI,NDO,OP5,SEU_FLAG,def,rst);
input  TCK;
input TDI,NDI;
output TDO,NDO;
reg  TDIreg,TDOreg;
input [4:0]OP5;
input  def,rst;
output  SEU_FLAG;
wire TDIregin,UPregout;
wire xr,xs;

always @(posedge TCK)
begin
if(OP5[1] == 1'b1)
        TDIreg <= TDIregin;
end

always @(negedge TCK)
begin
if(OP5[2] == 1'b1)
        TDOreg <= TDIreg;
```

```
end
assign TDO = TDOreg;

assign xs = ~(rst && def);
assign xr = (~rst)|def;
dumvdffr UPreg(rst|(OP5[3] && (~TCK)),TDIreg,UPregout,SEU_FLAG,xr,xs);

assign TDIregin = OP5[0] ? TDI : NDI;
assign NDO = OP5[4] ? UPregout : NDI;

endmodule
//****************************** DR data register

module DR(TCK,TDI,TDO,NDI,NDO,OP5);
input TCK;
input TDI,NDI;
output TDO,NDO;
reg  TDIreg,TDOreg,UPreg;
input [4:0]OP5;
wire TDIregin;

always @(posedge TCK)
begin
if(OP5[1] == 1'b1)
        TDIreg <= TDIregin;
end

always @(negedge TCK)
begin
if(OP5[2] == 1'b1)
        TDOreg <= TDIreg;
if(OP5[3] ==1'b1)
UPreg <= TDIreg;
end

assign TDO = TDOreg;
assign TDIregin = OP5[0] ? TDI : NDI;
assign NDO = OP5[4] ? UPreg : NDI;
endmodule
```

付 録B　Patch Panel ASICに関する補足

ここではPatch Panel ASICに関して補足的な説明、またシミュレーションの結果を述べる。

B.1　Patch Panel ASICにおけるユーザー定義レジスタの機能

各レジスタの機能は以下のようになっている。

- MASK(16bit RW)
  1チャンネル毎に信号をマスクすることが出来る。またマスクはBCID回路の一部として組み込まれている。0がマスク無し、1がマスクあり。初期値は全て0でマスクがかかっていない状態になる。

- TESTPULSE_FINE (5bit RW)
  テストパルス回路のトリガー信号の遅延回路のうち、DLL回路によるサブナノ秒精度の遅延の大きさを決める。00000が遅延無し、11111が最大遅延の25nsec。初期値は00000で遅延無し。

- TESTPULSE_COARSE (2bit RW)
  テストパルス回路のトリガー信号の遅延回路のうち、クロック周期単位の遅延の大きさを決める。00が遅延無し、11が最大遅延の2LHCCLK。初期値は00で遅延無し。

- TESTPULSE_AMP (2bit RW)
  テストパルスの振幅を設定する。初期値は00で振幅最大。

- TESTPULSE_VETO (1bit RW)
  テストパルスにvetoをかける。

- BCID_DEL (5bit RW)
  BCIDに入るクロックのタイミングを調節する。DLL回路によってサブナノ秒精度の遅延を設定できる。初期値は00000で遅延無し。

- BCID_GATE (5bit RW)
  BCIDのゲート幅設定用のクロックのタイミングを調節する。DLL回路によってサブナノ秒精度の遅延を設定できる。初期値は00000で遅延無し。

- SIGNAL_DEL (5bit RW)
  16チャンネルの信号に共通でかける遅延を設定する。DLL回路によってサブナノ秒精度の遅延を設定できる。初期値は00000で遅延無し。

- SEU (1bit R)
  チップ全体からのSEUのフラッグのORをとり読み出すためのレジスタで、読みだし専用のレジスタを使う。

- FUSE (2bit R)
  ASD に供給する電源のヒューズの断線を検出し読み出すためのレジスタで、読みだし専用のレジスタを使う。

B.2 Patch Panel ASIC における JTAG の動作の検証

主に JTAG プロトコルからなるデジタル部分に関しては、verlogHDL の記述に基づいて動作シミュレーションを行ない、以下に示すようにその動作を確認した。

- TAP コントローラの動作確認
  TAP コントローラに関して、TMS,TCK からパターンを入力して全ての状態と遷移経路を通り、その動作が正しいことを確認した。(図 B.1)

- バイパス命令のテスト
  TMS,TDI,TCK によりパターンを入力しインストラクションレジスタに値 “00:0000_0000” を書き込み、バイパスモードにして TDI からの信号が TDO へバイパスされることを確認した。

- インストラクションレジスタの動作確認
  TMS,TDI,TCK によりパターンを入力しインストラクションレジスタに値 “51:0101_0001” を書き込み、インストラクションに値を設定し、さらにこれを読みだした。(図 B.2) さらに幾つかのパターンに関して同様のことを行ないインストラクションレジスタが正しく機能することを確認した。

- EXTEST 命令の動作確認
  BSC の入力側にパターン (f0f0)、出力側にパターン (3333) を設定しておきインストラクションレジスタに”11:1111_1111” を書き込み、EXTEST 命令にして、TDO よりこれらの値を読み出されることを確認した。

- R モードのデータレジスタの動作確認
  R モードのレジスタ、SEU、FUSE に対してそれぞれインストラクションの設定を行ない、設定値の読みだしが正確に行なわれることを確認した。図 B.4 にレジスタ FUSE の動作確認の波形を示す。

- RW モードのデータレジスタの動作確認
  RW モードのレジスタ、MASK、PULSE_FINE_TIM,PULSE_COARSE_TIM, PULSE_AMP, BCID_DEL,BCID_GATE,SIGNAL_DEL, CLK_DEL,TPG_VETO に対して、それぞれインストラクションの設定とデータパターンの読み書きを確認した。図 B.5 にレジスタ MASK に関する動作の確認を行なった波形を示す。

- リセット機能の動作確認
  今回の JTAG プロトコルから初めて搭載した初期値の設定に関して、リセット (RST) を入力して各 RW モードのレジスタが正しく動作することを確認した。

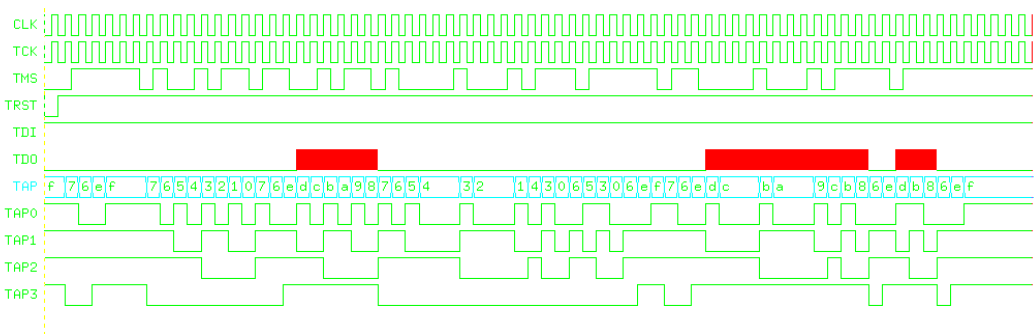


図 B.1: JTAG のテスト波形 tap state の遷移
TAP コントローラに関して、TMS,TCK からパターンを入力して Test-Logic-Reset, Run-Test/Idle,Slect-DR(IR)-Scan,Capture-DR(IR),Shift-DR(IR),Exit-DR(IR), Pause-DR(IR),Exit2-DR(IR),Update-DR(IR) と TAP コントローラの全ての状態とその間の全ての遷移経路を通り、正しく動作していることを確認した。

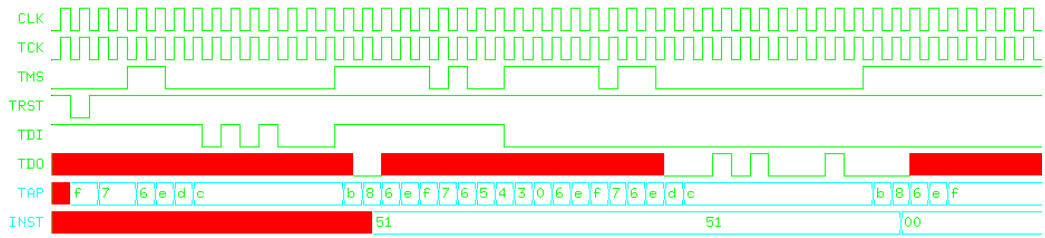


図 B.2: JTAG のテスト波形 インストラクション
TAP コントローラに関して、TMS,TCK からパターンを入力してインストラクションレジスタに値 “51:0101_0001” を書き込み、さらにこれを読みだした。

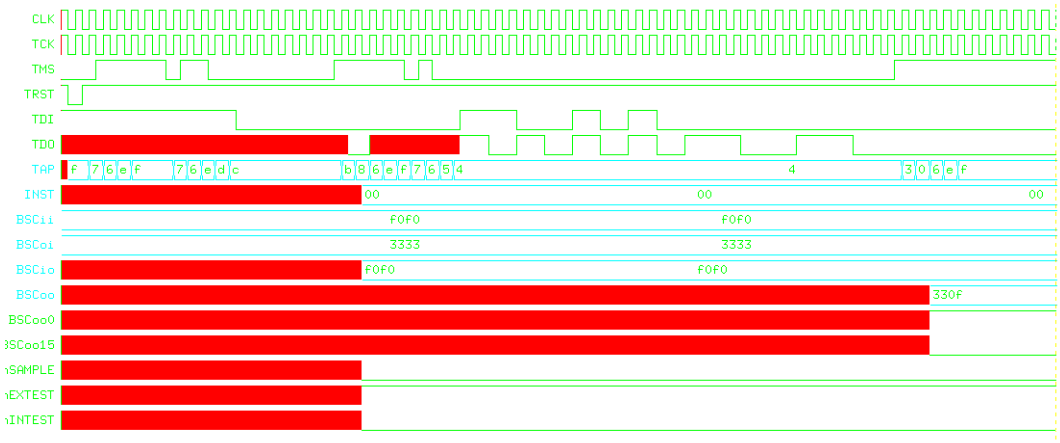


図 B.3: JTAG のテスト波形,EXTEST
BSC の入力側にパターン (f0f0)、出力側にパターン (3333) を設定しておきインストラクションレジスタに”11:1111_1111” を書き込み、EXTEST 命令にして、TDO よりこれらの値を読み出されることを確認した。

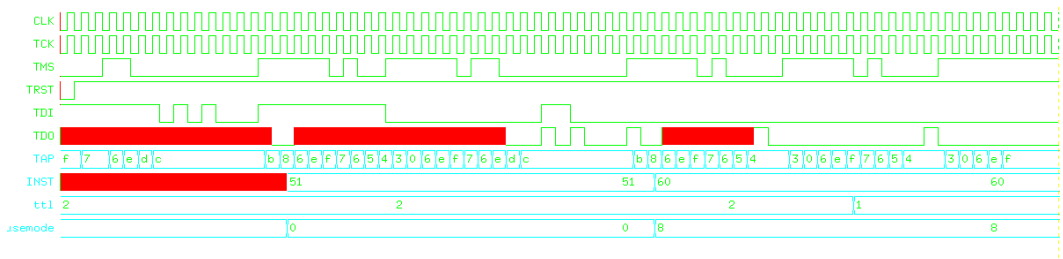


図 B.4: JTAG のテスト波形, データレジスタ FUSE
TDI、TMS、TCK からパターンを入力しインストラクションに”60:0110_0000” を書き込みレジスタ FUSE を読みだしモードにして、設定されていた値”01” を読みだした。

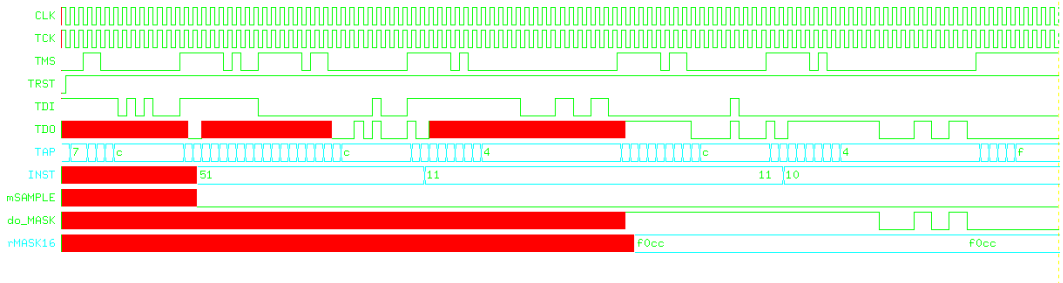


図 B.5: JTAG のテスト波形,RW モードデータレジスタ
TDI、TMS、TCK からパターンを入力しインストラクションに”11:0001_0001” を書き込み RW モードであるレジスタ MASK を読みだしモードにして、パターン”f0cc:1111_0000_1100_1100” を書き込み、さらにインストラクションに”10:0001_0000” を書き込みレジスタ MASK を読みだしモードにしてこのパターンを読みだした。

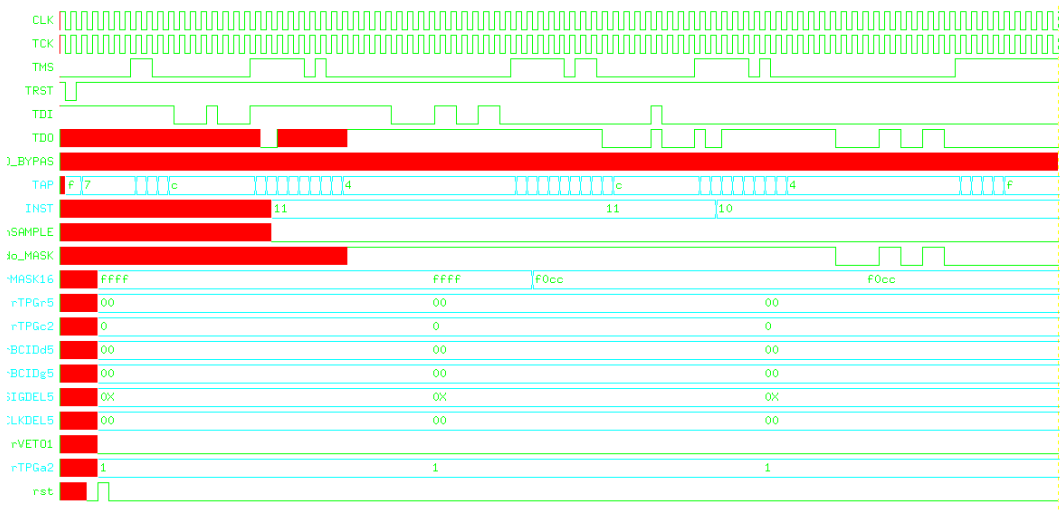


図 B.6: JTAG のテスト波形 リセット
RW モードのレジスタに搭載されている初期値の設定の機能を確認する。リセット信号 (RST) と同時に全てのデータレジスタが初期化される。

付 録C　Slave Board ASIC,High-pT ASIC に関する補足

ここでは Slave Board ASIC,High-pT ASIC に関して補足的な説明、またシミュレーションの結果を述べる。

C.1　Slave Board ASIC におけるユーザー定義レジスタの機能

Slave Board ASIC のレジスタの機能は、以下のようになっている。

- MASK1 (160bit RW)
  読みだし部とマトリックス部に共通でかかるマスクの有無を設定する。初期値は全て 0 でマスク無し。

- MASK1P (160bit RW)
  読みだし部とマトリックス部に共通でかかるマスクのパターンを設定する。初期値は全て 0 でマスクとして 0 を設定している。

- MASK2 (160bit RW)
  マトリックス部にかかるマスクの有無を設定する。初期値は全て 0 でマスク無し。

- MASK2P (160bit RW)
  マトリックス部にかかるマスクのパターンを設定する。初期値は全て 0 でマスクとして 0 を設定する。

- TPP(Test Pulse Pattern) (160bit RW)
  テストパルスのパターン (1 or 0) を設定する。初期値は全て 0 でパターンとして 0 を設定している。

- DELAY (4bit RW)
  入力部の信号の遅延を設定する。4bit のうち 2bit が A 列 B 列にかかる遅延、残る 2bit が C 列 D 列にかかる遅延である。初期値は全て 0 で遅延は無し (CLK の上がりエッジのラッチはある。)

- TRGMODE(1bit RW)
  トリガーモード (マトリックスのコインシデンス条件) を設定する。設定値 0 が通常のコインシデンス条件で 1 が厳しいコインシデンス条件になる。初期値は 0。

- TESTPULSE(5bit RW)
  テストパルスに関する設定を行なう。4bit が 0∼15CLK で 15 段階の遅延回路の設定に使われる。残る 1bit は veto。初期値は全て 0 で遅延無し、veto 無し。

- L1AVETO(1bit RW)
  LEVEL1ACCEPT 信号に veto をかける。初期値は 0 で veto 無し。

- CLKINV(1bit RW)
  TTC 信号受信部に付けられる CLK を反転させる回路を設定する。初期値は 0 で反転は無し。

- RESET(1bit RW)
  デランダマイザーに入るオプションのリセットで、初期値は 0 でレジスタの値を 1 に設定した直後に 1TCK 幅のパルスを出力しリセットに使われる。1 度パルスを出して次に出すには、一端レジスタの値を 0 にしてからもう一度 1 を書き込む。

- SEU(1bit R)
  SEU のフラッグを読み出す。読みだし専用のレジスタを使う。

- MODULE(8bit R)
  基板上で設定されるモジュールのタイプ (3bit) とアドレス (5bit) を読み出す。読みだし専用のレジスタを使う。

- OVERFLOW(8bit R)
  デランダマイザーのオーバーフローのカウンターの値を読み出す。読みだし専用のレジスタを使う。

- ID(32bit R)
  チップの ID を読み出す。なにを設定するかは未定。読みだし専用のレジスタを使う。

C.2 Slave Board ASIC に関する JTAG の動作の検証

主に JTAG プロトコルからなるデジタル部分に関しては、verlogHDL の記述に基づいて動作シミュレーションを行ない、以下に示すようにその動作を確認した。

- JTAG プロトコルの EXTEST 命令の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに"00:0000_0000" を書き込み、EXTEST モードにし、パターン"0ffffffff" を BSC の出力側にロードした。またこの他にも幾つかのパターンに関してもテストを行なって正しく動作することを確認した。これにより入力部、制御部のバウンダリースキャンセルも含めた EXTEST の動作確認が行なわれたことになる。(図 C.1)

- JTAG プロトコルの INTEST 命令の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに"02:0000_0010" を書き込み、INTEST モードにし、IC の出力信号のうち BSC のついたものを TDO から読みだした。これにより入力部、制御部のバウンダリースキャンセルも含めた INTEST の動作確認が行なわれたことになる。(図 C.2)

- JTAG プロトコルの bypass 命令の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに"00:0000_0000" を書き込み、バイパスモードにして、TDI からの信号が TDO へバイパスされることを確認した。(図 C.3)

- JTAG プロトコルのデータレジスタ CLKINV と TTC 受信回路の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力しインストラクションレジスタに"51:0101_0001" を書き込み、CLKINV を書き込みモードにして値"1" を書き込んだ。(図 C.4) またこれにより CCT 信号受信回路の CLK 反転回路の動作が確認できた。

- JTAG プロトコルのデータレジスタ DELAY と位相補正回路の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力しインストラクションレジスタに"19:0001_1001" を書き込み、DELAY を書き込みモードにして幾つかのパターンを書き込み、このパラメータが正しく設定されることを確認した。(図 C.5) また同時に入力部の遅延回路が正しく動作することを確認した。

- JTAG プロトコルのデータレジスタ L1AVETO と L1Aveto 回路の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力しインストラクションレジスタに"51:0101_0001" を書き込み、L1AVETO を書き込みモードにして幾つかのパターンを書き込んだ。

  (図 C.6) また同時に L1A 信号に VETO がかかることを確認し、この VETO をかける回路の動作も確認した。

- JTAG プロトコルのデータレジスタ MASK1,MASK1P とマスク回路の動作確認
  データレジスタ MASK1,MASK1P にそれぞれに対して、インストラクションの設定、幾つかのパターンに関してデータの書き込みを行ない、マトリックス、読みだし部に共通でかかるマスク回路の動作を確認した。(図 C.7)

- JTAG プロトコルのデータレジスタ MASK2,MASK2P とマスク回路の動作確認
  データレジスタ MASK2,MASK2P にそれぞれに対して、インストラクションの設定、幾つ

かのパターンに関してデータの書き込みを行ない、マスク回路が正しく動作することを確認した。

- JTAG プロトコルのデータレジスタ OVERFLOW の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに”78:0111_1001” を書き込み、レジスタ OVERFLOW を読みだしモードにしてレジスタに設定していたパターンを正確に読み出した。さらに幾つかのパターンに対してテストを行ない正確に動作することを確認した。(図 C.8)

- JTAG プロトコルのデータレジスタ SEU の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに”38:0011_1000” を書き込み、レジスタ SEU を読みだしモードにしてレジスタに設定していたパターンを正確に読み出した。

- JTAG プロトコルのデータレジスタ MODULE の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力しインストラクションレジスタに”40:0100_0000” を書き込み、レジスタ MODULW を読みだしモードにしてレジスタに設定していたパターンを正確に読み出した。さらに幾つかのパターンに対してテストを行ない正確に動作することを確認した。

- JTAG プロトコルのデータレジスタ TESTPULSE とテストパルス用遅延回路の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに”39:0011_1001” を書き込み、レジスタ TESTPULSE を書き込みモードにして、パターンの書き込みを行ない、テストパルス回路の遅延及び VETO 回路が正しく動作することを確認した。

- JTAG プロトコルのデータレジスタ TRGMODE の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに”41:0100_0001” を書き込み、レジスタ TRGMODE を書き込みモードにして幾つかのパターンに対して書き込み動作のテストを行ない、正確に動作することを確認した。

- JTAG プロトコルのデータレジスタ DEPTH の動作確認
  TDI,TMS,TCK から JTAG を制御するパターンを入力し、インストラクションレジスタに”31:0011_0001” を書き込み、レジスタ DEPTH を書き込みモードにして幾つかのパターンに対して書き込み動作のテストを行ない、正確に動作することを確認した。

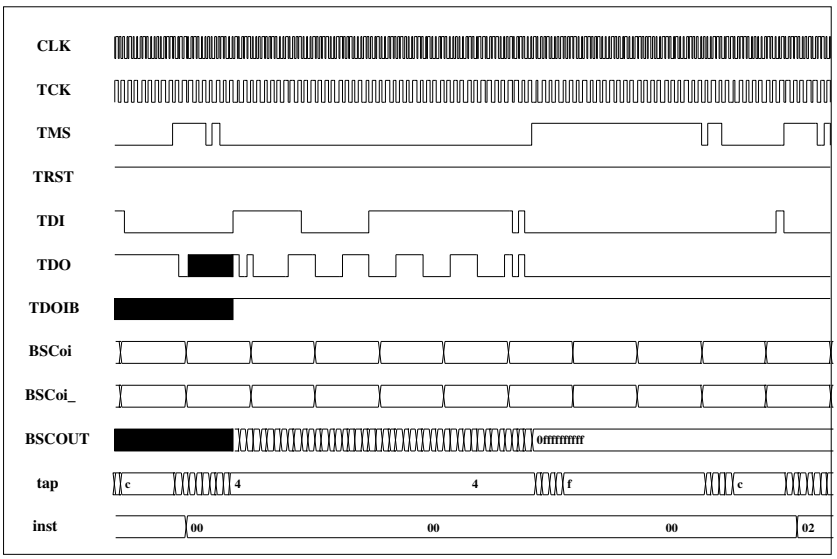


図 C.1: JTAG プロトコルのテスト波形,EXTEST 命令による
インストラクションに”00:0000_0000” を書き込み、EXTEST モードにし、パターン”0ffffffff” を BSC の出力側にロードした。

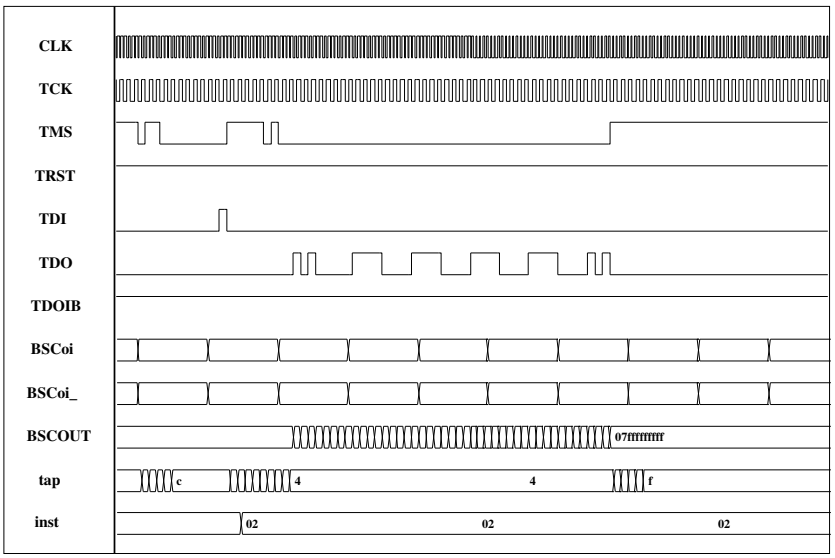


図 C.2: JTAG プロトコルのテスト波形、INTEST
インストラクションに”02:0000_0010” を書き込み INTEST モードにし、IC の出力信号のうち BSC のついたものを TDO から読みだした。

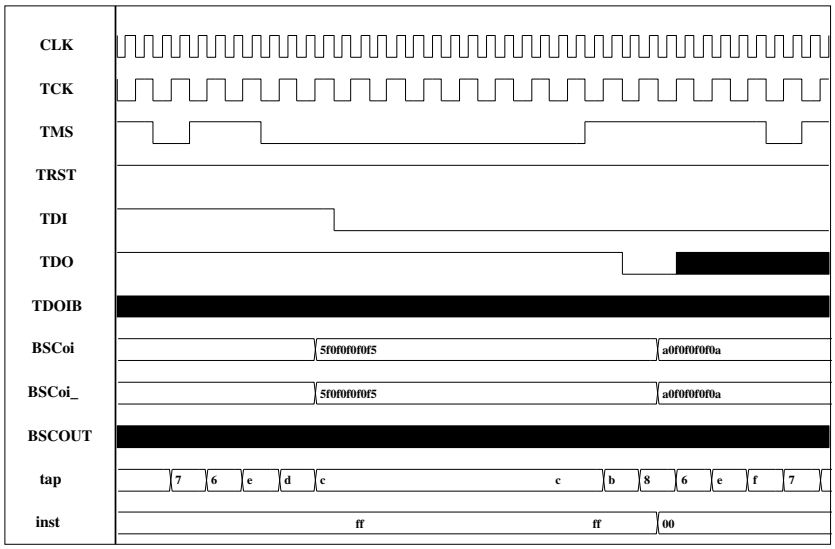


図 C.3: JTAG プロトコルのテスト波形、データレジスタ bypass
インストラクションに”ff:1111_1111” を書き込み bypass モードにし、TDI からの信号が TDO にバイパスされることを確認した。

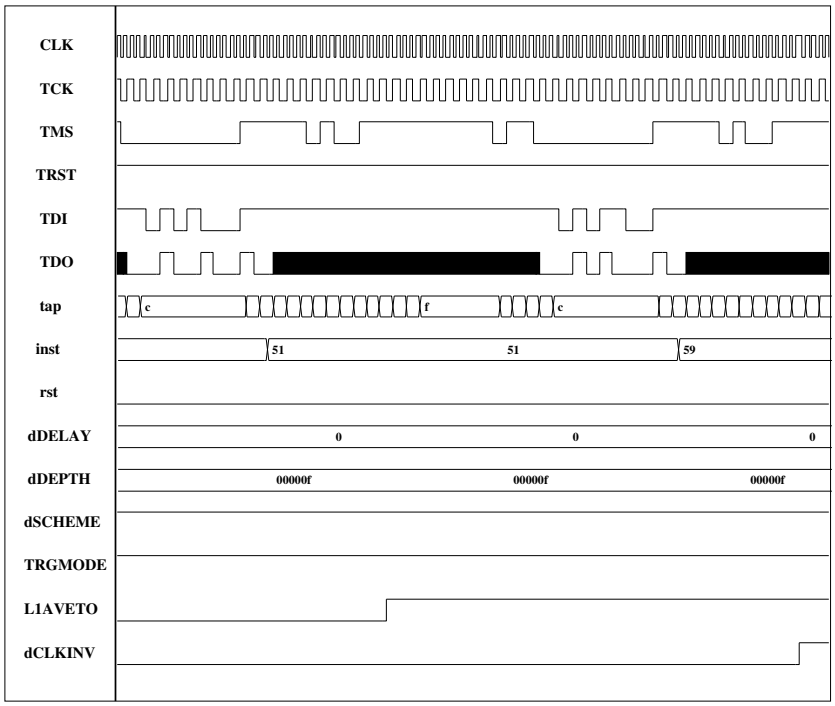


図 C.4: JTAG プロトコルのテスト波形、データレジスタ clkinv
インストラクションに”51:0101_0001” を書き込み、レジスタ clkinv に 1 を書き込んだ。
インストラクションに”59:0101_1001” を書き込み、TDO からレジスタ clkinv の値を読みだした。

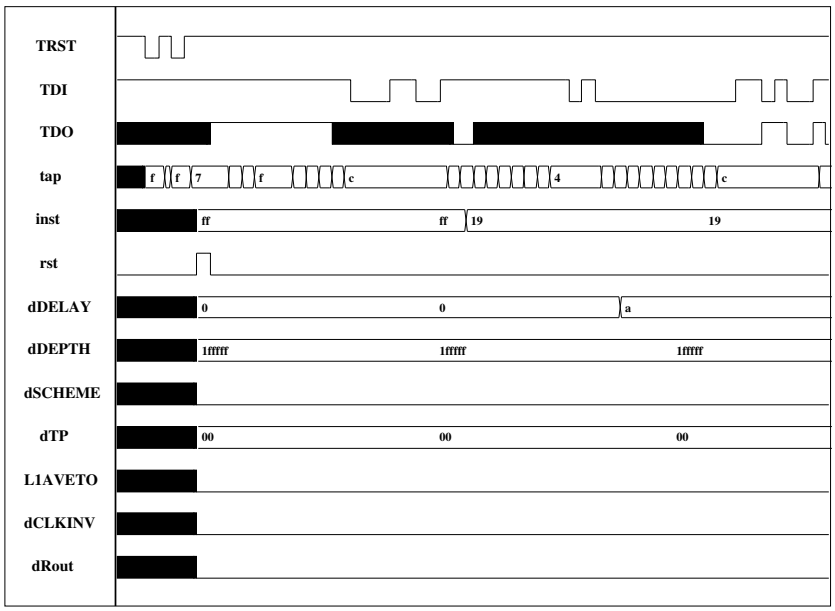


図 C.5: JTAG プロトコルのテスト波形、データレジスタ delay
インストラクションに”19:0001_1001” を書き込み、レジスタ delay にパターン”1010” を書き込んだ。さらにインストラクションに”18:0101_1000” を書き込み、TDO からレジスタ delay の値を読みだした。

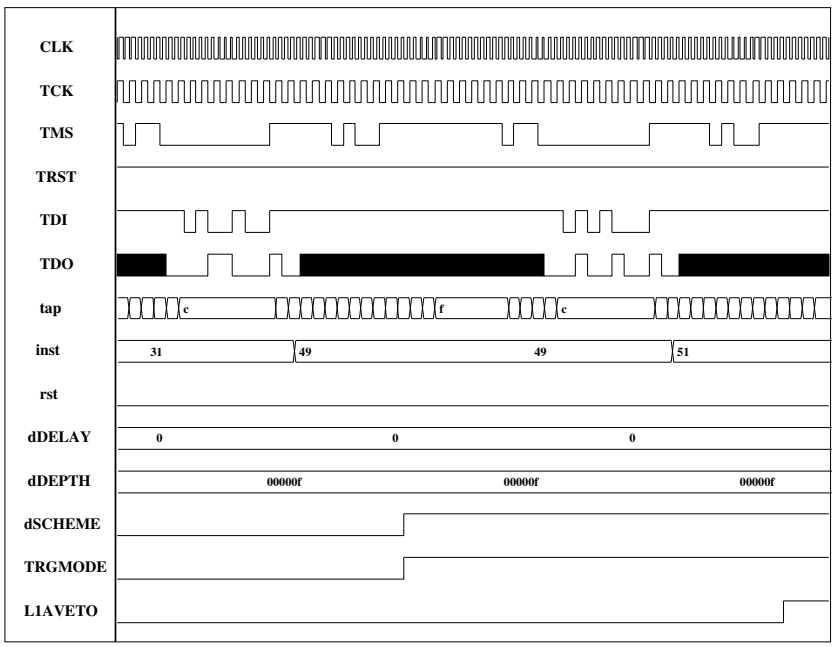


図 C.6: JTAG プロトコルのテスト波形、データレジスタ L1Aveto
インストラクションに”51:0101_0001” を書き込み、レジスタ L1Aveto にパターン”1” を書き込んだ。さらにインストラクションに”50:0101_0000” を書き込み、TDO からレジスタ L1Aveto の値を読みだした。

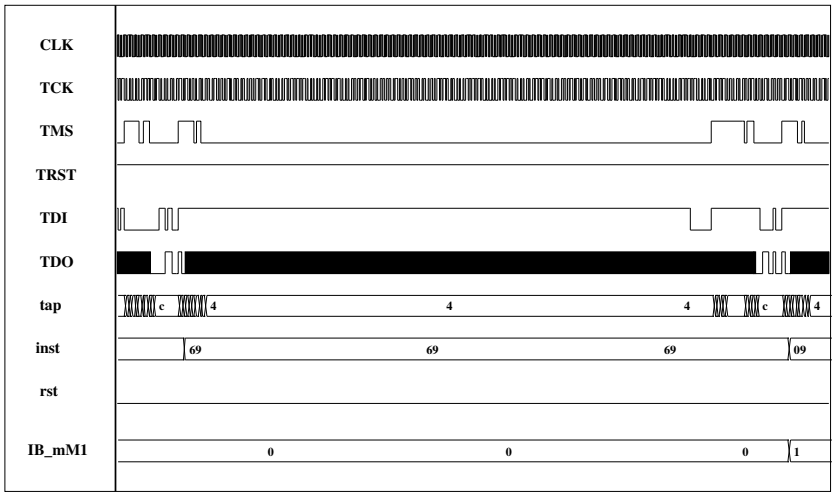


図 C.7: JTAG プロトコルのテスト波形、データレジスタ MASK1P
インストラクションに”69:0110_1001” を書き込み、レジスタ MASK1P にパターンを書き込み、マトリックス部分へのマスク値を変更した。

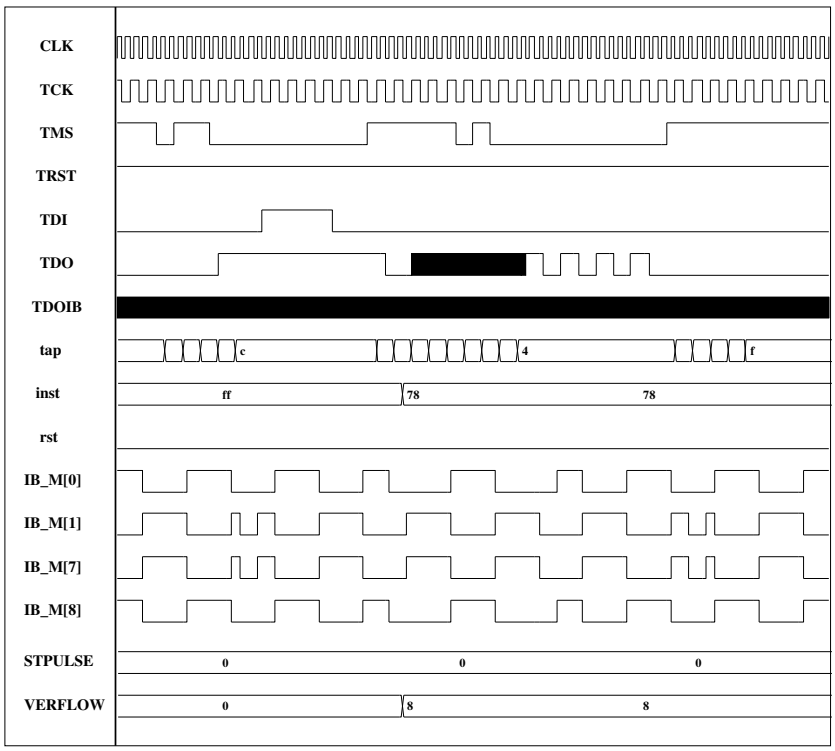


図 C.8: JTAG プロトコルのテスト波形、データレジスタ OVERFLOW
インストラクションに”78:0111_1000” を書き込み、レジスタ OVERFLOW を読みだしモードにして設定されていた値”10101010” を読みだした。

C.3 High-pT ASIC におけるユーザー定義レジスタの機能

High-pT ASIC のユーザー定義レジスタの機能は、以下のようになっている。

- DELAY 0∼3 (3bit RW)
  4 つの triplet 用 Slave Board ASIC からの信号の遅延を調節する。4 つそれぞれに設定できる。RW モードのレジスタを使う 。遅延は最初のクロックのエッジから 1/2CLK 毎に 8 段階に設定可能である。

- DELAY 4∼5 (3bit RW)
  3 つの doublet 用 Slave Board ASIC からの信号の遅延を調節する。3 つそれぞれに設定できる。RW モードのレジスタを使う 。遅延は最初のクロックのエッジから 1/2CLK 毎に 8 段階に設定可能である。

- GLINK(3bit RW)
  Hewlet-Packard 社の G-LINK ドライバーの制御用のワイヤ G-RESET,G-CAV, G-DAV を設定する。

- SEU (1bit R)
  SEU のフラグを読み出す。読みだし専用のレジスタを使う。

C.4 High-pT ASIC における JTAG の動作の検証

High-pT ASIC の JTAG 部分に関して verlogHDL の記述に基づいて動作シミュレーションを行ない、以下に示すようにその動作を確認した。

- データレジスタ (DELAY0) への値の読み書き
  “ff:1111_1111” の状態にあるインストラクションレジスタに値” 09:0000_1001” を書き込み、DELAY0 レジスタを書き込みモードにし、DELAY0 レジスタに対してパターン”101” を書き込んだ。さらにインストラクションレジスタに値”08:0000_1000” を書き込み、DELAY0 レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。またこのパターンの他にも幾つかのパターンに関してテストし正しいことを確認した。(図 C.9)

- データレジスタ (DELAY1) への値の読み書き
  インストラクションレジスタに値” 11:0001_0001” を書き込み、DELAY1 レジスタを書き込みモードにし、DELAY1 レジスタに対してパターン”111” を書き込んだ。さらにインストラクションレジスタに値”10:0001_0000” を書き込み、DELAY1 レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。またこのパターンの他にも幾つかのパターンに関してテストし正しいことを確認した。(図 C.10)

- データレジスタ (DELAY2) への値の読み書き
  インストラクションレジスタに値” 19:0001_1001” を書き込み、DELAY2 レジスタを書き込みモードにし、DELAY2 レジスタに対してパターン”101” を書き込んだ。さらにインストラクションレジスタに値”10:0001_0000” を書き込み、DELAY2 レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。(図 C.11)

- データレジスタ (DELAY3) への値の読み書き
  インストラクションレジスタに値" 21:0010_0001" を書き込み、DELAY3 レジスタを書き込みモードにし、DELAY3 レジスタに対して 3bit のパターンを書き込んだ。さらにインストラクションレジスタに値"20:0010_0000" を書き込み、DELAY3 レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。

- データレジスタ (DELAY4) への値の読み書き
  インストラクションレジスタに値" 29:0001_1001" を書き込み、DELAY4 レジスタを書き込みモードにし、DELAY4 レジスタに対してパターン"101" を書き込んだ。さらにインストラクションレジスタに値"28:0010_1000" を書き込み、DELAY3 レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。(図 C.12)

- データレジスタ (DELAY5) への値の読み書き
  インストラクションレジスタに値" 31:0011_0001" を書き込み、DELAY5 レジスタを書き込みモードにし、DELAY5 レジスタに対してパターン"101" を書き込んだ。さらにインストラクションレジスタに値"30:0011_0000" を書き込み、DELAY3 レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。(図 C.13)

- データレジスタ (DELAY6) への値の読み書き
  インストラクションレジスタに値" 39:0011_1001" を書き込み、DELAY6 レジスタを書き込みモードにし、DELAY6 レジスタに対して 3bit のパターンを書き込んだ。さらにインストラクションレジスタに値"38:0011_1000" を書き込み、DELAY6 レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。

- データレジスタ (GLINK) への値の読み書き
  インストラクションレジスタに値" 41:0100_0001" を書き込み、GLINK レジスタを書き込みモードにし、GLINK レジスタに対してパターン"101" を書き込んだ。さらにインストラクションレジスタに値"40:0400_0000" を書き込み、GLINK レジスタを読みだしモードにし、このパターンを TDO から読みだし、この間の動作が正しいことを確認した。(図 C.14)

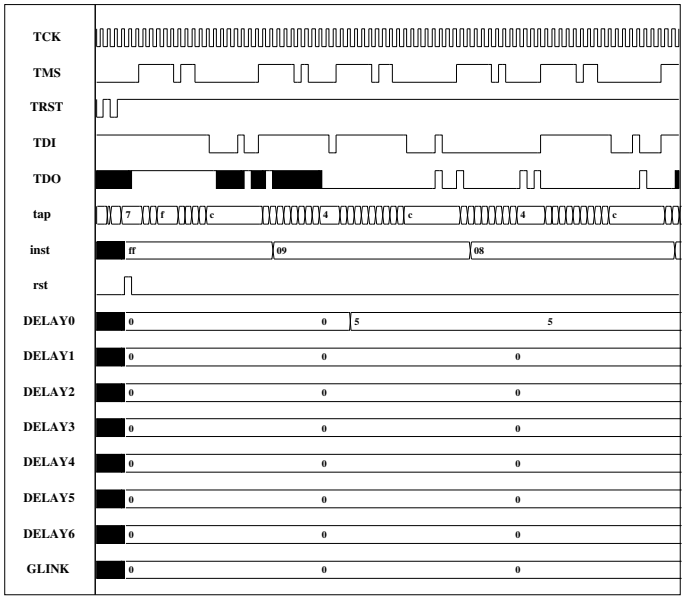


図 C.9: JTAG プロトコルの波形、データレジスタ DELAY0 への値の読み書き
インストラクションを”09:0000_1001” にして、DELAY0 レジスタを書き込みモードにし、パターン”101” を書き込んだ。さらにインストラクションを”08:0000_1000” として、DELAY0 レジスタを読みだしモードにし、このパターンを TDO から読みだしている。

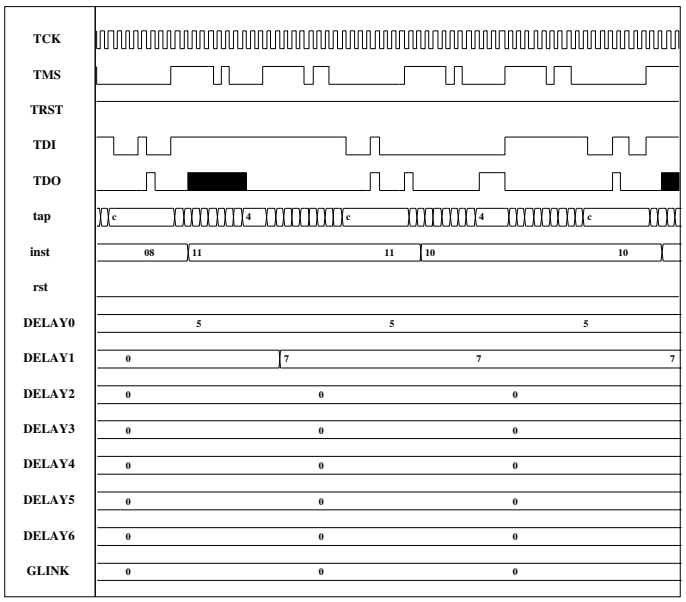


図 C.10: JTAG プロトコルの波形、データレジスタ DELAY1 への値の読み書き
インストラクションを”11:0001_0001” にして、DELAY1 レジスタを書き込みモードにし、パターン”101” を書き込んだ。さらにインストラクションを”10:0001_0000” として、DELAY1 レジスタを読みだしモードにし、このパターンを TDO から読みだしている。

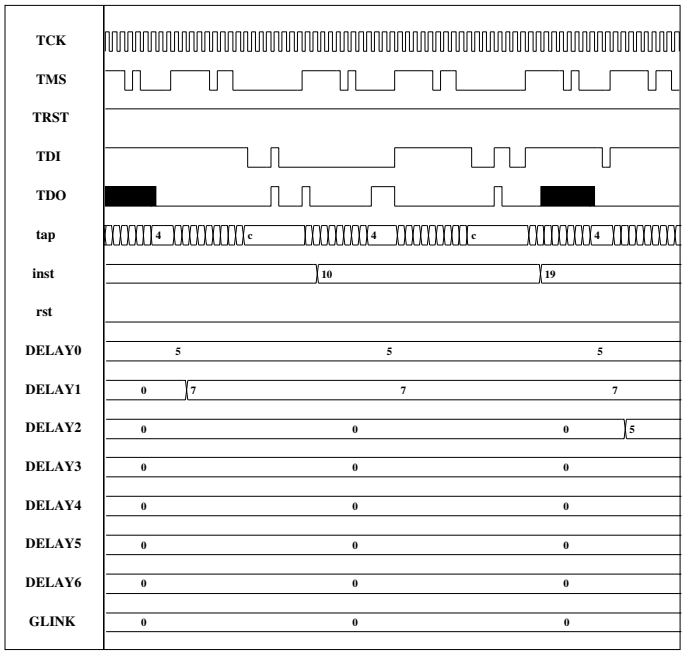


図 C.11: JTAG プロトコルの波形、データレジスタ DELAY2 への値の読み書き
インストラクションを”19:0001_1001” にして、DELAY2 レジスタを書き込みモードにし、パターン”101” を書き込んだ。さらにインストラクションを”18:0001_1000” として、DELAY2 レジスタを読みだしモードにし、このパターンを TDO から読みだしている。

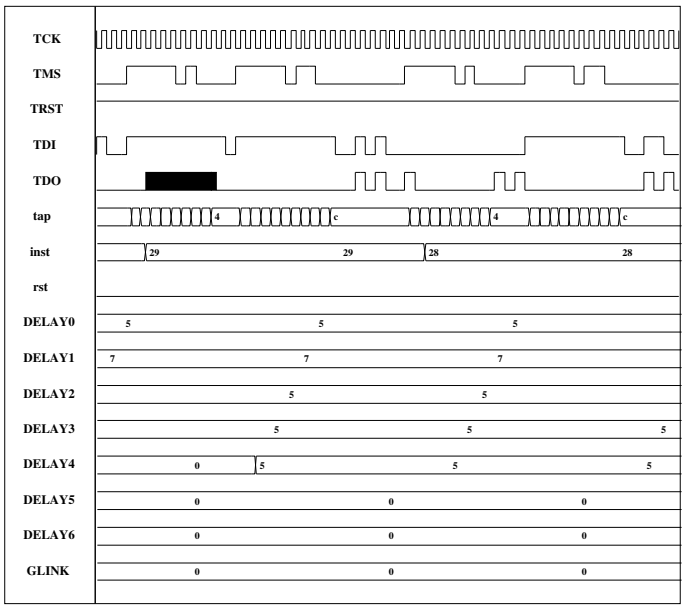


図 C.12: JTAG プロトコルの波形、データレジスタ DELAY4
インストラクションを”29:0010_1001” にして、DELAY4 レジスタを書き込みモードにし、パターン”101” を書き込んだ。さらにインストラクションを”28:0010_1000” として、DELAY4 レジスタを読みだしモードにし、このパターンを TDO から読みだしている。

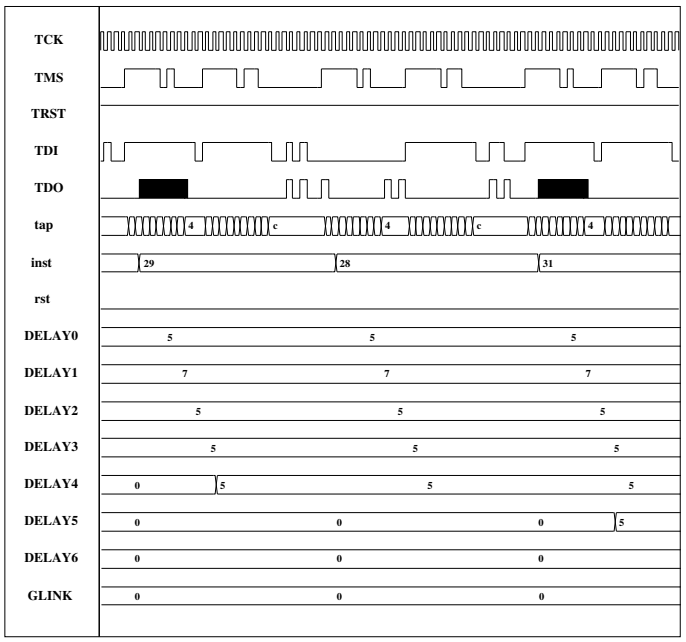


図 C.13: JTAG プロトコルの波形、データレジスタ DELAY5
インストラクションを”39:0011_1001” にして、DELAY5 レジスタを書き込みモードにし、パターン”101” を書き込んだ。さらにインストラクションを”38:0011_1000” として、DELAY5 レジスタを読みだしモードにし、このパターンを TDO から読みだしている。

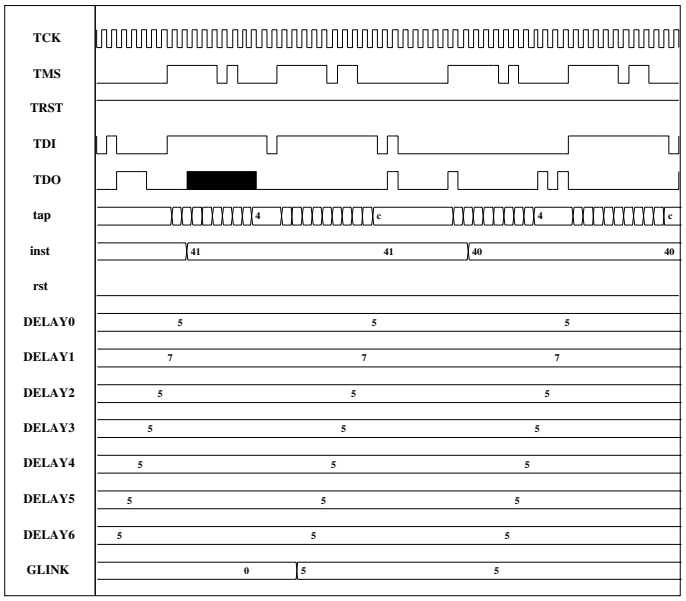


図 C.14: JTAG プロトコルの波形、データレジスタ GLINK
インストラクションを”41:0100_0001” にして、GLINK レジスタを書き込みモードにし、パターン”101” を書き込んだ。さらにインストラクションを”40:0100_0000” として、GLINK5 レジスタを読みだしモードにし、このパターンを TDO から読みだしている。

参考文献

[1] ATLAS Technical Proposal, CERN/LHCC (1994)

[2] ATLAS Detecter and Physics Performance Technical Design Report (1999)

[3] ATLAS Muon Technical Design Report, CERN/LHCC (1997)

[4] ATLAS Level-1 Trigger Technical Design Report, ATLAS TDR-12 (1998)

[5] 東京大学 松浦聡 修士学位論文「ATLAS 実験 前後方部ミューオントリガーシステムの開発」(1999)

[6] 東京大学 仁木太一 修士学位論文「ATLAS 実験ミューオントリガー用タイミング調整 IC の開発」(2000)

[7] 東京大学 戸谷大介 修士学位論文「ATLAS 実験ミューオントリガーシステムにおける運動量選別用 ASIC の開発」(2000)

[8] 京都大学 西田昌平 修士論文「ATLAS TGC エレクトロニクス 読み出し系の開発」(2000)

[9] 坂巻著 CQ 出版社 「JTAG テストの基礎と応用」

[10] IEEE Std 1149.1(JTAG) Testability,Texas instrument(1997)

[11] IEEE Standard Test Access Port and Boundary-Scan Architecture,IEEE Computer Society (1993)

[12] VDEC 監修 浅田邦博編 「ディジタル集積回路の設計と試作」(2000)

[13] 柴田直 他 平凡社「VLSI テクノロジー入門」(1986)

[14] 富沢孝 他 丸善 「CMOS VLSI 設計の原理」(1998)

[15] Martion Dentan,ATLAS Policy on Radiation Tolerant Electronics ATLAS project Document (2000)

[16] ATLAS Radiation Tolerance Criteria,ATLAS projetct Document(ATC-TE-QA-0001)(2000)

[17] Michael Shupe, RADIATION LEVELS IN THE ATLAS DETECTER
http://isnwww.in2p3.fr/atlas/andrieux/mshupe.html

[18] J Fulcher, Single Event Upset Studies on the APV6 Front End Readout Chip (Proceedings of the forth workshop on electronics for LHC experiments)(1998)

謝辞

本研究を行なう機会、また適切な指導、助言を頂いた小林富雄教授 [A] に深く感謝致します。また本研究において終始懇切丁寧に指導して頂いた佐々木修氏 [B] に感謝致します。またエレクトロニクスグループにおいて様々な指導、助言を頂いた蓮子和巳氏 [A]、坂本宏氏 [C]、池野正弘氏 [B]、藏重久弥氏 [D]、福永力氏 [E]、新井康夫氏 [B]、大須賀鬨雄氏 [B]、長谷川庸司氏 [H]、に感謝致します。さらに高エネルギー加速器研究機構において貴重な意見を頂いた近藤敬比古氏 [B]、岩崎博行氏 [B]、田中秀治氏 [B]、山内一夫氏 [B] 他 ATLAS 日本グループの方々に心から感謝致します。またエレクトロニクスグループで ASIC 開発などで度々貴重な助言を頂いた狩野博之氏 [E]、松浦聡氏 [G]、仁木太一氏 [G]、西田昌平氏 [C]、戸谷大介氏 [G]、に深く深く感謝致します。またエレクトロニクスグループで共に研究を行なった一宮亮氏 [D]、溝内健太郎氏 [C]、田中賢一氏 [E]、石田康明氏 [E]、戸塚真義氏 [H]、に深く感謝致します。研究生活を通じて惜しみない協力を頂いた、浅井祥仁氏 [A]、陣内修氏 [G]、吉田光宏氏 [G]、佐藤構二氏 [G]、津野総司氏 [G]、鈴木修氏 [D]、宮島明里氏 [H]、南篠創氏 [G]、長島荘洋氏 [G] 中村佳央氏 [G] に感謝致します。また池田誠氏を始めとする VDEC (東京大学大規模集積システム設計教育研究センター) の方々、ローム (株)、凸版印刷 (株) の方々に感謝致します。

また本論文の ASIC の試作は東京大学大規模集積システム設計教育研究センターを通しローム (株) および凸版印刷 (株) の協力で行われたものです。

所属：東京大学素粒子物理国際研究センター [A]、高エネルギー加速器研究機構 [B]、京都大学理学研究科 [C]、神戸大学自然科学研究科 [D]、東京都立大学理学研究科 [E]、東京大学理学系研究科 [G]、信州大学理学系研究科 [H]