

携帯型放射線検出器
多チャンネル読み出しロジック回路の開発

広島大学理学部物理科学科
高エネルギー宇宙・可視赤外線天文学研究室

B065458

平野敬純

主査: 深澤泰司 / 副査: 佐々木茂美

日付:2013/02/08

概要

一般的な携帯型放射線検出器はガイガー=ミュラー管を用いた放射線量計測器であるが、代わりにシンチレータとMPPCを使うことにより放射線スペクトルを取得し核種同定ができる。さらに、多チャンネル読み出しが可能になるとコインシデンス設定でバックグラウンド除去, ミューオン検出器, コンプトンカメラとして用いることが出来るようになるが、当研究グループのデータ処理用の小型ロジック回路は1チャンネル読み出しにしか対応していないもの、あるいは多チャンネル読み出し可能ではあるが特定の検出器にしか用いることができないものしかない。

そこで、今回の研究では放射線の多チャンネル読み出し可能な集積回路FPGA用汎用ロジック回路を開発し、携帯型の放射線検出器でバックグラウンドの除去やコンプトンカメラやミューオン検出などで使用できることを目的とする。

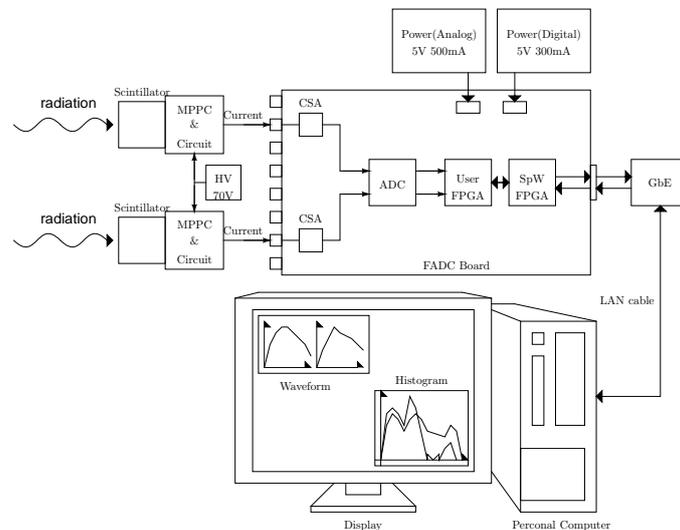


図 1: 実験装置の略図

- 第一版 20120208 提出

目次

第 1 章 序論	6
1.1 背景	6
1.2 多チャンネル検出器で可能になること	6
1.3 研究の目的	7
第 2 章 放射線と放射線検出器の基本	9
2.1 放射線とは	9
2.2 放射線の種類	9
2.2.1 α 線	9
2.2.2 β 線	9
2.2.3 X 線, γ 線	10
2.2.4 中性子線	10
2.2.5 ミューオン (μ 粒子)	10
2.3 放射線の放出	10
2.3.1 α 崩壊	11
2.3.2 β^- 崩壊	11
2.3.3 β^+ 崩壊	12
2.3.4 γ 崩壊	12
2.3.5 対消滅	12
2.3.6 自発核分裂	13
2.3.7 チェレンコフ放射	13
2.4 放射線源	14
2.5 放射線検出器の種類	14
2.5.1 電離作用を利用した放射線検出器	14
2.5.2 蛍光作用を利用した放射線検出器	17
2.5.3 多チャンネルの放射線検出器	18
2.6 FPGA と VHDL	19
2.6.1 FPGA	19
2.6.2 VHDL	19
第 3 章 PoGOLino 用 3ch プログラム	22
3.1 PoGOLino とは	22
3.2 PoGOLite 気球実験	22

3.3	PoGOLino 用検出器	23
3.3.1	FADC ボード	23
3.4	PoGOLino 用ロジック回路	25
3.4.1	RMAP	26
3.4.2	TriggerModule	26
3.4.3	SaveWaveModule	28
3.4.4	HistogramModule	30
第 4 章	8ch 読み出しロジック回路の開発	31
4.1	8ch 読み出しロジック回路の構成	31
4.1.1	TriggerModule	32
4.1.2	コインシデンス設定	32
4.1.3	WaveFormModule : 波形保存	36
4.1.4	HistogramModule : エネルギーヒストグラム生成	36
第 5 章	放射線検出実験	38
5.1	実験装置	38
5.1.1	シンチレータ (Scintillator)	39
5.1.2	MPPC& 回路 (circuit)	40
5.1.3	SpaceWire to Gigabit Ether(GbE)	40
5.2	実験 1 : ^{22}Na の対消滅 γ 線の取得	42
5.2.1	実験 1:実験環境	42
5.2.2	実験 1:放射線の入射量予想	43
5.2.3	実験 1:ゲインの調整	44
5.2.4	実験 1:コインシデンス設定	45
5.2.5	実験 1:実験結果	45
5.2.6	実験 1:考察	46
5.3	実験 2 : コンプトンカメラの基礎実験	46
5.3.1	実験 2:実験環境	47
5.3.2	実験 2:放射線の入射エネルギー予想	47
5.3.3	実験 2:ゲイン・較正直線	48
5.3.4	実験 2:実験結果	49
5.3.5	実験 2:考察	49
第 6 章	まとめと今後	50
付 録 A	多チャンネルの制御プログラム	51
A.1	checkFADC	51
A.2	trigstartup_8ch	53
A.3	coincidence_setting	55
A.4	Read_wf_8histo512_8ch	57

A.5	CompressedWave	58
A.5.1	wave.dat の保存形式	59
A.6	gnuplot によるヒストグラム表示	60
A.7	メモ (備忘録)	62
A.7.1	coincidence_setting(全取得モード)	62

目 次

1	実験装置の略図	1
1.1	実験装置の略図	6
1.2	コンプトンカメラの原理	7
1.3	ミュオン検出器の原理	7
1.4	バックグラウンド除去	8
2.1	核図表 (Web 引用)	11
2.2	チェレンコフ放射	13
2.3	電離箱	15
2.4	比例計数管	16
2.5	ガイガー=ミュラー計数管	16
2.6	半導体検出器	17
2.7	スーパーカミオカンデのミュオン粒子検出	17
2.8	コンプトン散乱	18
2.9	コンプトンカメラ	18
2.10	ミュオン検出器	19
2.11	ハーフアダー回路	20
3.1	偏光した硬 X 線の検出原理	22
3.2	PoGOLite メイン検出部	22
3.3	PoGOLite 用の多チャンネル用検出器の構成	23
3.4	PoGOLino の検出器	24
3.5	PoGOLino のシンチレータ	24
3.6	FADC ボード	24
3.7	SpaceWire ケーブル	25
3.8	SpaceWire コネクタ	25
3.9	PoGOLino 用プログラムの略図	26
3.10	シンチレータの立ち上がりの違い	27
3.11	BGO の波形と各ディレイ	27
3.12	ディレイの差分	27
3.13	Fast 信号の差分	28
3.14	Middle 信号の差分	28
3.15	Slow 信号の差分	28

3.16	ビットシフト	28
3.17	1 波形の保存構成	29
3.18	波形保存を間引く	30
3.19	ch1 ヒストグラム例	30
3.20	ch2 ヒストグラム例	30
3.21	ch3 ヒストグラム例	30
4.1	8ch 読み出しロジック回路の構成ダイアグラム	31
4.2	コインシデンスの条件数	32
4.3	ミュオン入射位置検出器	33
4.4	コインシデンスアルゴリズム (例)	36
4.5	波形取得プログラム	37
4.6	ヒストグラム	37
5.1	実験装置の略図	38
5.2	MPPC(S10362-33-050C)	40
5.3	MPPC によるシンチレーション光の読み出し回路	41
5.4	シンチレータと MPPC	41
5.5	読み出し回路 (左に HV, 右へ出力)	41
5.6	SpaceWire to Gigabit Ether(GbE)	42
5.7	実験セットアップ図	42
5.8	$^{137}\text{Cs}(661\text{keV})$ を用いたゲイン測定	45
5.9	Na22 のコインシデンス設定による違い	46
5.10	実験 2 コンプトン散乱の検出 ($\theta=90$ 度)	47
5.11	実験 2 コンプトン散乱の検出 ($\theta=45$ 度)	47
5.12	実験装置の取得 γ 線量 (90 度)	47
5.13	実験装置の取得 γ 線量 (45 度)	47
5.14	較正用ヒストグラム (^{137}Cs と ^{22}Na)	48
5.15	実験 2 結果 (90 度)	49
5.16	実験 2 結果 (45 度)	49
A.1	Threshold,UD_Threshold,MABIKU のイメージ図 (左:MABIKU=0)(右:MABIKU=1)	54
A.2	Delay の波形	54
A.3	ミュオン入射位置検出器	56
A.4	コインシデンスロジック	56
A.5	一波形の保存データ構成	59
A.6	gnuplot 表示例	60

第1章 序論

1.1 背景

研究室の所有する多チャンネル入力可能な PoGO Lite 気球実験用検出器のデータ処理回路である FADC ボードは、複数の FADC ボードと DIO ボードを用いての使用を想定しているため FADC ボード単体で動かすことのできないロジック回路が搭載されている。また、FADC ボード単体で動かせるロジック回路は 1 チャンネル用のものがあるが、多チャンネル同時入力に対応したロジック回路は無い。

その為、FADC ボード単体で動かせる多チャンネル読み出しロジック回路の開発を行う。その際にコインシデンス設定 (複数チャンネルの同時発生による分岐) を出来るようにする。

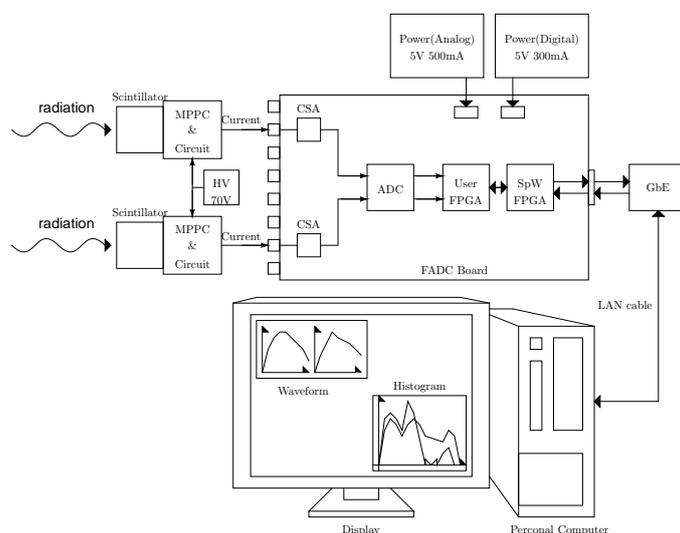


図 1.1: 実験装置の略図

1.2 多チャンネル検出器で可能になること

複数の信号が入力可能になればコンプトンカメラ、ミュオン検出器やバックグラウンド測定など活用範囲が増える。

コンプトンカメラは入射したガンマ線が多層の半導体検出器でコンプトン散乱と吸収の 2 つの反応が起こった際に、散乱と吸収の位置とエネルギーにより入射方向を決定することの出来る超広角の γ 線検出カメラである。同時発生したコンプトン散乱と吸収のそれぞれの位置とエネルギーより、以下の式からコンプトン散乱と吸収した点をつなぐ直線からの角度 θ 方向を特定することが出来る。

$$\cos\theta = 1 + \frac{m_e c^2}{E_1 + E_2} - \frac{m_e c^2}{E_1} \dots\dots \left\{ \begin{array}{l} \theta \quad : \text{入射光子の方向} \\ E_1, E_2 \quad : \text{コンプトン散乱エネルギー, 吸収エネルギー} \\ m_e \quad : \text{電子の質量} \end{array} \right. \quad (1.1)$$

さらに複数回測定することで γ 線源の位置を絞り込むことができ方向を特定することが出来る。[6]

また、ミューオン検出にも応用することが出来る。図 1.3 のように細長い検出器を並べて交互に重ねた層を作る。2 層で縦軸と横軸の位置が分かるので、複数のシンチレータを距離を離して重ねることで距離の離れた 2 点の位置が特定できるため、点をつなぐことによりミューオンの入射方向を特定することが出来る。

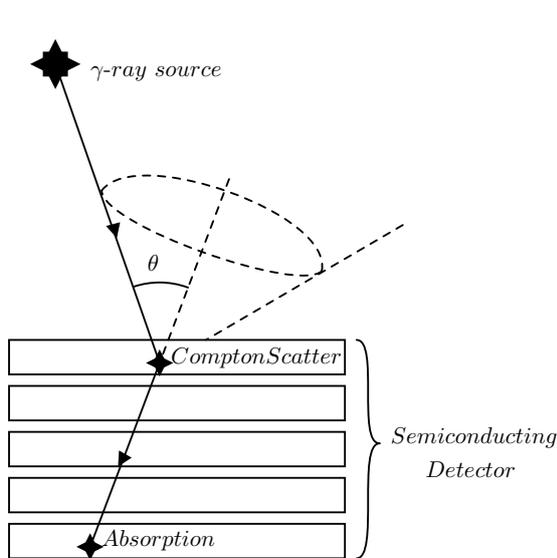


図 1.2: コンプトンカメラの原理

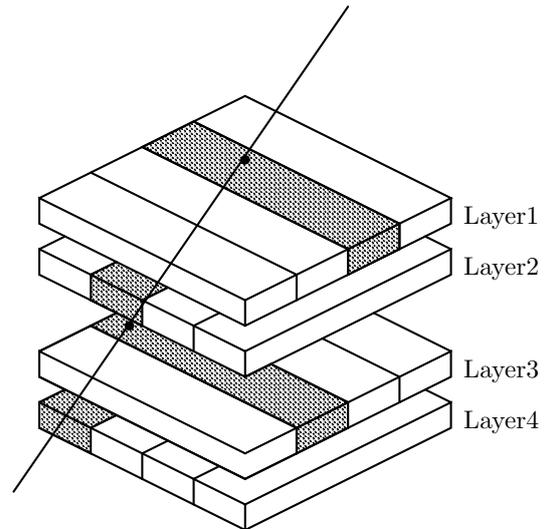


図 1.3: ミューオン検出器の原理

更に、図 1.4 の様に検出器をシンチレータで覆い入力信号を取得することでバックグラウンドの測定や観測方向以外からの入力をカットする事も可能になる。メイン検出器の反応以外にシンチレータが反応があった場合、それは視野外から入った信号であると考えることが出来る (アンチコインシデンス)。

1.3 研究の目的

研究室の所有する小型検出器 1 チャンネル読みだし可能な FPGA 用ロジック回路を改造して多チャンネル信号を読み出し、波形保存や放射線エネルギーのヒストグラムの生成を可能にするロジック回路を開発する。その際に VHDL で設計された 1 チャンネル用の FPGA 用ロジック回路設計のアルゴリズムを 8ch 用書き換える。

また、コインシデンス設定 (入力信号の同時発生時の挙動) の条件を行い、複数のシンチレータが同時に反応したデータのみを取得もしくは排除することでミューオン検出器やコンプトンカメラなどへの使用を可能にする。

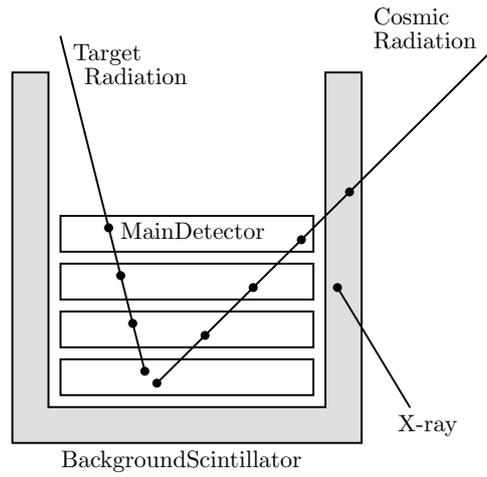


図 1.4: バックグラウンド除去

ロジック回路作成後は波形保存やエネルギーヒストグラムの生成が正常に行われているかについて検証を行い、実際にバックグラウンドの除去やコンプトンカメラやミュオン検出器などの多チャンネルでの使用が可能であるかどうかの評価を行なっていく。

第2章 放射線と放射線検出器の基本

2.1 放射線とは

放射線とは空気を電離する電磁波や粒子線の事を言う。放射線は直接人間の DNA などの重要な生体分子を傷つけたり、水分子を分解した際に発生する活性酸素によって間接的に傷つけるガンの元とも言われる危険なものである。しかし、身近な建築資材などに微量に含まれる放射性同位体や、宇宙空間に飛び交う放射線が地表まで飛来する宇宙線等の放射線を日常的に放射線は浴びている。そして、人間には損傷した DNA を修復する機能が備わっているため少量の欠損であれば再生できる (二重螺旋構造の DNA はお互いに遺伝子情報を同期しており、片方の螺旋が損傷であれば健康な遺伝子情報より損傷部分を修復できる)。

一概に放射線と言っても色々な種類があり、有効な防護方法が違って来る。この章では放射線の種類や対策、検出器に関連した知識を深めて行く。

2.2 放射線の種類

放射線は物質を通過する際に原子等を電離させる物を総称して呼ぶ。この放射線はまず大きく分けて電磁波と粒子線とに分けられ、粒子線はさらに α 線や β 線 (電子線・陽電子線), 中性子線などに分けられる。電離過程には原子核を直接電離するものや電子にエネルギーを与えることで電離させる間接電離があり、発生方法の違いや物質の透過度、有効な遮蔽方法などが違って来る。

ここでは代表的な放射線である α 線, β 線, X 線・ γ 線, 中性子線, ミューオンについて解説していく。

2.2.1 α 線

α 線は高い運動エネルギーを持つヘリウム原子核 (α 粒子) からなる放射線である。後述の α 崩壊によって生成される。原子や分子に衝突した際に励起・電離するエネルギーを与える。 α 粒子は +2 の電荷を持って居るためローレンツ力や電場磁場によって曲げることができ、また相互作用しやすいため紙 1 枚程度で遮蔽できる。ただし、電離能力が他の放射線に比べて高く、人体に与える損傷は他の放射線に比べて 5-20 倍と非常に高いので、 α 線を出す物質を体内に取り込んで内部被曝しない様に十分注意する必要がある。

2.2.2 β 線

β 線は β^- 崩壊, β^+ 崩壊によって放出される電子線, 陽電子線である。電荷を β^-, β^+ でそれぞれ $-1, +1$ 持つため、 α 線同様ローレンツ力や電場磁場によって曲げることが出来る。 β 線は数 mm の厚さのアルミニウム板で防ぐことが出来る。

2.2.3 X線, γ 線

X線, ガンマ線は波長が $10nm$ 以下の非常に短い電磁波の事を指す。元々の X線と γ 線の定義は、X線が原子核外で放出されたもので、 γ 線は原子核内からの励起で発生するといったものである。波長が $1pm - 10nm$ 程度の物を X線、 $1pm$ よりも短い物を γ 線と分けることもある。種類分けは発生仕方によって区別している。

X線, γ 線は電荷を持たない電磁波で、物質と作用して物質中の電子が光電効果, コンプトン散乱, 電子対生成などによりエネルギーを受け取る事で電離する。X線, γ 線は質量の大きい (原子番号の大きい) 物質によって弱めることが出来るため、数 cm の鉛のブロックによって弱めることが出来る。

2.2.4 中性子線

中性子線は電荷を持たないため透過力が非常に高く、物質中の原子や原子核と衝突した際にエネルギーを放出していく。エネルギーの放出は以下の式に有るように質量数 A の小さい軽い原子に多くのエネルギーを放出傾向があるため、水素原子や水素を含む分子は電離しやすい。ちなみに θ は物質と中性子線の衝突の位置によって決まるため、エネルギーの放出に関してはランダム要素と考えることが出来る。

$$E_A = \frac{2A}{(A+1)^2} E_n (1 - \cos\theta) \dots\dots \begin{cases} E_A & : \text{物質の反跳エネルギー} \\ E_n & : \text{中性子のエネルギー} \\ \theta & : \text{散乱方向} \\ A & : \text{質量数} \end{cases} \quad (2.1)$$

以上の事より、原子番号の小さい水素を多く含む水 H_2O やコンクリート、プラスチックなどに効率的に遮断することが出来る。

2.2.5 ミューオン (μ 粒子)

ミューオンは第二世代の荷電レプトンである。第一世代の電子の質量 ($0.511 MeV/c^2$) に比べて約 200 倍重い質量 ($106 MeV/c^2$) を持つ。ミューオンは他の宇宙線に比べて減少率が非常に低く、安定な硬成分であるため、地上からでも十分に観測することの出来る粒子である。数 cm の鉄板であれば通り抜けることができる。宇宙線ミューオンは二次宇宙線である pi 中間子が崩壊することでミューオンとニュートリノを生成する。

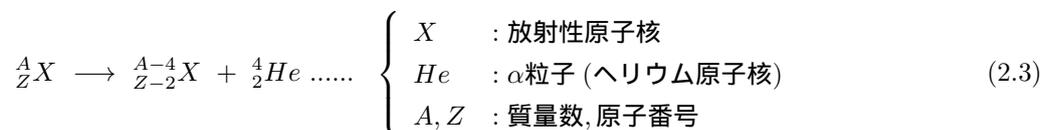
$$\begin{cases} \pi^+ \longrightarrow \mu^+ + \nu_\mu \\ \pi^- \longrightarrow \mu^- + \bar{\nu}_\mu \end{cases} \quad (2.2)$$

2.3 放射線の放出

放射線は放射性物質の崩壊、あるいはサイクロトロンや X線発生装置などで発生させることもできる。代表的な α 崩壊、 β^- 崩壊、 β^+ 崩壊、 γ 崩壊、自発核分裂、チェレンコフ放射について見ていく。

2.3.1 α 崩壊

α 崩壊は不安定な重い原子核が分裂して α 粒子 (=ヘリウム原子核) を放出する。放出した原子核はより安定した原子核へ変化し、 α 粒子を放出した分だけ軽くなった原子核になる。式で表すと以下のように表される。



不安定な重い原子核であればなんでも α 崩壊を起こすわけではなく、 α 崩壊を起こす代表的な物質はウラン 238, ウラン 234, トリウム 230, ラジウム 226, ラドン 222 などがある。

2.3.2 β^- 崩壊

原子核にはそれぞれ原子によって安定な陽子数と中性子数の関係がある。例えば酸素 O であれば陽子数 8 個に対して中性子数 8 個の ${}^{16}O$ が最も安定で、次いで中性子数 9 個の ${}^{17}O$, 中性子数 10 個 ${}^{18}O$ が安定同位体である。しかし、中性子数がそれ以上もしくはそれ以下であれば放射性同位体核種となる。中でも安定同位体よりも中性子数が多い原子核が β^- 崩壊を起こす。 β^- 崩壊を起こすと中性子が電子と反電子ニュートリノを伴って陽子に崩壊する。

安定な陽子と中性子との比率は軽い原子のうち陽子 : 中性子 = 1 : 1 だが、重くなるにつれて中性子の比率が増えてくる。

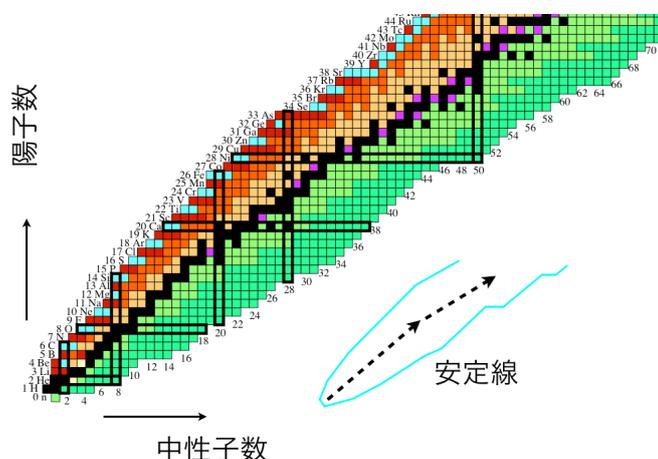


図 2.1: 核図表 (Web 引用)

(メモ : 核図表 http://www.rri.kyoto-u.ac.jp/NBMP/study/kaku_reactor.html)

核図表は中央の黒ラインが安定同位体であり、また自然中に存在する比率も多い。この安定同位体のライン

よりも中性子数が多い緑の領域の原子は β^- 崩壊を起こし、逆に中性子数が少ない赤の領域の原子は後述の β^+ 崩壊を起こす。 β^- 崩壊は式で表すと以下の様に表される。

$$n \rightarrow p + e^- + \bar{\nu}_e \dots\dots \left\{ \begin{array}{l} n, p : \text{中性子, 陽子} \\ e^- : \text{電子} \\ \bar{\nu}_e : \text{反電子ニュートリノ} \end{array} \right. \quad (2.4)$$

α 崩壊と違い軽い原子核でも放射性同位体核種であれば β^- 崩壊は発生する。また中性子から陽子への変化なので原子の質量自体には変化がないが、陽子数が増えているため原子番号が1つ大きい元素に変化する。

2.3.3 β^+ 崩壊

β^+ 崩壊は前述の β^- 崩壊とは逆に安定同位体よりも中性子数が少ない場合に起こる。具体的には β^+ 崩壊を起こすと物質中の陽子が陽電子と電子ニュートリノを放出して中性子に崩壊する。 β^+ 崩壊を指揮で表すと以下のように表される。

$$p \rightarrow n + e^+ + \nu_e \dots\dots \left\{ \begin{array}{l} p, n : \text{陽子, 中性子} \\ e^+ : \text{陽電子} \\ \nu_e : \text{電子ニュートリノ} \end{array} \right. \quad (2.5)$$

β^- 崩壊同様原子の質量数自体には変化がないが、陽子数が減っているため原子番号が1つ小さい原子に変化する。 β^- 崩壊と β^+ 崩壊は全く逆の変化ではあるが、どちらも不安定な放射性同位体から安定同位体へ崩壊する事が共通する。

2.3.4 γ 崩壊

γ 崩壊は励起された原子核がガンマ線を放出して安定した原子核になる崩壊である。 γ 線は α 崩壊や β^\mp 崩壊と違い核種が変わらない。励起状態の原子核がガンマ線を放出する時間は $10^{-10}s$ と非常に短く、 α 崩壊や β 崩壊で生成された娘核種が励起状態の際に放出するため、放射性物質の放出する放射線に α 崩壊や β 崩壊で生成された娘核種が放出する γ 線も同時に放出すると表現されることが有る。

2.3.5 対消滅

対消滅は電子陽電子の様に電荷が対になっている粒子と反粒子が衝突した際にエネルギーや他の粒子に変換される現象である。 ^{22}Na が放出する 511keV の γ 線はこの対消滅によって生成されるもので、 β^+ 崩壊で放出された陽電子が周囲の電子と衝突し、 511keV のエネルギーを持った2つの γ 線が放出される。放出する方向は互いに反対方向に放出される。

2.3.6 自発核分裂

ウランなどの重い原子核が中性子などの放射線を受けた際に核分裂するが、自発核分裂は質量数の非常に大きな原子核自身が耐えられなくなり2つ以上に分裂する。自発核分裂が起こる条件は以下の式で近似的に与えられる。

$$\frac{Z^2}{A} \geq 45 \dots\dots \left\{ \begin{array}{l} Z, A : \text{原子番号, 質量数} \end{array} \right. \quad (2.6)$$

実際に原子核分裂が起こる原子はトリウム付近以上の原子から可能になる。

2.3.7 チェレンコフ放射

チェレンコフ放射は物質中で荷電粒子の速度 $v_{particle}$ が光の速度 v_c よりも速く通過する時に起こる。この時の光の速度は物質の屈折率による影響を受けた速度で、物質中の屈折率を n とすると物質中の光の速度 v_c は $(v_c = c/n)$ となる。発生原理は音速を超えたジェット機が発生させる衝撃は(ソニックブーム)と同じで、高速で駆け抜ける荷電粒子が生成する電磁波が一定の方向に対して位相が揃い強め合う事で発生する。チェレンコフ放射の方向は以下の式より求めることができる。

$$\cos\theta = \frac{v_{particle}t}{v_c t} = n \frac{v_{particle}}{c} \dots\dots \left\{ \begin{array}{l} v_{particle} : \text{放出される振動数間隔} \\ v_c \end{array} \right. \quad (2.7)$$

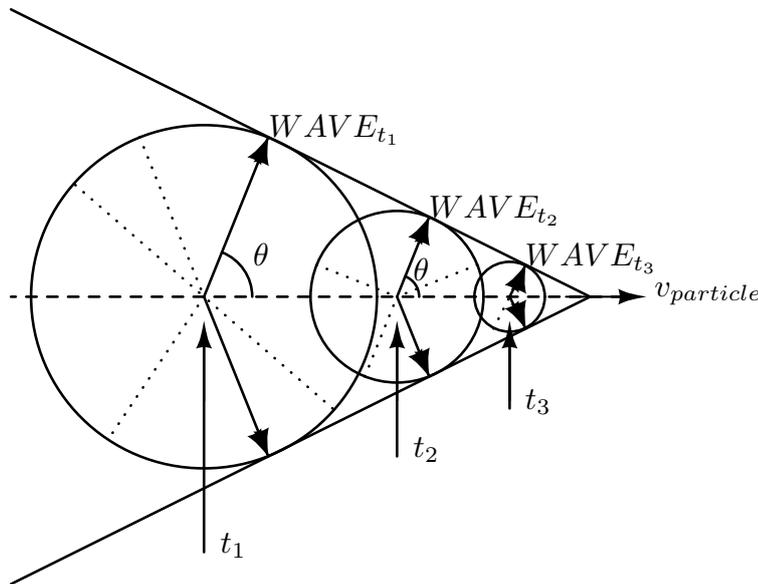


図 2.2: チェレンコフ放射

2.4 放射線源

今回の実験で使った線源を以下の表にまとめた。

放射線物質	半減期	放射線	崩壊エネルギー (keV)	備考
セシウム ^{137}Cs	30.1y	β 線	512,1176 keV	β^- 崩壊
		γ 線	661 keV	β^- 崩壊後の ^{137m}Ba の γ 崩壊
コバルト ^{60}Co	5.27y	β 線	318 keV	β^- 崩壊
		γ 線	1170,1330 keV	β^- 崩壊後の ^{60m}Ni の γ 崩壊
ナトリウム ^{22}Na	2.6y	γ 線	1274 keV	β^+ 崩壊後の ^{22}Ne の γ 崩壊
		γ 線	511 keV	β^+ 崩壊で放出する陽電子の対消滅
カリホルニウム ^{252}Cf	2.65y	中性子線		自発核分裂

放射性物質にはそれぞれ半減期があり、半減期経つごとに全体の 50% が崩壊する。それに従って単位時間当たりの放射線放出量 Bq も少なくなってくるので、放射線源として使用する場合 (特に半減期の短いもの) は注意する必要がある。半減期による放射量の減少は以下の式より求めることが出来る。

$$N = N_0 \times \exp\left(-0.693 \frac{t}{t_{half}}\right) \dots\dots \left\{ \begin{array}{l} N : \text{現在の放射線放出量} \\ N_0 : \text{元の放射線放出量} \\ t : \text{経過時間} \\ t_{half} : \text{半減期} \end{array} \right. \quad (2.8)$$

例えば、半減期が 2.6 年の ^{22}Na の線源が 4ヶ月前 (0.33year) に 100kBq であった場合、現在の放出量は以下の式より求められる。

$$N = 100 \times \exp\left(-0.693 \frac{0.33}{2.6}\right) = 91.5[\text{kBq}] \quad (2.9)$$

Bq(ベクレル) は単位秒あたりの原子核の崩壊割合であるので、この線源は毎秒平均 91,500 個が崩壊を起こしている事になる。

2.5 放射線検出器の種類

どの放射線も五感では感じ取ることができないため、何らかの形に変換し視覚化する必要がある。以下では放射線検出器について紹介していく。

2.5.1 電離作用を利用した放射線検出器

電離箱

電離箱は希ガスで充満した箱の両端に 2 枚の板状の導体を設置し、導体に一定の電圧をかけたものである。希ガスに X 線が反応すると、X 線の電離作用によってイオン対が X 線のエネルギーに比例した数だけ

け発生する。導体に電圧が掛かっていることにより正イオンはマイナス電極に、負イオンはプラスの電極に引かれる事により電流が発生するので、流れた電流の量を測定する。この原理はその他の電離作用を利用した放射線検出器の軸となる基本的なものである。

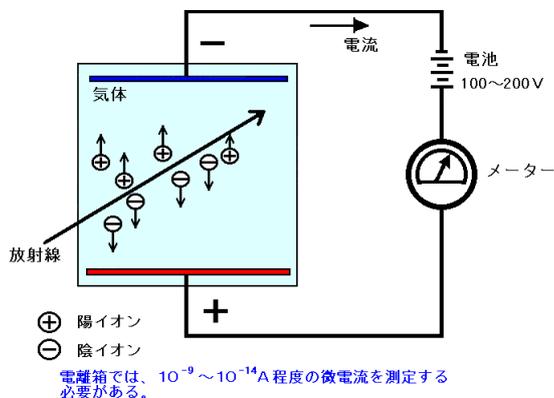


図2 電離箱による測定原理

[出典]日本原子力研究所 国際原子力総合技術センター:原子力基礎用語集(1997年)、p.67

図 2.3: 電離箱

比例計数管

比例計数管は古くから使用されてきた放射線観測機器である^[3]。アルゴンやキセノンなどの希ガスを封入した円筒型の密閉容器と容器の中心に高電圧が印加された細い芯線よりなる。容器内に入射した放射線は光電効果により希ガスに吸収され、放射線のエネルギーに比例した数の1次電子が発生する。発生する数は希ガスの平均電離エネルギーが約 $20eV$ なので、入射放射線のエネルギーが $6keV$ の場合約300個の1次電子が発生することになる。この1次電子で発生した微弱な1次電子は高電圧の細い芯線の作る電場により加速され2次電子を発生させる。

$$E(r) = \frac{V}{r \ln(b/a)} \dots\dots \begin{cases} E(r) & : \text{電場} \\ V & : \text{芯線の電圧} \\ r & : \text{芯線からの距離} \\ a, b & : \text{芯線の半径, 円筒容器の半径} \end{cases} \quad (2.10)$$

1次電子,2次電子どちらともエネルギーに比例した数の電子を作るため、その電子を測定すると入射X線のエネルギーを取り出すことが出来る。

ガイガー=ミュラー計数管

ガイガー=ミュラー計数管は放射線量を計測する検出器で、放射線の入射回数を数えることが出来る。放射線の検出原理は比例計数管と同じで、比例計数管よりも高電圧を掛ける。それによって弱い放射線にも反

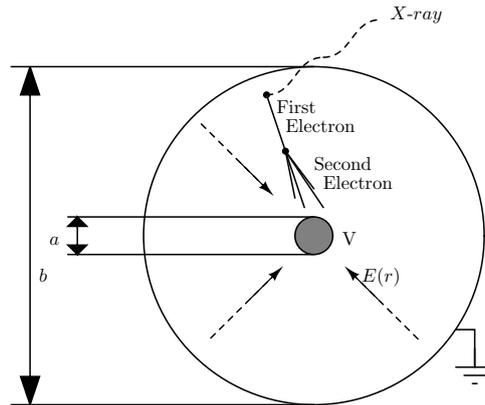


図 2.4: 比例計数管

応することが出来るようになり、発生したパルス電流の回数を数えることで入射してきた放射線量を計測することが出来る。しかし、比例計数管と違い2次電子の増倍率は比例関係でなくなるのでエネルギーは想定することができなくなる。

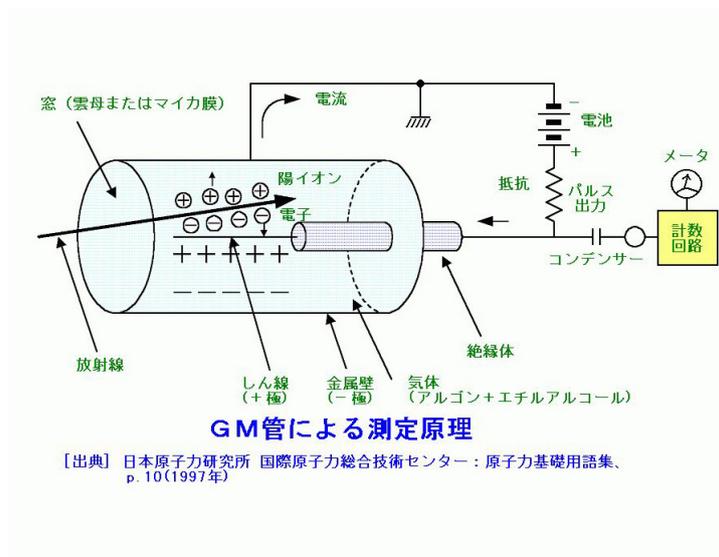


図 2.5: ガイガー=ミュラー計数管

半導体検出器

半導体検出器はガスの代わりにシリコンやゲルマニウムなどの半導体を用いて検出を行なう。純粋な真性半導体に少量の不純物をドーピング(付加)することで2つの異なる性質(P型,N型)を持つ。P型は原子の共有結合で使われずに余った負の電荷を持ち、N型は共有結合に必要な電子が不足した正の電荷を持つ。このP型,N型の半導体を接合させ、両端から電流の流れない方に電圧を掛けることで半導体内に空乏層がで

きる。この空乏層に放射線が入射すると光電効果により半導体内でイオン対が生成され、逆バイアス電場によって電極側に集まる。この電流を測定することで分解能の良いエネルギーを取得することが出来る。

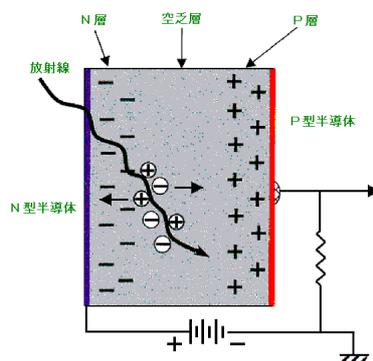


図4 半導体検出器(P-N接合)の動作原理
【出典】江藤 秀雄ほか：放射線の防護(三版)、丸善(1978年)、p.86

図 2.6: 半導体検出器

2.5.2 蛍光作用を利用した放射線検出器

チェレンコフ検出器

チェレンコフ放射による蛍光を光電子増倍管で電気信号に変えて検出する。スーパーカミオカンデで用いられており、大きな水槽には純水が満たされており、また水槽の側面や底には1万本以上の光電子増倍管が取り付けられており、外から入ってきたニュートリノが水中の粒子にぶつかった際に放出される荷電粒子がチェレンコフ放射を起こす。チェレンコフ光から散乱方向や速度等を求めることが出来る。

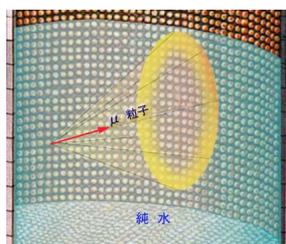


図 2.7: スーパーカミオカンデのミュオン粒子検出

シンチレーション検出器

シンチレーション検出器は、ガンマ線が入射時にシンチレーション光を蛍光するシンチレータと、微弱なシンチレーション光を電荷に変えて増幅する光電子増倍管から成る。シンチレーションは入射した放射線のエネルギーに比例したシンチレーション光を発光し、光電子増倍管で光量に比例した電荷を発生させるた

め、エネルギーを測定することが出来る。最近では高電子増倍管の代わりに半導体の光検出器を用いるものもある。詳しい性質については第 5 章参照。

2.5.3 多チャンネルの放射線検出器

コンプトンカメラ

コンプトン散乱は X 線や γ 線のエネルギーの一部を電子に与えて進行方向を変える非弾性散乱の一種である。コンプトンカメラはそのコンプトン散乱の運動学を元にしており、1 つの放射線の入射でコンプトン散乱時に与えられる反跳電子のエネルギー (E_1) と光電吸収エネルギー (E_2) の 2 つのエネルギーと散乱位置と吸収位置は以下の式で示される角度で起こる。この原理を用いてコンプトン再構成を行い、入射 γ 線の入射方向を特定する。コンプトン再構成を行なうと 2 つの反応位置から角度 θ とした円錐の斜辺方向から来たことが分かる。複数のコンプトンコーンを重ねることで図 2.9 のように入射 γ 線源の位置を特定することが出来る。

$$\cos\theta = 1 + \frac{m_e c^2}{E_1 + E_2} - \frac{m_e c^2}{E_2} \dots\dots \begin{cases} \theta & : \text{入射光子の方向} \\ E_1, E_2 & : \text{コンプトン散乱エネルギー, 吸収エネルギー} \\ m_e c^2 & : \text{電子の質量 } (\approx 511[\text{keV}]) \end{cases} \quad (2.11)$$

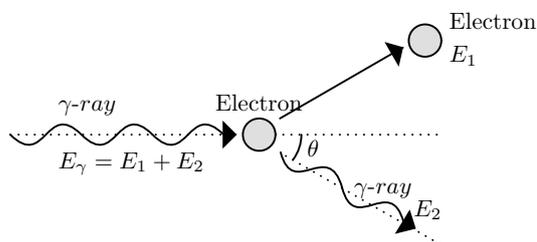


図 2.8: コンプトン散乱

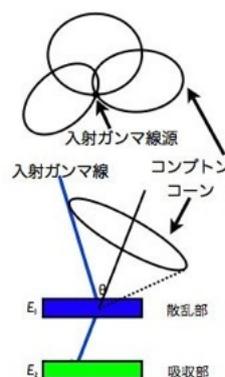


図 2.9: コンプトンカメラ

ミューオン検出器

ミューオン検出器は 2.10 のように細長い検出器を並べたレイヤーで 4 層作り、レイヤーとコンクリートでサンドイッチしたものである [15]。1 つのレイヤーの幅が 10cm の比例計数管を用いるとミューオンが入ってきた位置を 10cm の精度ではかることができ、4 層それぞれで測定したミューオンの位置を組み合わせでミューオンの入射方向を測定することが出来る。ミューオン検出器の上に 2m のコンクリートの層を置くことで空気シャワー中のほとんどの電子を吸収され、透過性の高いミューオンのみを検出しやすく成る。

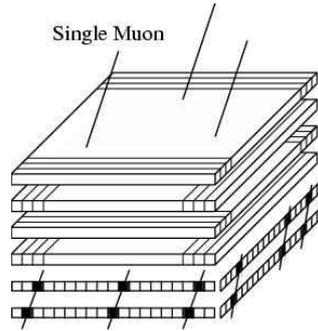


図 2.10: ミューオン検出器

2.6 FPGA と VHDL

2.6.1 FPGA

FPGA(Field Programmable Gate Array) は作業現場 (Field) でプログラム可能 (Programmable) なゲートアレイ (Gate Array) という名の通り、ユーザがプログラムをすることで集積回路内部の配線を行う事で自由に電気回路を作る事が出来るものである [16]。ゲートアレイは予めマスタスライス (AND,OR などの論理ゲート) を大量に用意しておき、マスタスライスに配線を行うことができる半導体製品である。

FPGA のメリットは「ユーザが」「集積回路の配線を行える」という所が優れている。通常の IC チップであれば回路設計を行った後で工場に発注し完成品を受け取るという工程により最低 2 週間かかるが、ユーザ自身で配線できるので設計が終われば数分で回路を使うことができる。配線する環境も特別な物を用意する必要がなく、PC と FPGA にプログラムを転送するケーブルを用意すれば良い。また、プログラムを書き換える事も自由に行う事が出来るため設計ミスをしていても気軽に修正したり、または全く別のプログラムを書き込むことも出来る。FPGA のロジック回路は後述の VHDL と呼ばれる特別な言語で記述する必要がある。

2.6.2 VHDL

プログラムには VHDL(VHSIC Hardware Description Language) を用いる。VHDL は VHSIC(Very High Speed Integrated Circuit : 超高速集積回路) 委員会の開発したハードウェア記述言語 (Hardware Description Language) で、記述能力が高い言語である。

従来の電気回路の設計であれば入力信号 (High-Low) に対し NOT 回路や AND 回路や OR 回路などを配線して電気回路を作っていくのだが、複雑な電気回路を組もうとすると非常に複雑になり設計ミスを起こしやすく、また設計ミスの問題箇所を見つけるのが難しく一つ一つ論理回路を確認していく必要があり大変な作業になる。また、設計者以外には回路のアルゴリズムをつかむ事は非常に難しい。VHDL を用いることで、C 言語と似たような感覚でロジック回路を組むことができるため設計しやすく、設計者以外でもソースを見ればどういった内容が書かれているかと言うのは理解しやすく、また回路の改良もしやすくなる。

例^[2] : ハーフアダダーの回路について解説していく。ハーフアダダーとは、2進法同士の足し算を行った際に $1 + 1 = 10$ の繰り上げを考えずに $1 + 1 = 0$ という計算結果を出力する演算回路の基礎の一つである。繰り上げを考えるものはフルアダダーとし、 $1 + 1 = 10$ となる。

下の表はハーフアダダーの計算結果が $A \text{ xor } B$ が出力結果で $A \text{ and } B$ は繰り上げの有無を表す。フルアダダーでは $A+B$ の結果を出力する。

A	B	A and B	A xor B	A+B
0	0	0	0	00
0	1	0	1	01
1	0	0	1	01
1	1	1	0	10

ロジック回路で表すと以下ようになる。表の論理式の様に繰り上がりを and, ハーフアダダーを xor で表現できるが、引用した教科書の表現に則り以下のように表現した。

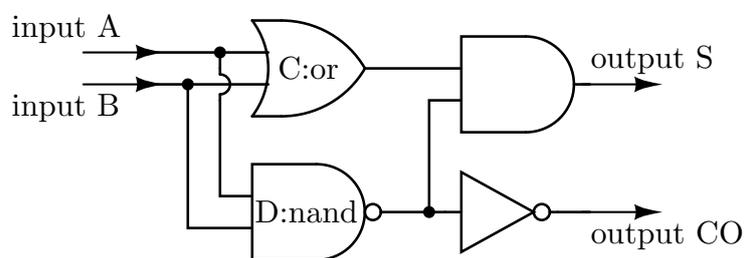


図 2.11: ハーフアダダー回路

これを VHDL で記述すると以下のように書かれる。

```
library IEEE;
use IEEE.std_logic_1164.all;

entity HALF_ADDER is
  port(
    A,B : in std_logic;
    S,co : out std_logic;
  );
end HALF_ADDER

architecture DATAFLOW of HALF_ADDER is
  signal C,D : std_logic;
begin
  C <= A or B;
  D <= A nand B;
  CO <= not D;
  S <= C and D;
end DATAFLOW;
```

ソースを簡単に解説していくと、まずはじめに以下の2行よりライブラリ宣言と使用するパッケージの指定を行う。

```
> library IEEE;
> use IEEE.std_logic_1164.all;
```

次にエンティティ宣言では HALF_ADDER の入出力ポート (A,B,S,co) が入力なのか出力なのか、そしてデータタイプがなんで有るかという事を厳密に定義していく。in,out はそれぞれ入力のみ、出力のみ。std_logic は High or Low(1 or 0) のみを扱える変数として定義している。

```
> entity HALF_ADDER is
>   port(
>     A,B : in  std_logic;
>     S,co : out std_logic;
>   );
> end HALF_ADDER
```

そして HALF_ADDER の DATAFLOW では実際にどんなこういった処理を行うかを記述していく。C,D の変数を定義し、ロジック回路を作っていく。

```
> architecture DATAFLOW of HALF_ADDER is
>   signal C,D : std_logic;
>   begin
>     C <= A or B;
>     D <= A nand B;
>     CO <= not D;
>     S <= C and D;
>   end DATAFLOW;
```

このソースをコンパイルして FPGA に転送することで、入力 A,B に信号 1 or 0 を入力するとハーフアダダの計算結果 S と繰り上がりの有無の結果 co を出力する回路として動く。この FPGA 周りに入力にスイッチ、出力に LED やマイコンと繋げることでユーザの望む処理をさせることが出来る様になる。

第3章 PoGOLino用3chプログラム

3.1 PoGOLino とは

PoGOLino は気球搭載用の硬 X 線・軟 γ 線検出器 PoGOLite の観測環境で大気上空の中性子バックグラウンドを測定するための小型の検出器である。^[13] 今回の研究で直接関係するわけではないが、PoGOLino の FADC ボードとロジック回路を本実験で用いるためここで説明する。

3.2 PoGOLite 気球実験

PoGOLite は気球搭載用の硬 X 線・軟 γ 線偏光検出器である。^[13] 偏光観測はパルサーやブラックホールや活動銀河角やガンマ線バーストなどの高エネルギー現象の研究で重要になってくるが、検出の行われた天体はかに星雲と CygX-1 のみと撮像観測や分光観測や測光観測と違い非常に少ない。また、エネルギー領域もかに星雲は $2.6/5.2keV$ と $200keV$ 以上、CygX-1 は $600keV$ 以上のみと少ない。そこで偏光観測に特化した PoGOLite 観測機 ($25 - 80keV$) による観測を行う計画が進められている (2013 年夏予定)。

PoGOLite の偏光 X 線の検出原理は、偏光している硬 X 線のコンプトン散乱角の特異性を用いる。図 3.1 の様に 61 本のシンチレータをハニカム構造で配置し、偏光している硬 X 線がシンチレータ内でコンプトン散乱を起こすと偏光方向とは垂直な方向に散乱しやすくなる。この性質により硬 X 線のコンプトン散乱の角度分布を調べることで天体の偏光度と偏光方位角を調べることが出来る。

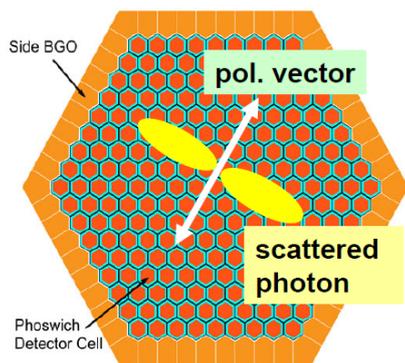


図 3.1: 偏光した硬 X 線の検出原理

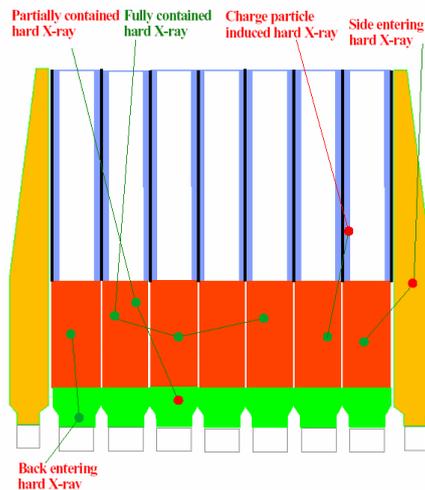


図 3.2: PoGOLite メイン検出部

また、PoGOLite には天体からの硬 X 線・軟 γ 線を検知する主検出部 (PDC) に加え、バックグラウンドとなる視野外からの放射線を検知するアクティブシールド部 (SAS) とそれらを管理する DIO ボードとにわかれている。PDC と SAS はそれぞれ PDC 用、SAS 用のロジック回路の書き込まれた FADC ボードに繋がっており、FADC はルータを介して DIO ボードによる管理を受けている。

このように、PoGOLite 用のロジック回路は複数チャンネルの入力処理が可能であるが、図 3.3 の用に FADC ボードとは別に DIO ボードを用いて制御する必要があるため、ロジック回路の規模が大きくこのままでは本研究用には適さない。

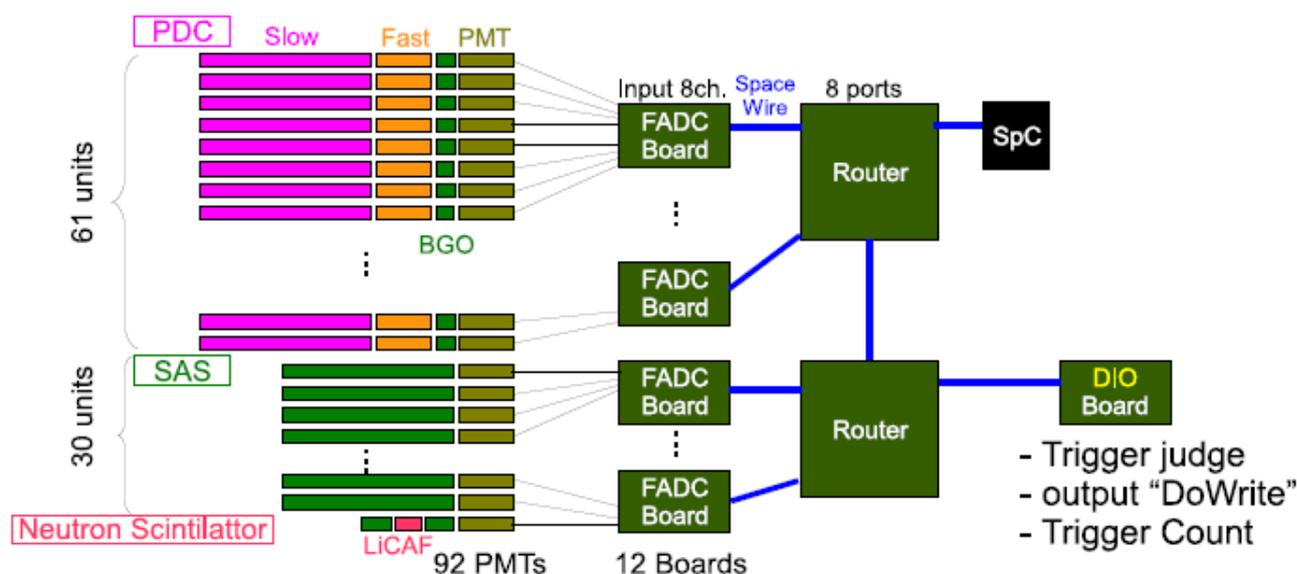


図 3.3: PoGOLite 用の多チャンネル用検出器の構成

3.3 PoGOLino 用検出器

PoGOLino は「BGO+Fast シンチレータ +BGO」を 1 つのセットとして放射線を取得するメインシンチレータ 2 本に「BGO+LiCAF シンチレータ +BGO」のバックグラウンド測定用シンチレータ 1 本の 3 本のシンチレータで観測を行なう。BGO と BGO の間にシンチレータを挟む方法は Phoswich(3.4 節参照) と呼ばれ、シンチレータの立ち上がり速度の違いによりどのシンチレータで反応したかの弁別を行い、1 つの入力信号で複数の信号を取得する。PoGOLino は FADC ボード単体で動かすことができるロジック回路なので、本研究ではこのプログラムを汎用化していく。

3.3.1 FADC ボード

今回用いる FADC ボードは以下の様な物を用いる。このボードは PoGOLite で用いられているデータ処理基板である。シンチレータと MPPC の信号はこの FADC ボードで処理して波形やエネルギーヒストグラムの生成を行う。以下で FADC ボードに備わっている機能について解説していく。

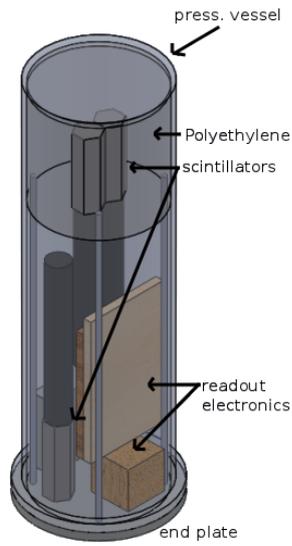


図 3.4: PoGOLino の検出器

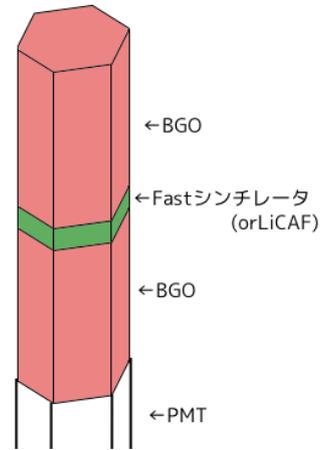


図 3.5: PoGOLino のシンチレータ

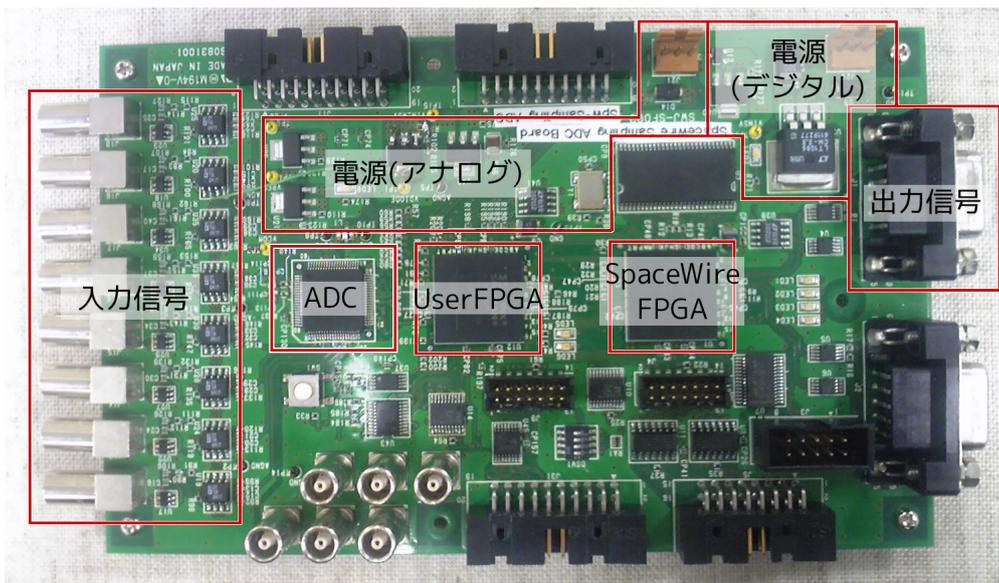


図 3.6: FADC ボード

ADC(ADS5271)

ADC(ADS5271) は 8ch 入力に対応したサンプリング型 Analog-Digital Converter で、サンプリング周期 37.5MHz で 2V までの電圧の高さ (アナログ) を FPGA で読み込めるデジタル信号に 8ch 同時に変換し UserFPGA に出力する。

MPPC とシンチレーション光の読み出し回路から得られた信号は微弱であり最大波高値のピークが短い
ため ADC でうまく読み込めない。そのため、デジタル値に変換する前にプリアンプとシェーピングアンプ
で増幅と最大波高値のピークを長くする。また、入力信号にオフセットを付加しており、信号の入力電圧が
負でも波形が正常に AD 変換できるようになっている。

UserFPGA

XILINX 製の FPGA(XC3S1000_FT256)^[17]。UserFPGA では ADC の取得した信号に対してユーザが自
由にロジック回路を組むことが出来る。今回の研究ではこの UserFPGA を用いてロジック回路を作成する。
FPGA はメモリが揮発性のため電源をオフにすると消えてしまうが、不揮発性のメモリに書き込んでおく
と FPGA 起動時にメモリを読み込むため再度回路を書き込む必要がなくなる。作ったロジック回路につい
ては第 4 章で説明する。

SpaceWireFPGA,SpW Port

SpaceWire は科学衛星に搭載される機器 (観測装置、各種センサ、データ処理装置、記憶装置など) の
装置間で通信を行う際に用いられる通信規格である。SpaceWire 通信用のロジック回路の書き込まれた
SpaceWireFPGA が SpW Port から命令を受け取り、UserFPGA 上のメモリデータを参照・書き換えを
行う。

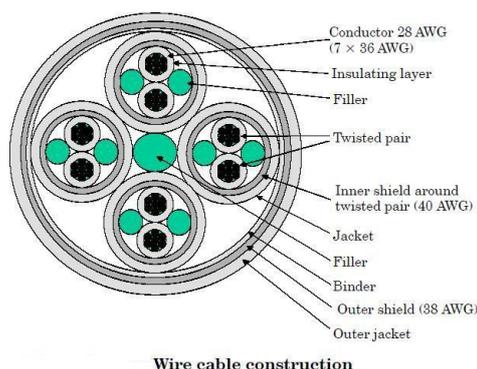


図 3.7: SpaceWire ケーブル

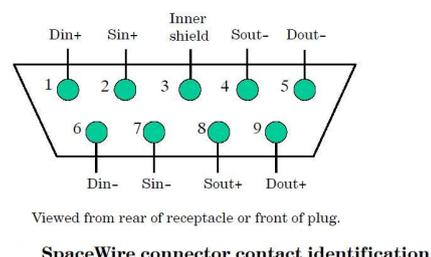


図 3.8: SpaceWire コネクタ

3.4 PoGOLino 用ロジック回路

PoGOLino 用ロジック回路は FADC ボード単体で動かすことができる PoGOLino 用の 3 チャンネルデー
タ処理回路である。入力チャンネルに対して 3 つの波形と 9 つのヒストグラム (3histo/ch) を生成する。ま
た、Phoswich による波形弁別を行い、1 つのチャンネルにつき Fast,Middle,Slow のヒストグラムを生成す
ることが出来る。Threshold などの値や設定を変えるには RMAP 上の特定領域の値を書き換えることでそ
の値を各 Module が読み込み変更することができる。以下ではデータ処理ロジックを解説していく。

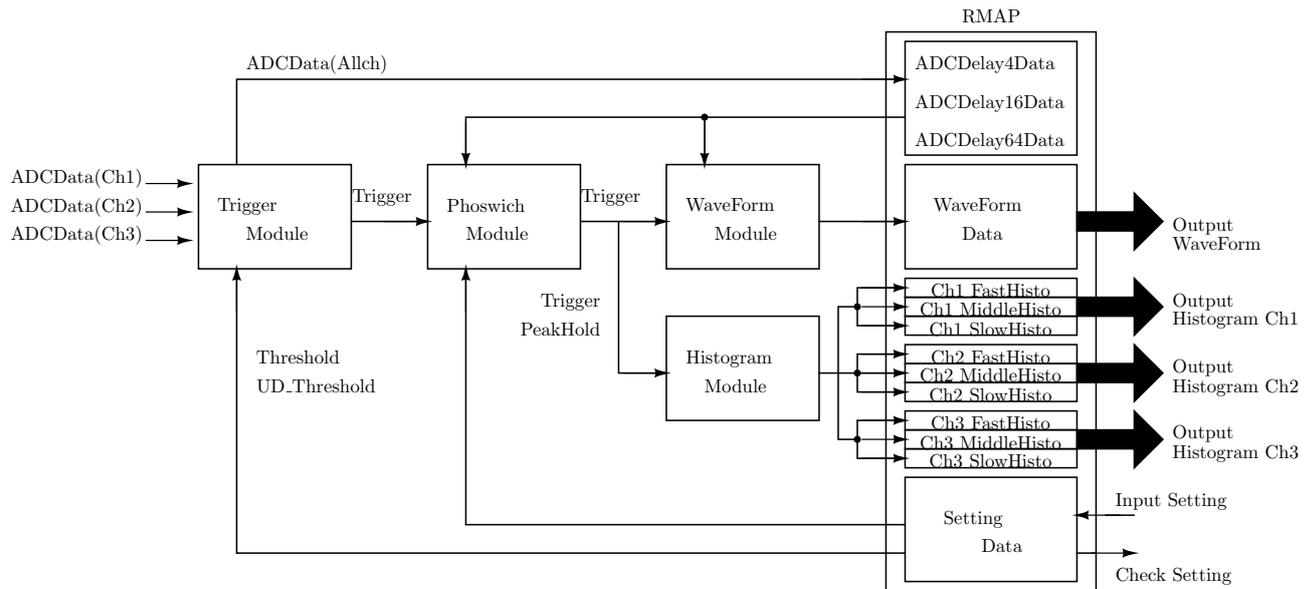


図 3.9: PoGOLino 用プログラムの略図

(<http://www-heaf.hepl.hiroshima-u.ac.jp/gakkai/tenmon0609/tanaka0609.pdf>)

3.4.1 RMAP

^[18]RMAP(Remote Memory Access Protocol) は SpaceWire ネットワークに接続されたノードでのデータの読み書きできる機能。他のノードから FPGA 内のメモリを自身のメモリのように読みだすことが出来るため、RMAP 内に出力を行なうデータや設定データを置き、ノード間でやり取りを行える様にする事で、保存した波形やヒストグラム、Threshold や後述の Phoswich などの設定を変更できる。

3.4.2 TriggerModule

TriggerModule では主に ADC からの入力信号を監視し波形を RMAP に一時的に保存する。ADC の波高が Threshold(閾値) 以上であれば MPPC からの信号が入ってきたとし、PhoswichModule に処理をさせるための信号 Trigger を送る。また、シンチレータが複数の放射線を検出するなどサチュレーションが発生した場合 (UD_Threshold 以上の入力) は UD_Flag を立て、一度波形が Threshold を下回るまで Trigger 出力を停止する。

PhoswichModule

PhoswichModule では立ち上がりの違う波形の弁別を行なう。

Phoswich は Phosphor(蛍光物質) と Sandwich を合わせたもので、シンチレータの立ち上がりの違いから 3 種類のシンチレータからの反応かを検出する。立ち上がり時間はそれぞれ異なり、Fast シンチレータ

が $10ns$ 、BGO が $300ns$ 、LiCAF が $1500ns$ より波形で見ると以下の様に 3 つの異なる波形を見ることが出来る。

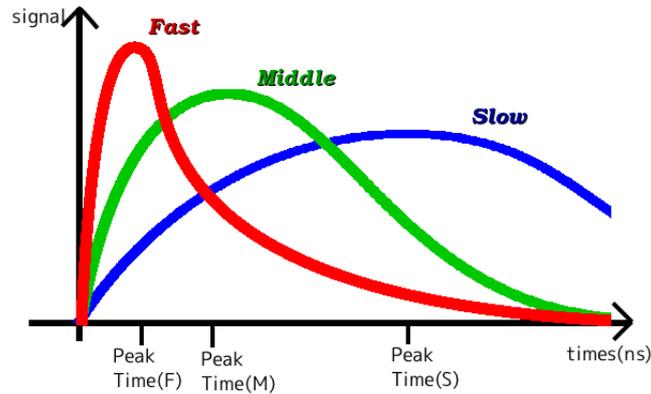


図 3.10: シンチレータの立ち上がりの違い

立ち上がりの違いの判別をプログラム上で行わせるには、入力波形から 3 種類の Delay 波形との差分の最大波高を比較することで弁別することが出来る。

例として BGO(Middle) に X 線が反応した場合を想定すると、図 3.11 の用に 3 種類の $Delay1, Delay2, Delay3$ を作る。各 Delay を引いた値の波形を図 3.12 の色の部分がの差分が取ることが出来る。図 3.13 ~ 図 3.15 に差分波形を示す。Delay1 との差分による波高は赤の領域、Delay2 との差分による波高は赤+緑、Delay3 はとの差分による波高は赤+緑+黄色の領域の波形に成り、これは図 3.11 の色の場所に対応する。

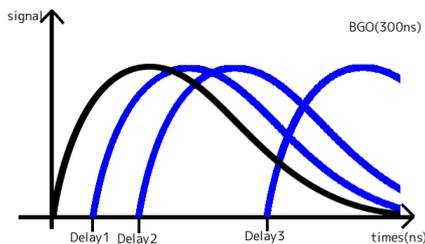


図 3.11: BGO の波形と各ディレイ

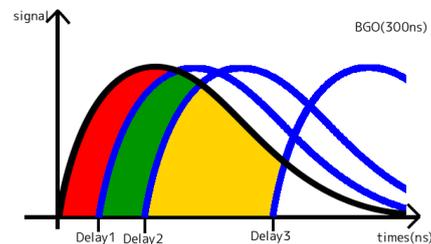


図 3.12: ディレイの差分

この 3 種類の差分の最大波高は図 3.14 より Delay2, Delay3 の最高波高 (PeakHold2, PeakHold3) が同じに値になり、Delay1 の最高波高 (PeakHold1) は他のものよりも低くなる。Fast, slow も同様にして各 Delay1, 2, 3 に対する PeakHold との関係を図 3.13、図 3.15 のようにして求めると各 PeakHold の値はそれぞれ以下の関係を持つ。

$$\left\{ \begin{array}{l} \text{Fast} : \text{PeakHold1} \simeq \text{PeakHold2} \simeq \text{PeakHold3} \\ \text{Middle} : \text{PeakHold1} < \text{PeakHold2} \simeq \text{PeakHold3} \\ \text{Slow} : \text{PeakHold1} < \text{PeakHold2} < \text{PeakHold3} \end{array} \right. \quad (3.1)$$

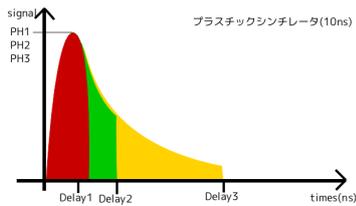


図 3.13: Fast 信号の差分

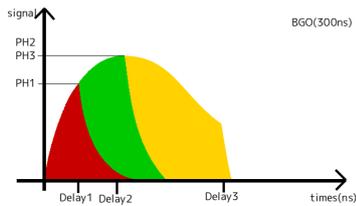


図 3.14: Middle 信号の差分

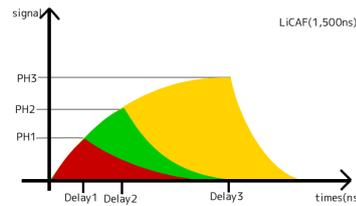


図 3.15: Slow 信号の差分

プログラム上での処理は以下の様に行う。ここで、例えば PeakHold2 と PeakHold1 の比を決めるのは A, ShiftA である。

```

if (PeakHold2(11 downto 0) < A * PeakHold1(11 downto shiftA) ) then Fast
elsif (PeakHold3(11 downto 0) < B * PeakHold2(11 downto shiftB) ) then Middle
else Slow

```

ロジック回路では割り算をすることが出来ないが、ビットシフトを行う事によって2の倍数で割ることが出来るので、A と ShiftA を組み合わせて任意の少数を表す。図 3.16 を参考にすると、最高波形が PeakHold= 2955 とする。ロジック上では整数を2進数で扱うため、PeakHold=2955(10進数) は 0x101110001011(2進数) と表記される。2進数の1桁目の数をなくした整数 (0x10111000101) は10進数だと1477と元の数の1/2(端数切り捨て)になる。更に、元の数から1,2桁目の2つを取り除いた 0x1011100010 という値は、元の PeakHold から $1/4 (= \frac{1}{2}^2)$ になる。このようにビットシフトを行なう毎に PeakHold の値は1/2倍される(端数切り捨て)。よって、各右辺の PeakHold の値はそれぞれ $(A / 2^{shiftA}), (B / 2^{shiftB})$ 倍している事になり、より細かい倍率で調整出来る。

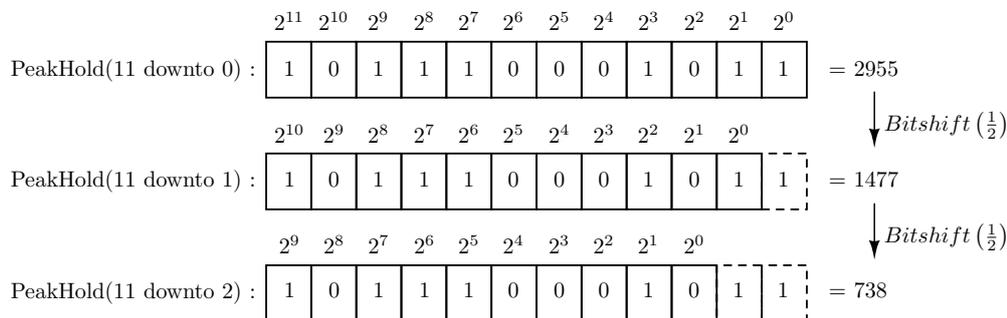


図 3.16: ビットシフト

3.4.3 SaveWaveModule

SaveWaveModule は波形を保存するモジュールである。TriggerModule から保存する波形のチャンネルを受け取ると、1つの波形につきヘッダ 10byte と波高 100byte の計 110byte1 セットで保存される。ヘッダには波形保存の管理番号である EventNumber(3byte), 保存を行なうチャンネルの Save_Ch(1byte)、保存中のチャンネルの Working_Ch(1byte)、波形を取得した時間 NowTime(4byte) が保存されており、WaveHeightData には 1clock あたり 2byte の波高値が 50clock 分 (100Byte) が保存される。

保存形式は図 3.17 で示す。

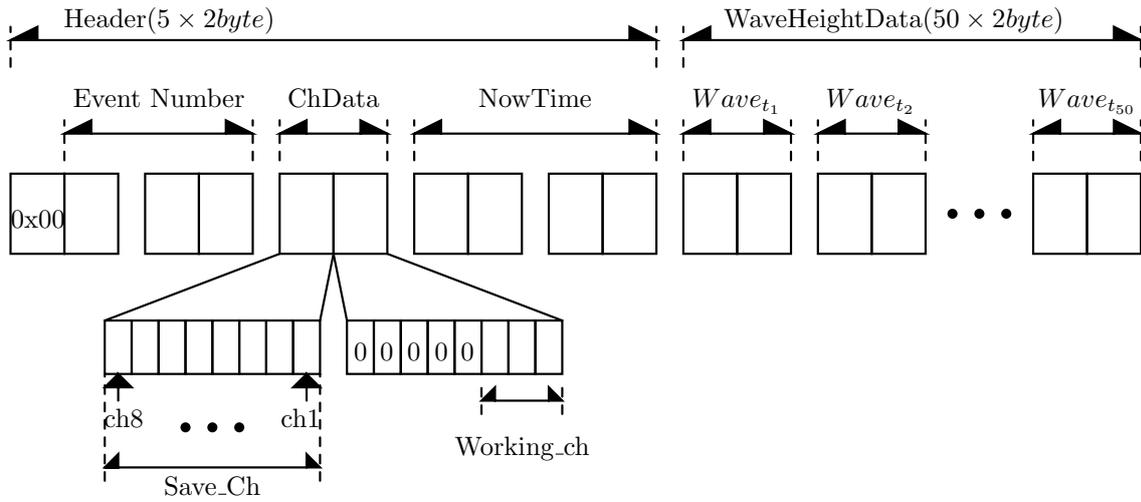


図 3.17: 1 波形の保存構成

EventNumber は PhoswichModule から Trigger を受け取った回数、NowTime は検出器の起動から Trigger を受け取るまでの時間を出力する。Save_Ch は同時入力あったチャンネルの Trigger の有無を 1 チャンネルにつき 1bit の 1 or 0 で保存し、その中で保存を行なう波形のチャンネルを Working_Ch($2^3 = 8$) で表す。ch1,3,7 の Trigger を同時に取得した場合は、まずは Save_Ch に Trigger 情報を式 (3.2) のように保存を行なう。

$$\begin{aligned}
 \text{Save_Ch} &= \{ \text{ch8} \ \text{ch7} \ \text{ch6} \ \text{ch5} \ \text{ch4} \ \text{ch3} \ \text{ch2} \ \text{ch1} \} \\
 &= \{ \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \} \leftarrow (\text{bit 値})
 \end{aligned}
 \tag{3.2}$$

Working_Ch は Save_Ch よりチャンネル数の低い順から参照し、今回の例では ch1 なので Working_ch=1 として ch1 の波形を保存する。ch1 の保存が終わると次の波形 ch3, ch7 の波形を順に保存を行なう。

波形の保存時間は 50clock 固定で、Trigger が立つ前の 5clock、立った後の 45clock を保存する。そのため、そのまま保存を行なうと Threshold を超えてから $1200ns (= 26.7ns \times 45Clock)$ までの入力波形しか保存できず、立ち上がり時間が $1500ns$ の LiCAF の波形はピークに達する前で波形出力が終わってしまう (図 3.18 左)。その場合は取得する波高値の取得を間引くことでピークを全て取得できるようになる。取得するタイミングを数個飛ばして保存することで 1clock あたりの時間が長くなるため、ピークを含めた波形を保存することが出来るようになる (図 3.18 右)。間引く事によって波形の時間分解能は悪くなるが、立ち上がりの遅いシンチレータによる波形は緩やかな変化なので問題ない。

波形の保存は一つの波形につき約 $4.3\mu s$ かかり、同時入力があった場合は信号の数だけ保存時間が増える。追加信号全ての波形を保存し終わるまでは波形を取得できないため、この間の時間はデッドタイムになる。

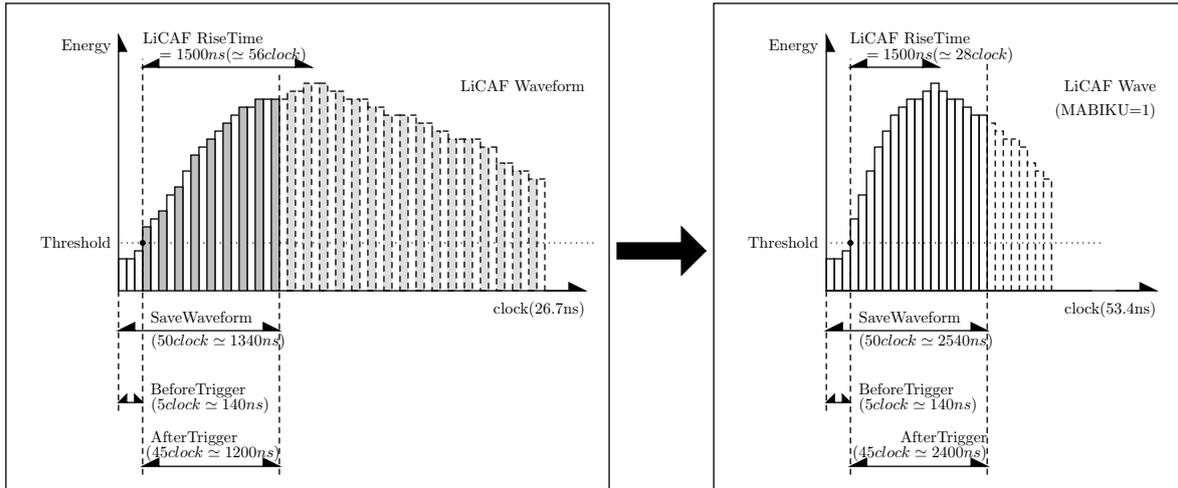


図 3.18: 波形保存を間引く

3.4.4 HistogramModule

TriggerModule から Trigger と PeakHold(最高波形) と Phoswich 弁別結果を受け取る毎に 1 カウントとしてヒストグラムを生成していく。チャンネル 1 つに対して Phoswich により弁別した 3 つの Fast, Middle, Slow のヒストグラムを 3ch 生成する。エネルギー分解能は 9bit(512[digital 値]) でカウント数は各パルスハイトチャンネルあたり 16bit(65535count) まで行える。

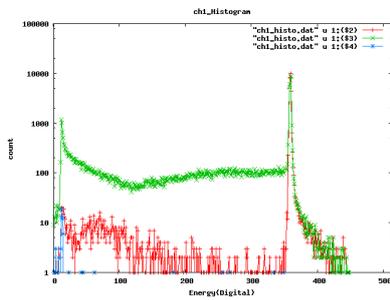


図 3.19: ch1 ヒストグラム例

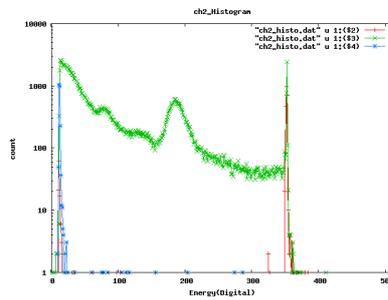


図 3.20: ch2 ヒストグラム例

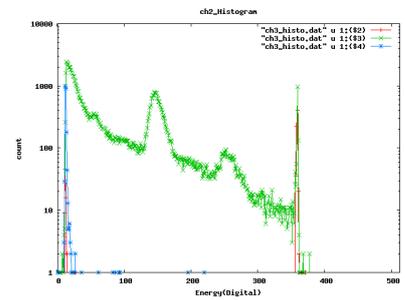


図 3.21: ch3 ヒストグラム例

第4章 8ch 読み出しロジック回路の開発

今回開発した 8ch 読み出しロジック回路は、入力信号の波形を保存して 8ch それぞれのエネルギーヒストグラムを生成することが出来る。また、コインシデンスの設定を変更することにより、ミュオン検出器やコンプトンカメラとしても扱えるようになる。PoGOLino 用のロジック回路をベースに作り替えているので、基本的なロジックについては第 3 章を参照。

4.1 8ch 読み出しロジック回路の構成

基本は PoGOLino 用のロジック回路と同様で、TriggerModule に波形を入力し Threshold 以上の入力があった場合は波形保存とヒストグラム生成を行う。大きな変更点は以下のようになっている。

- CoincidenceModule の追加
- Phoswich 機能縮小 (TriggerModule と統合)
- 8ch 入力可能
- 8ch 波形保存
- 8histo 生成 (1histo/ch)

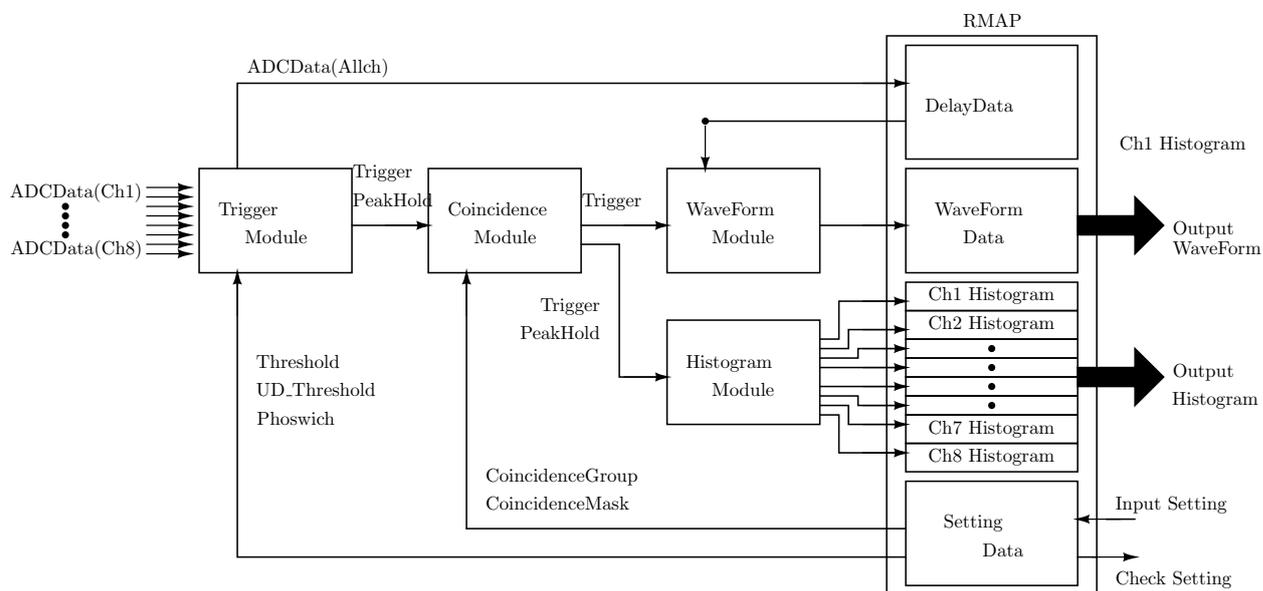


図 4.1: 8ch 読み出しロジック回路の構成ダイアグラム

4.1.1 TriggerModule

TriggerModule では 8ch の ADC を管理し、Threshold を超えた波形が入力された時に簡易的な Phoswich 処理を行い CoincidenceModule に PeakHold と各チャンネルのトリガーを出力する。FADC の乗算には多くのメモリを使うため乗算可能な数に制限があることと、8ch で 3 種類の Delay データを保存するだけのブロック RAM が余っていないため、8ch 用のロジックでは Phoswich 機能をそのまま使うことが出来なかった。その為、Phoswich は乗算を使わないビットシフトのみの弁別 2 種類とし、弁別数も Fast or Slow のどちらかのみを取得する様にした。

```
if (PeakHold2(11 downto 0) > PeakHold1(11 downto shiftA) ) then Fast
else Slow
```

4.1.2 コインシデンス設定

シングルチャンネルから多チャンネルになると、複数信号での同時取得 (コインシデンス) 時の挙動を考える必要がある。

コインシデンスの設定を 2 チャンネルで考えてみると、図 TriggerModule から出力される Trigger は [ch1] のみの信号、[ch2] のみの信号に加えて [ch1 と ch2] 同時の信号の 3 パターンがあり、使用状況に応じてその 3 パターンのうちどれかに制限を行う必要がある。例えば、コンプトンカメラの様に 2 つのシンチレータの同時反応が重要となる検出器では、[ch1] のみや [ch2] のみの Trigger では波形やヒストグラムを取得せず [ch1 と ch2] 同時の Trigger が出力された時のみ保存を行わせる必要がある。

チャンネル数と Trigger 出力パターン数の関係は、チャンネル数 3 の場合 [ch1] [ch2] [ch3] の 1 つの Trigger 出力、[ch1,ch2] [ch1,ch3] [ch2,ch3] の 2 チャンネル同時 Trigger 出力、[ch1,ch2,ch3] の全チャンネル Trigger 出力の計 7 パターンに増える。設定数はチャンネル数が増えるに従い $(2^{\text{チャンネル数}-1})$ と累乗スケールで増えていき、図 4.3 では 8ch 用いるので、 $255 (= 2^8 - 1)$ もある全 Trigger 出力パターンに設定を行う必要がある。

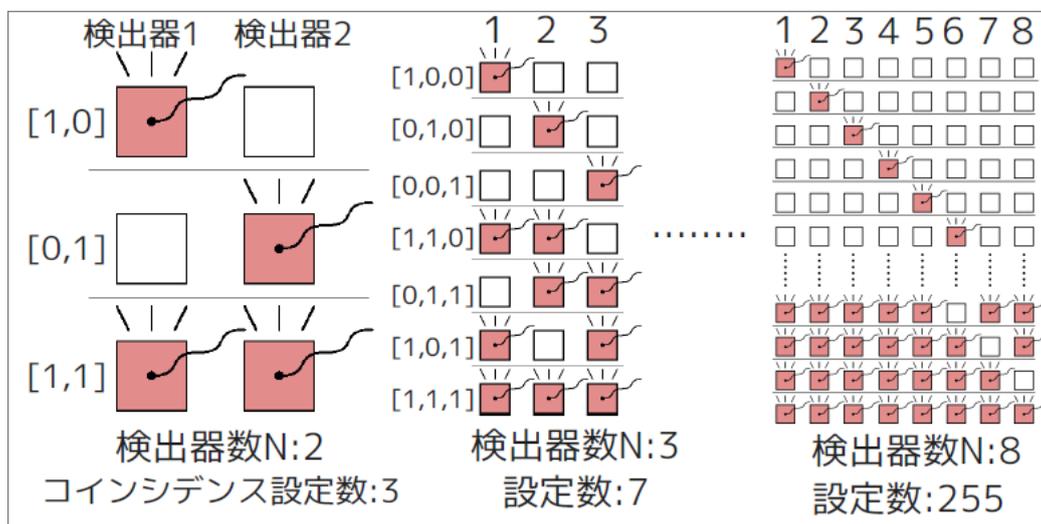


図 4.2: コインシデンスの条件数

例として図 4.3 の様に複数の細長いミュオン検出器を配置する事によりミュオンの入射位置を特定できる方法について考える。8 つの細長いミュオン検出器を上層に 4 つ (ch1-ch4) と下層に 4 つ (ch5-ch8) が交差するように配置すると、ミュオンが入射した場合に上層と下層の検出器が同時に反応する。反応した検出器の番号、つまり検出器の位置により上層で X 軸, 下層で Y 軸が分かるのでミュオンの入射位置を 1-16 のいずれかに特定することが出来る。図 4.3 の例では ch2 と ch6 が同時に反応しているので、図 4.3 右の 6 の位置に入射してきた事を特定することが出来る。この入射位置を特定するには各上層と下層の同時反応パターン 16 個を取得する条件として入力する必要があり、それ以外の反応では位置を特定することができないため除外する必要がある。

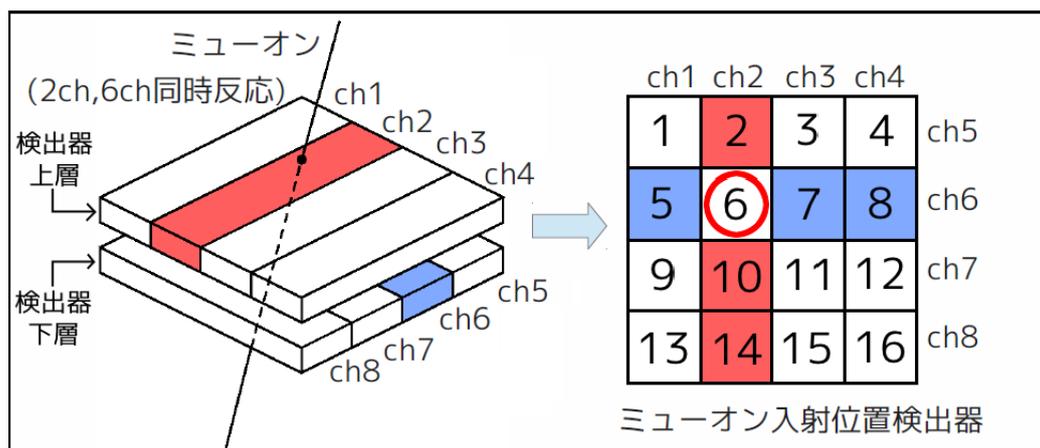


図 4.3: ミュオン入射位置検出器

ここで、16 個の入射位置を特定するには取得する条件である表 4.1 の同時イベント入力を全て入力する必要がある。

特定位置	反応グループ	同時イベント入力
1	ch1 & ch5	[1,0,0,0,1,0,0,0]
2	ch2 & ch5	[0,1,0,0,1,0,0,0]
3	ch3 & ch5	[0,0,1,0,1,0,0,0]
4	ch4 & ch5	[0,0,0,1,1,0,0,0]
5	ch1 & ch6	[1,0,0,0,0,1,0,0]
6	ch2 & ch6	[0,1,0,0,0,1,0,0]
7	ch3 & ch6	[0,0,1,0,0,1,0,0]
8	ch4 & ch6	[0,0,0,1,0,1,0,0]
9	ch1 & ch7	[1,0,0,0,0,0,1,0]
10	ch2 & ch7	[0,1,0,0,0,0,1,0]
11	ch3 & ch7	[0,0,1,0,0,0,1,0]
12	ch4 & ch7	[0,0,0,1,0,0,1,0]
13	ch1 & ch8	[1,0,0,0,0,0,0,1]
14	ch2 & ch8	[0,1,0,0,0,0,0,1]
15	ch3 & ch8	[0,0,1,0,0,0,0,1]
16	ch4 & ch8	[0,0,0,1,0,0,0,1]

表 4.1: ミュオン入射位置検出器のコインシデンス設定 (16 パターン入力)

この表 4.1 の全ての条件に対して設定を行うのが確実であり、PC 上でコントロールするなら良いが、今回の研究では携帯型放射線検出器用のロジック回路の開発により、ちまちま 1 つ 1 つイベントを入力するのは適していない。そこで、携帯機でも少ない設定数でコインシデンス設定を簡単に行えるものも実装した。簡易コインシデンスの設定方法についてこれから解説していく。

今回の例では、16 個の位置を特定するためには結局の所上層部と下層部の両方が反応したものを取得すれば位置を特定することが出来る。そのため、論理条件で表現するとこの取得条件は以下の様になる (上層部, 下層部を以降それぞれ A,B と表記する)。

$$\begin{aligned} & \text{上層部 (A)} \quad \text{and} \quad \text{下層部 (B)} \\ & (\text{ch1 or ch2 or ch3 or ch4}) \quad \text{and} \quad (\text{ch5 or ch6 or ch7 or ch8}) \end{aligned}$$

この論理条件を用いるために、今回用いたコインシデンス方法では、CoincidenceGroup と CoincidenceMask の二通りを設定することで設定することが出来る。今回のミュオン入射位置検出器では以下の値を入力することで A and B とすることができる。

- CoincidenceGroup = {AAAABBBB}
- CoincidenceMask = {01000000}

各解説を行なっていく。まずは各チャンネルのグループ分けを行う CoincidenceGroup の設定を行う。CoincidenceGroup 設定では、各チャンネルがどのグループに所属するかを決める。今回の場合は ch1-ch4 を A, ch5-ch8 を B とする為、表 4.2 のように CoincidenceGroup={AAAABBBB} と入力する。

	ch1	ch2	ch3	ch4	ch5	ch6	ch7	ch8
CoincidenceGroup	A	A	A	A	B	B	B	B

表 4.2: 検出器のグループ分け

次に、グループの論理条件を取るために、CoincidenceMask の設定を行う。CoincidenceMask は、グループの反応している数 (以下グループフラグ数 (GroupFlagCounter:GFC)) によって and,or,xor を取ることができる。A と B の入力の有無と GFC の関係は表 4.5 より、A,B 同時に反応したものは GFC=2 となり、A もしくは B のどちらか片方だけの反応の場合は GFC=1 となる。同時反応したものは A and B で、それ以外が A xor B となり、その両方が A or B という論理条件となる。

		A	
		1	0
B	1	GFC=2	GFC=1
	0	GFC=1	-

表 4.3: A,B の GFC 値

論理	GFC=1	GFC=2
A and B	0	1
A or B	1	1
A xor B	1	0

表 4.4: CoincidenceMask と論理条件

そこで、逆に GFC=2 の時のみ出力を行う様にすれば A and B という条件に、GFC=1 の時のみ出力を行う条件にすれば A xor B、両方 (GFC=1,2) の場合は A or B として条件を設定することが出来る。その

ため、CoincidenceMask の設定を表 4.6 の様に設定することでチャンネル 1 とチャンネル 2 の and,or,xor の 3 つの取得条件を設定できる。

ch1	ch2	ch3	GFC
1	0	0	GFC=1
0	1	0	
0	0	1	
1	1	0	GFC=2
1	0	1	
0	1	1	
1	1	1	GFC=3

表 4.5: A,B,C の GFC 値

論理	GFC=1	GFC=2	GFC=3
A and B and C	0	0	1
A or B or C	1	1	1
A xor B xor C	1	0	0
not(A and B and C)	1	1	0

表 4.6: CoincidenceMask と論理条件 3ch

今回用いるのは A and B の条件なので、GFC=2 の時に Mask が 1(出力許可)を行うため、CoincidenceMask は以下の値を代入すると良い。A,B の 2 つの場合は GFC は 2 までなので CoincidenceMask={01} で良いが、今回作ったプログラムは 8ch 全てを別のグループに所属した際に最大 A-H の 8 グループを設定することが出来るため、あらかじめ GFC=8 まで設定出来るようになっている。そのため、入力する際は GFC=3 から 8 まで入力する必要があるが、プログラムを動かす上では 0 でも 1 でも変動はない。

	GFC=1	GFC=2	GFC=3	GFC=4	GFC=5	GFC=6	GFC=7	GFC=8
CoincidenceMask	0	1	0	0	0	0	0	0

表 4.7: 検出器のグループ分け

したがって、実際に打ち込む量は CoincidenceGroup の 8 文字と CoincidenceMask の 8 文字 (8ch の場合) の 16 文字となり、通常的全パターン入力に比べて削減することが出来る。

今回の例ではアクティブなグループが A,B の 2 つのグループしか無いが、例えば A,B,C の 3 グループの場合は A and B and C は GFC=3 の時のみ、A xor B xor C の場合は GFC=1 の時のみ取得を行うようにすれば条件づけることが出来る。not(A and B and C) の場合は GFC=1,2 の時に取得を行う。

コインシデンスのアルゴリズムと例を図 4.4 に示す。設定は例と同じく以下の設定とし、ch2 と ch6 の同時反応があったと仮定する。

- CoincidenceGroup = {AAAABBBB}
- CoincidenceMask = {01000000}

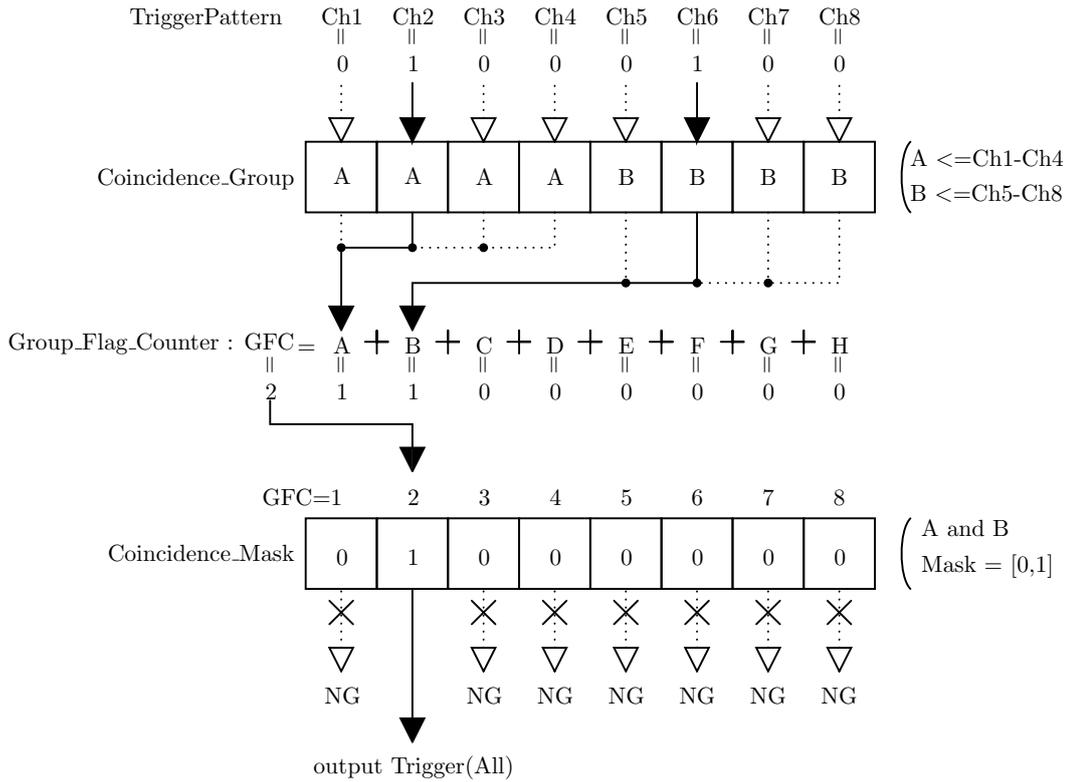


図 4.4: コインシデンスアルゴリズム (例)

4.1.3 WaveFormModule : 波形保存

CoincidenceModule から取得トリガーを受け取ると処理を開始する。8ch までの保存を行うことができる。ch7,ch8 の同時取得した波形を PC に送り表示した物を図 4.5 に示す。

4.1.4 HistogramModule : エネルギーヒストグラム生成

8ch のヒストグラムを 1ch につき 1histo の生成を行う。同時取得した例を図 4.6 に示す。このヒストグラムは FADC の中で生成したものを PC へ転送し表示したものである。

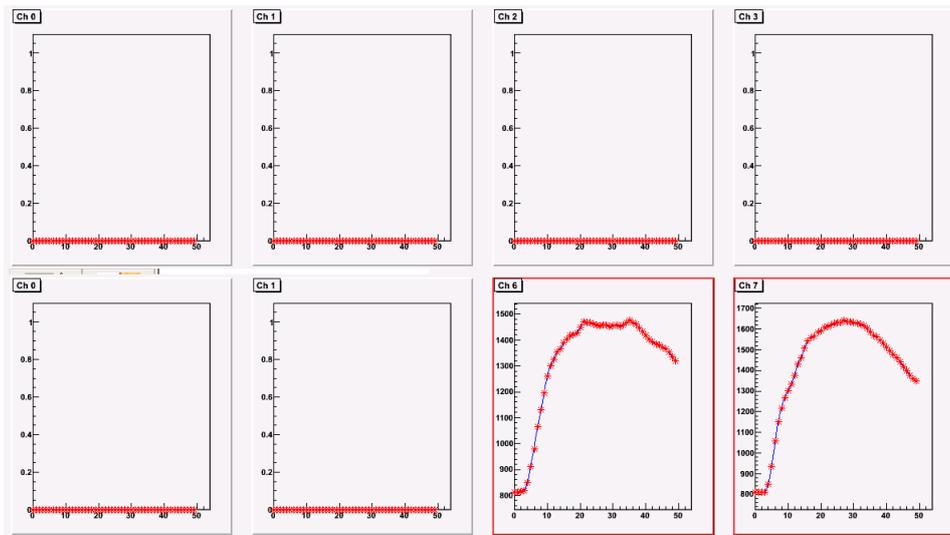


図 4.5: 波形取得プログラム

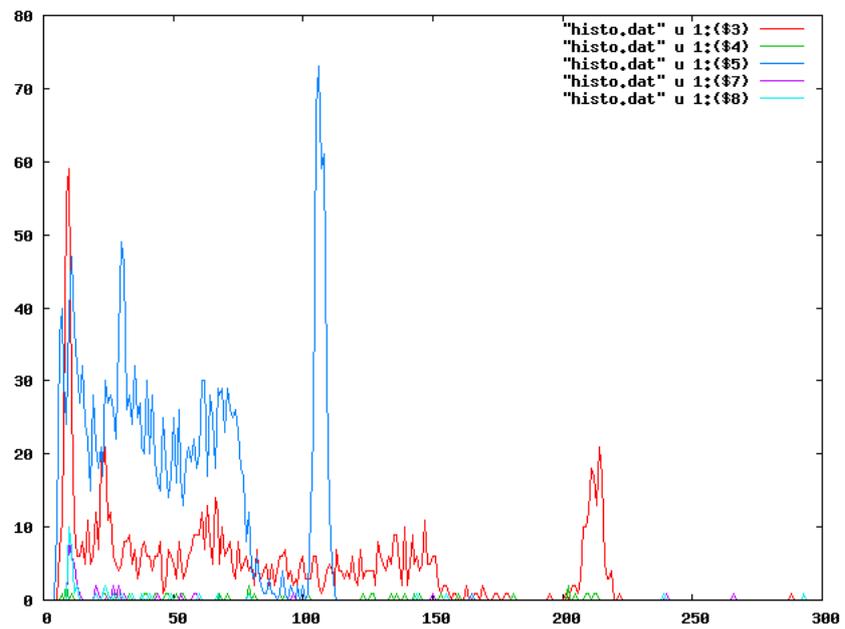


図 4.6: ヒストグラム

第5章 放射線検出実験

この章では第4章で製作したロジック回路を用いて実際に放射線検出実験を行う。

5.1 実験装置

実験装置は以下の図 5.1 の構成で行った。

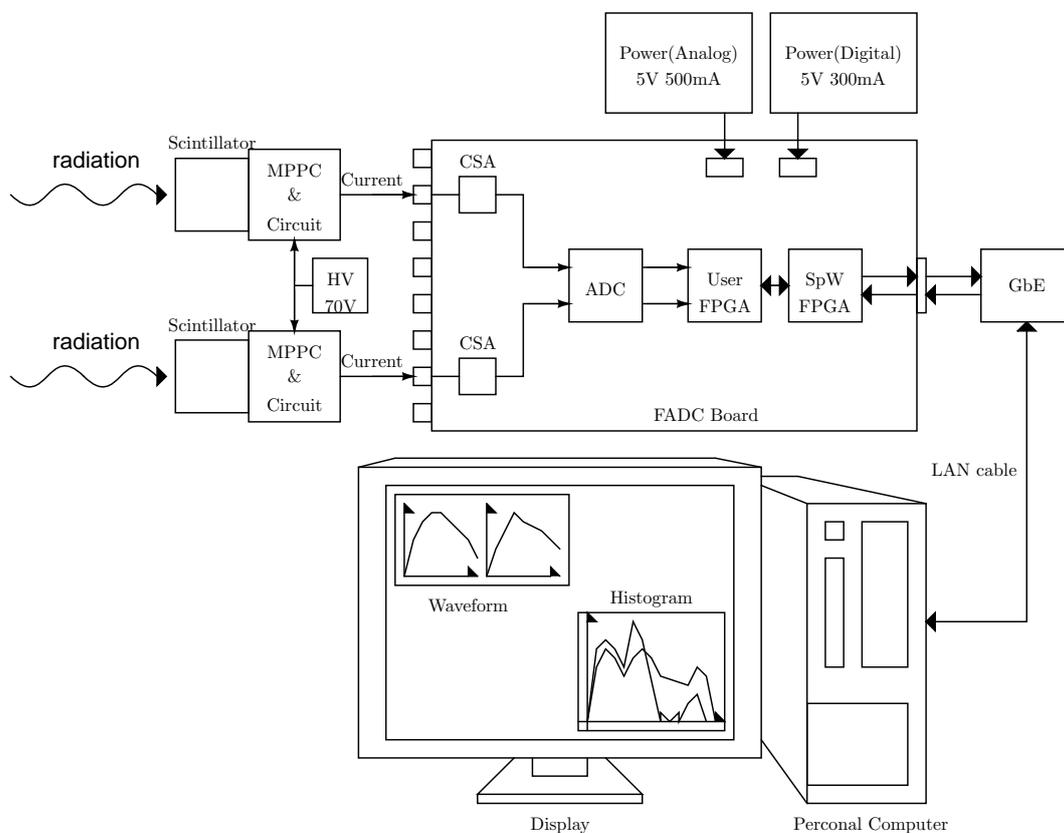


図 5.1: 実験装置の略図

放射線の取得から波形やヒストグラム生成までの流れは以下の様になる。

- ・ 放射線をシンチレータで可視光に変換する。
- ・ MPPC で可視光を電流に変換する。
- ・ FADC ボードで波形保存とヒストグラム生成を行う

- ・ SpW 通信, GbE により FADC と PC とでデータのやり取りを行う
- ・ 取得した波形データとヒストグラムデータをディスプレイに表示

以下では各装置の詳しい説明を行なっていく。

5.1.1 シンチレータ (Scintillator)

シンチレータは X 線や γ 線に反応してシンチレーション光 (可視光) を発する物質である。^[7] 入射 γ 線のエネルギーとシンチレーション光の光子数はほぼ比例関係を持つため、シンチレーション光の光子数を数える事ができれば逆算して γ 線のエネルギーを知ることが出来る。シンチレータの種類には大きく分けて有機物, 無機結晶の固体のものや液体, 気体のものがあり、またシンチレータの材質によって光量や光量がピークに達する立ち上がりの時定数や放射線の種類による反応率や温度による発光量などが変わってくるため、環境や目的に応じて変えていくと良い。

以下では一例として固体シンチレータの比較を載せる。

シンチレータ	密度 (g/cm^3)	屈折率	相対光量	蛍光波長 (nm)	時定数 (ns)
NaI(Tl)	3.67	1.85	100	415	230
CsI(Tl)	4.51	1.79	45	565	1000
CsI(Na)	4.51	1.84	85	420	630
BGO	7.13	2.15	12	480	300

表 5.1: シンチレータスペック

今回の実験では密度が高くガンマ線の光電吸収率の高い BGO を用いた。シンチレータに入射したガンマ線は光電吸収, コンプトン散乱, 貫通 (無反応) の 3 つに分かれる。光電吸収は γ 線の全エネルギーが光子となり、コンプトン散乱は全エネルギーの一部を受け取り、貫通した際は何も起こらず通り抜けて行く。その反応確率: P はシンチレータ密度: ρ , シンチレータの厚さ: d , 反応断面積: σ に依存する。

$$P = 1 - e^{-\rho d \sigma} \dots \dots \left\{ \begin{array}{l} \rho : \text{シンチレータ密度 } (g/cm^3) \\ d : \text{シンチレータの厚さ } (cm) \\ \sigma : \text{反応断面積 } (cm^2/g) \end{array} \right. \quad (5.1)$$

BGO に ^{22}Na の $511keV$ のガンマ線が入射した際の反応確率は以下の様になる。BGO の光電吸収とコンプトン散乱の反応断面積に関しては XCOM (<http://physics.nist.gov/PhysRefData/Xcom/html/xcom1.html>) よりエネルギーごとのコンプトン散乱, 吸収の反応断面積があるので $500keV$ の物を使用した。

$$\left\{ \begin{array}{l} P_{scat} = 1 - e^{-7.13 \times 1 \times 0.07206} = 40.2[\%] \quad \dots \text{コンプトン散乱確率} \\ P_{Abs} = 1 - e^{-7.13 \times 1 \times 0.05852} = 34.1[\%] \quad \dots \text{光電吸収確率} \\ P_{pass} = 1 - (P_{scat} + P_{Abs}) = 25.7[\%] \quad \dots \text{無反応確率} \end{array} \right. \quad (5.2)$$

その他、実験で用いた各確率を表にまとめた。

γ 線 [keV]	断面積 (散乱)[cm^2/g]	断面積 (吸収)[cm^2/g]	コンプトン散乱 [%]	光電吸収 [%]	貫通 [%]	備考
500	7.206×10^{-2}	5.852×10^{-2}	40.2	34.1	25.7	$^{22}Na(511keV)$ $^{137}Cs(661keV)$ の 45 度散乱(479keV)
1250	4.767×10^{-2}	8.293×10^{-3}	28.8	5.7	65.5	$^{22}Na(1274keV)$
600	6.693×10^{-2}	3.833×10^{-2}	37.9	23.9	38.2	$^{137}Cs(661keV)$
300	8.659×10^{-2}	2.073×10^{-1}	35.3	64.7	0	$^{137}Cs(661keV)$ の 90 度散乱(288keV)

厚さ 0.7cm のところで散乱と吸収の合計が 100[%] になったのでその値を用いた

表 5.2: γ 線反応率 (シンチレータ 1cm)

5.1.2 MPPC& 回路 (circuit)

MPPC は半導体光センサーで、シンチレーション光を検知して光電効果により電流を流す半導体の光検出器である。シンチレータのシンチレーション光が MPPC に欠損無く入るようにオプティカルグリッドで密着させ、シンチレーション光以外が MPPC に反応しないようにゴアテックスと呼ばれる伸縮する白いテープでシンチレータと MPPC を巻きつけ (図 5.4)、黒い布をかぶせることでノイズを軽減し放射線からの信号を取得できるようになる。

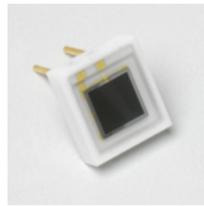


図 5.2: MPPC(S10362-33-050C)

実験で用いた MPPC は受光面積が $3 \times 3mm$ の浜松ホトニクス (<http://jp.hamamatsu.com/>) の S10362-33-050C を用いた。MPPC は受光面積が同じでも検出量が APD のピクセルサイズによって変わってしまう。APD はマルチアレイ状になっており APD と APD の間の隙間の検出できない不感領域が存在する為、細かい APD を敷き詰めると開口率が下がり検出量も下がる。その代わりに APD のピクセル数が多くなるほど高エネルギーの検出の精度が向上するため、取得効率を取るか精度を取るかによってピクセルサイズを変更する必要がある。S10362-33 系列のスペックを以下表 5.3 に表示する。

MPPC から取得した信号は以下の図 5.3(実物図 5.5) の回路を用いて FADC ボードに入力する事が出来るようになる。

5.1.3 SpaceWire to Gigabit Ether(GbE)

SpaceWire to Gigabit Ether(以下 GbE) は FADC ボードと PC とのデータのやり取りを行うための通信機器。図 5.6 の前面で PC と LAN ケーブルをつなぎ、背面にある SpaceWire ケーブル (第 3.3.1 節) で FADC ボードとつなぐことで PC と FADC 間でデータの送受信を行うことが出来るようになる。

項目	記号	S10362-33			単位
		-025C	-050C	-100C	
開口率 *1	-	30.8	61.5	78.5	%
感度波長範囲	λ	320-900			nm
最大感度波長	λ_p	440			nm
動作電圧範囲	-	$70 \pm 10^{*2}$			V
ダークカウント *3	Typ.	4	6	8	<i>Mcps</i>
	Max.	8	10	12	<i>Mcps</i>
端子間容量	<i>Ct</i>	320			<i>pF</i>
時間分解能 (FWHM)*4	-	500-600			<i>ps</i>
逆電圧の温度計数	-	56			<i>mV°C</i>
増倍率	M	2.75×10^5	7.5×10^5	2.4×10^6	-

*1:1 ピクセル中で受光部の占める割合

*2:個別製品の推奨動作電圧については製品データを参照

*3:0.5p.e.(閾値レベル)

*4:シングルフォトンレベル

表 5.3: 半導体 S10362-33 系列のスペック

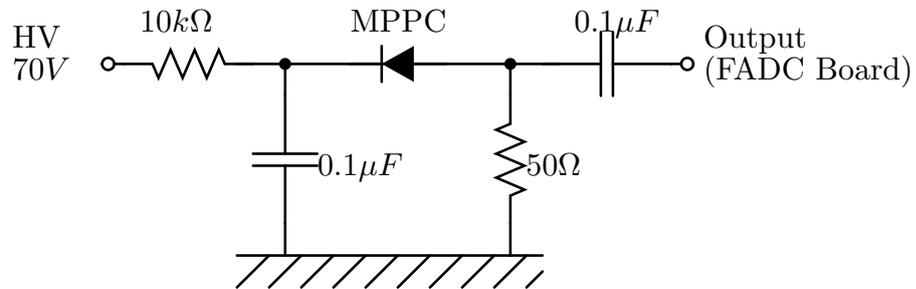


図 5.3: MPPC によるシンチレーション光の読み出し回路



図 5.4: シンチレータと MPPC



図 5.5: 読み出し回路 (左に HV, 右へ出力)



図 5.6: SpaceWire to Gigabit Ether(GbE)

5.2 実験 1 : ^{22}Na の対消滅 γ 線の取得

^{22}Na 線源は半減期 2.6 年で β^+ 崩壊を起こし、陽電子と 2+励起状態 ^{22}Ne に崩壊する。陽電子は対消滅を起こし 511keV の γ 線を、 ^{22}Ne は γ 崩壊を起こし 1274keV の γ 線を放出する。対消滅で発生した 511keV の γ 線は 2 本の γ 線が真逆の方向に同時に発生するため、検出器 2 つの中心に ^{22}Na 線源を設置すると同時に検出器が反応すると考えられる。そこで、同時反応した時のみのエネルギーヒストグラムを生成すれば、 511keV の信号のみを取得し、 1274keV は取得しないはずである。

以上の実験より多チャンネル入力ロジック回路のテストを行った。

5.2.1 実験 1:実験環境

^{22}Na 線源 91.5kBq (式 (2.4) より) を挟む様に等間隔で BGO を配置する。MPPC は外部の光に反応しない様に黒幕をかけ暗室にする。MPPC からの信号は読み出し回路を通じて FADC ボードに入力する。BGO の性能は温度によって変化するので 20°C 固定で行った。

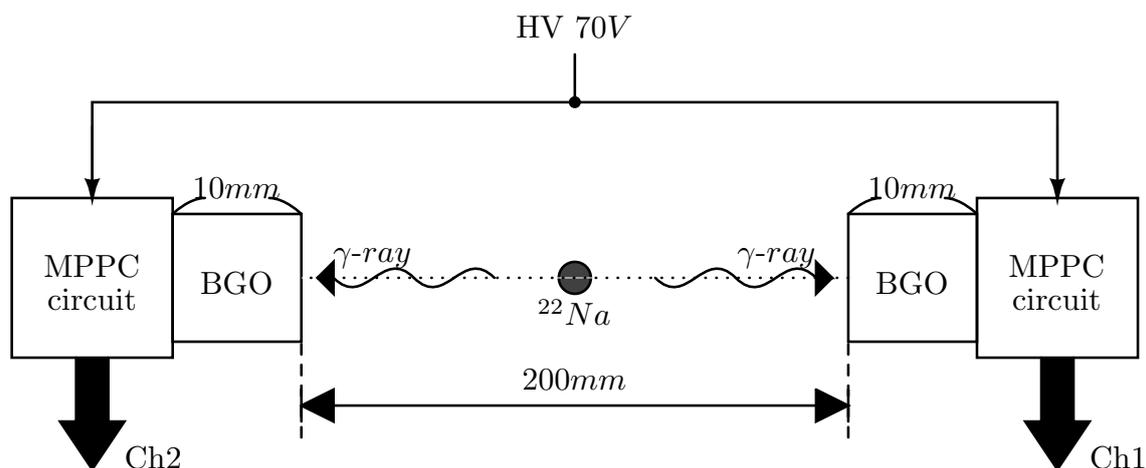


図 5.7: 実験セットアップ図

5.2.2 実験 1:放射線の入射量予想

単位時間あたりのシンチレータの反応率を求めていく。

まずは放射線源 ^{22}Na が 1 つのシンチレータに入射する確率を求める。 ^{22}Na の放射方向は完全ランダムと仮定する。線源 ^{22}Na とシンチレータとの距離が 100mm より半径 100mm の作る表面積は $S_r = 4\pi r^2 = 4\pi 100^2 [\text{mm}^2]$ 、シンチレータサイズが一辺 10mm より入射面積は $S_s = 10^2 = 100\text{mm}^2$ により、線源からランダムで放出された放射線の入射確率は $S_s/S_r = 0.0796[\%]$ となる。用いた線源 ^{22}Na の線量は 91.5kBq より、1 秒間にシンチレータに入射する数は平均 $91500[\text{個}/\text{s}] \times 0.0796[\%] = 72[\text{個}/\text{s}]$ となる。1 秒間に以下の表 5.4 の数が反応する。511keV,1274keV の各反応率は表 5.2 の $500\text{keV},1250\text{keV}$ を用いた。

γ 線 [keV]	コンプトン散乱 [個/s]	光電吸収 [個/s]	備考
511(\approx 500)	57.9 (= 40.2[%] \times 72[個/s] \times 2[個])	49.1 (= 34.1[%] \times 72[個/s] \times 2[個])	1 回の崩壊で γ 線が 2 本発生
1274(\approx 1250)	20.7 (= 28.8[%] \times 72[個/s])	4.1 (= 5.7[%] \times 72[個/s])	

表 5.4: ^{22}Na の γ 線反応量

1 つのシンチレータに対して 1 秒あたり表 5.4 の数だけ取得を行うので実験での 1 秒間あたりの取得数はいかに要になる。

$$65.9[\text{個}/\text{s}] (= 57.9 + 49.1 + 20.7 + 4.1)[\text{個}/\text{s}] \quad (5.3)$$

10 分間取得を行えば以下の数の放射線を検出できると考えられる。

$$39540[\text{個}] (= 65.9[\text{個}/\text{s}] \times 600[\text{s}]) \quad (5.4)$$

取得するコインシデンス設定で 511keV を取得する場合は、両方のシンチレータが同時に反応するには表 5.5 の確率で起こると考えられる。

γ 線 511keV		シンチレータ 1		
		コンプトン散乱 (40.2[%])	光電吸収 (34.1[%])	無反応 (25.7[%])
シンチレータ 2	コンプトン散乱 (40.2[%])	16.1[%]	13.7[%]	10.3[%]
	光電吸収 (34.1[%])	13.7[%]	11.7[%]	8.8[%]
	無反応 (25.7[%])	10.3[%]	8.8[%]	6.6[%]

表 5.5: ^{22}Na の γ 線コインシデンス反応率

コインシデンス設定で両方のシンチレータが反応した時の取得する確率は、両方のシンチレータがコンプトン散乱, 光電吸収のどちらかが起こした時に発生する。その為、コインシデンスの場合のみの取得の際に 511keV での反応確率は以下の確率になると考えられる。

$$55.2[\%] (= 16.1 + 13.7 + 13.7 + 11.7)[\%] \quad (5.5)$$

また、アンチコインシデンスを設定した場合の取得確率は、片方のシンチレータが無反応，もう片方がコンプトン散乱や光電吸収を起こす場合に保存するのでその確率は以下の様になると考えられる。

$$38.4[\%] (= 10.3 + 8.8 + 10.3 + 8.8)[\%] \quad (5.6)$$

反応しない確率は 6.6[%] になる。

コインシデンス設定を変えた場合の取得量は、コインシデンスに設定したものは 511keV の波形のみを取得し、もう一方のシンチレータも反応する確率 (74.3[%]) で取得し、逆にアンチコインシデンス設定したものは波形取得時にもう一方のシンチレータが反応しない事が条件になるので 1274keV はそのまま 511keV は 25.7[%] まで取得数が減ると考えられる。

コインシデンス設定	エネルギー [keV]	コンプトン散乱 [個/s]	光電吸収 [個/s]	小計 [個/s]	合計 [個/s]
全取得 Normal	511	57.9	49.1	107.0	131.8
	1274	20.7	4.1	24.8	
アンチコインシデンス AntiCC	511	14.9	12.6	27.5	52.3
	1274	20.7	4.1	24.8	
コインシデンス CC	511	43.0	36.5	79.5	79.5
	1274	0	0	0	

表 5.6: ^{22}Na のコインシデンス設定による γ 線の単位秒あたりの反応理論値

5.2.3 実験 1:ゲインの調整

取得したヒストグラムのエネルギー値はデジタル値である。デジタル値は同じ線源からの入力でもシンチレータの種類や増幅器のゲインや気温など様々な要因で変化するため、初期設定としてデジタル値をエネルギーに変換するための較正直線を既知の線源を用いて導出する必要がある。また、今回の小型放射線検出器では取得出来るエネルギーの上限と下限が決まっているため、検出したい帯域をうまく範囲内に調整する必要がある。

そこで、調整のしやすい MPPC 読み出し回路の HV を変更すると入力信号の電荷の増加量を変えることが出来るので、デジタル値もそれに比例して買えることが出来る。ゲインの設定は 611keV の γ 線を放出する ^{137}Cs を使用し、測定時間を 10 分、電圧を 71V から 72V を 0.25V 刻みで取得を行い適切な HV を求めた。結果は以下の図 5.8 の様になった。

72V と 71.75 のピークが同じになっているのは、UD_Threshold が 420 により 420 付近で収束している為であると考えられる。また、低エネルギー側にもピークが見られるのは、増幅されたノイズが Threshold を超えて観測範囲に入ってきた為である。今回の実験では、 ^{22}Na 線源の放出する 511keV, 1274keV の両方を測定するため、611keV のピークから約 2 倍の Energy(1274keV) が UD_Threshold に収まる様に HV= 71V と設定して実験を行った。

この実験では 2 つのチャンネルで信号を取得しているが、2 つのチャンネルでゲインが違うことと、各 BGO の反応は対称性によりほぼ同じ波形になると考えて良いので、以降片側のヒストグラムのみを表示する。

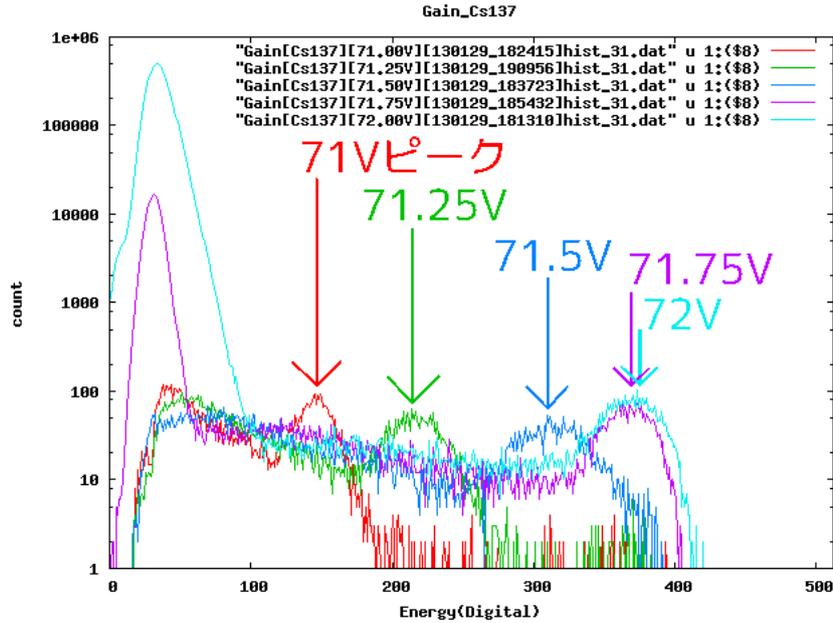


図 5.8: $^{137}\text{Cs}(661\text{keV})$ を用いたゲイン測定

5.2.4 実験 1: コインシデンス設定

コインシデンス設定のパターンは全波形取得 (以下:Normal)、アンチコインシデンス (AntiCC)、コインシデンス (CC) の 3 パターンを取得した。Normal は全ての γ 線のエネルギーをヒストグラムに保存、AntiCC はどちらか片方の BGO が反応した時のみにヒストグラムに保存、CC は両方の BGO が反応した時にヒストグラムに保存する。今回は ch1 と ch2 をそれぞれ A,B とし、取得条件は以下のように設定した。

		Ch1(A)	
		1	0
Ch2 (B)	1	GFC=2	GFC=1
	0	GFC=1	-

表 5.7: コインシデンス条件と GFC

取得条件	論理	GFC=1	GFC=2
Normal	Ch1 OR Ch2	1	1
AntiCC	Ch1 XOR Ch2	1	0
CC	Ch1 AND Ch2	0	1

表 5.8: CoincidenceMask の設定

5.2.5 実験 1: 実験結果

以上の設定で放射線の計測を行った。測定時間は Normal,CC,AntiCC 設定それぞれで 10 分のヒストグラムを取得した。

図 5.9 の取得結果より、Normal 取得から CC 設定のものは 1277keV の取得を除いたヒストグラムの取得を行うことができた。また、AntiCC については逆に同時反応した 511keV の波形の取得量のみを減らすことが出来た。

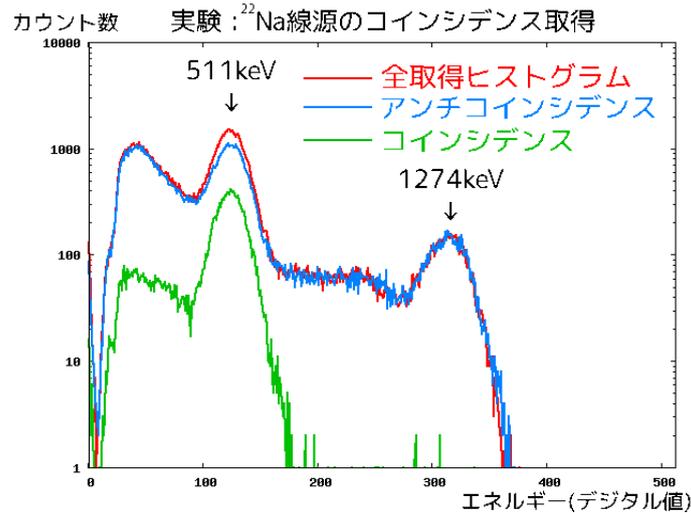


図 5.9: Na22 のコインシデンス設定による違い

5.2.6 実験 1: 考察

今回の実験で、コインシデンスの設定がうまく行っていることが確認された。

取得量に関しては理論値に比べて 511keV の同時反応イベントの CC の取得量、AntiCC の減少量は 40% 程度の値となったが、この問題は線源の配置が 2.5mm ずれただけで取得量が半分になってしまうため配置の微調整が問題していると考えられる。また、線源は線源に対して不透明で線源に対して大きなケースに入っているため、位置を確認しづらい。そのため、再度実験を行う場合は、²²Na の配置がシンチレータの中心にきっちりあう工夫が必要だと考えられる。

5.3 実験 2: コンプトンカメラの基礎実験

実験 2 ではコンプトンカメラの基礎実験を行う。シンチレータ内に入射した γ 線がコンプトン散乱を起こすと、エネルギーの一部を落とした γ 線が放出される。エネルギー量は散乱角によって決まっており、散乱角と放出される γ 線のエネルギーは第 2.5.3 節での式 (2.11) より求められる。コンプトン散乱後の γ 線のエネルギー E_2 とコンプトン散乱時に電子が受け取るエネルギー E_1 とすると以下の量になる (電子の静止エネルギー: $m_e c^2 = 511 \text{keV}$ とした)。

$$E_2 = \frac{m_e c^2}{(1 - \cos\theta) + m_e c^2 / E_\gamma} \quad , \quad E_1 = E_\gamma - E_2 \quad (5.7)$$

線源からの γ 線をシンチレータ 1 でコンプトン散乱を起こさせ、散乱した γ 線がシンチレータ 2 に入り光電吸収を起こした場合の波形とヒストグラムの取得を行う。

5.3.1 実験 2:実験環境

^{137}Cs からの 661keV の放射線を 50mm 離れたシンチレータ 1 に入射させ、シンチレータ内でコンプトン散乱を起こしたガンマ線を 50mm 先のもう 1 つのシンチレータ 2 が光電吸収を起こし 2 つのシンチレータが同時反応した場合に取得を行う。線源からの γ 線がシンチレータ 2 に入らない様に鉛ブロックで遮蔽した。実験を行ったコンプトン散乱角は 90 度散乱の図 5.10 と、 45 度散乱の図 5.11 の 2 パターン行った。

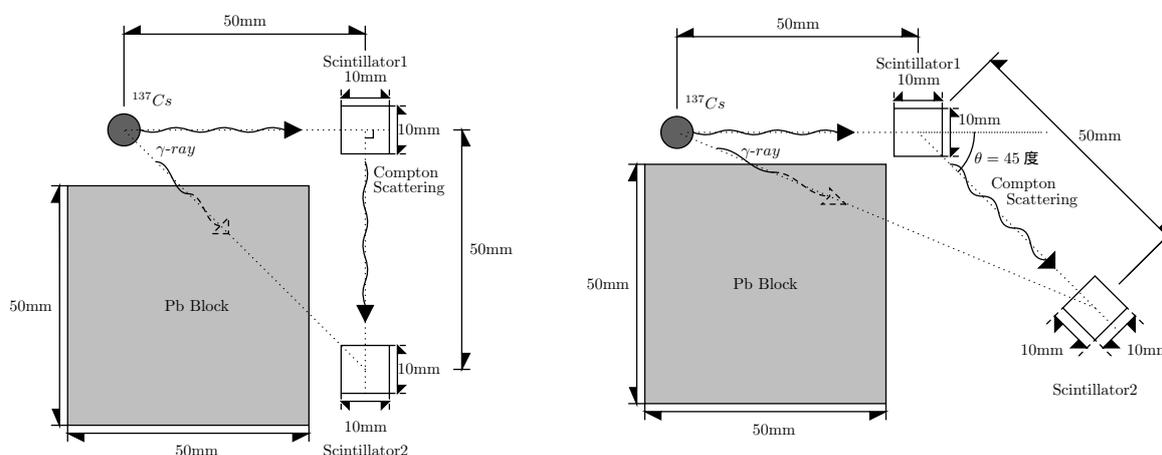


図 5.10: 実験 2 コンプトン散乱の検出 ($\theta=90$ 度) 図 5.11: 実験 2 コンプトン散乱の検出 ($\theta=45$ 度)

実験装置の配置が間違っていないか、コインシデンス設定を行う前に線量の取得を 10 分行った。図 5.12, 図 5.13 よりシンチレータ 1 (緑線) に ^{137}Cs の光電吸収のピークが見え、シンチレータ 2 (赤線) に ^{137}Cs の光電吸収ピークが見られない事を確認してから実験を行った。

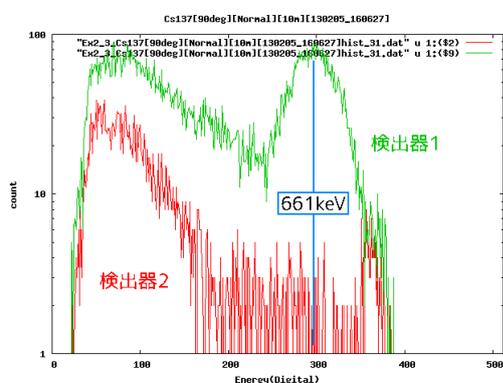


図 5.12: 実験装置の取得 γ 線量 (90 度)

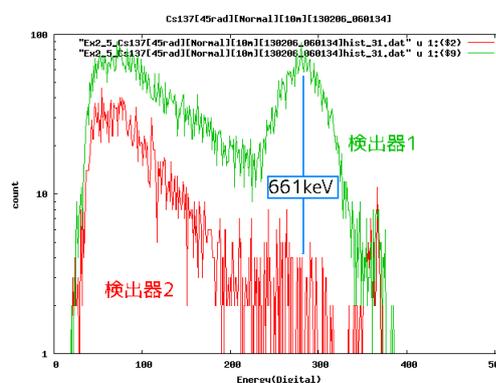


図 5.13: 実験装置の取得 γ 線量 (45 度)

5.3.2 実験 2:放射線の入射エネルギー予想

線源のエネルギー $E_\gamma = 661\text{keV}$ と角度 $\theta = 90$ 度, 45 度 と固定して行うので、この条件でコンプトン散乱と光電吸収が起こった際の各 E_1, E_2 のエネルギー [keV] は表 5.9 の様になる。

散乱角度	$E_1[keV]$	$E_2[keV]$
90 度	372	288
45 度	182	479

表 5.9: コンプトン散乱エネルギー (理論)

5.3.3 実験 2:ゲイン・較正直線

エネルギー [Digital 値] を keV に直すためにゲインの測定を行った。上段が ^{137}Cs による照射で下段が ^{22}Na による照射、左側がシンチレータ 1 の取得結果で右側がシンチレータ 2 での取得結果である。較正用のエネルギーは ^{137}Cs の $661keV$ と ^{22}Na の $511keV$ を用いた。 ^{22}Na には $380[Digital]$ 付近のピークはサチレーションと考えられるので除外した。

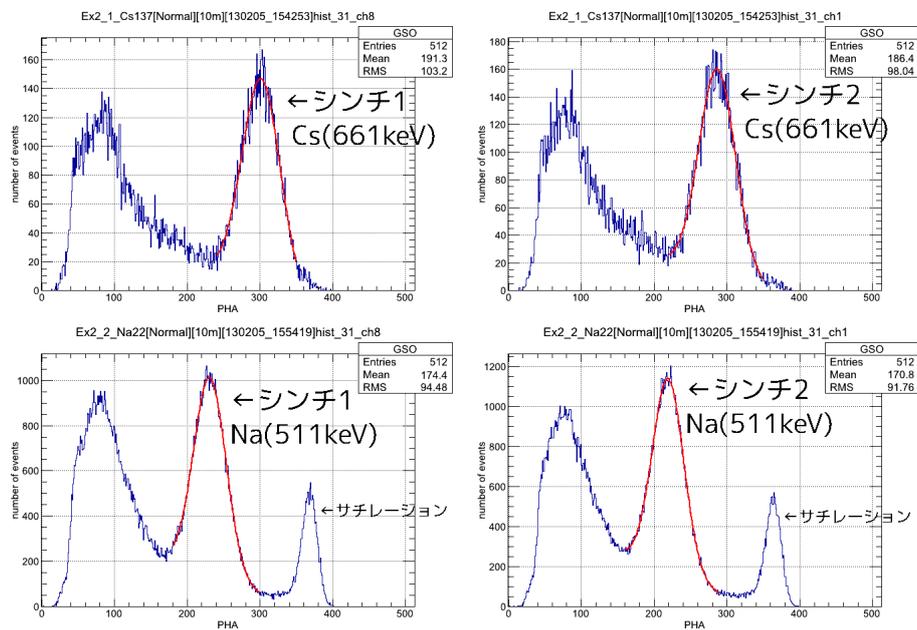


図 5.14: 較正用ヒストグラム (^{137}Cs と ^{22}Na)

$661keV$ と $511keV$ のピークはガウスフィットした頂点を求めると表 5.10 のように求まった。

エネルギー	シンチレータ 1	シンチレータ 2
$661(^{137}Cs)$	$300.8[Digital]$	$286.5[Digital]$
$511(^{22}Na)$	$230.9[Digital]$	$219.3[Digital]$

表 5.10: $^{137}Cs\gamma$ 線のコインシデンス検出量 (理論)

シンチレータの較正直線を表 5.10 より求めると以下の様に求まる。この結果を用いて、各角度における

エネルギー値を求めていく。

$$\begin{cases} \text{シンチレータ 1: エネルギー [keV]} = 2.146 \times \text{Digital 値} + 15.51 \\ \text{シンチレータ 2: エネルギー [keV]} = 2.232 \times \text{Digital 値} + 21.49 \end{cases} \quad (5.8)$$

5.3.4 実験 2: 実験結果

6 時間測定して取得した実験結果は以下のようになった。緑がシンチレータ 1 のエネルギーヒストグラムで、赤がシンチレータ 2 のエネルギーヒストグラムである。

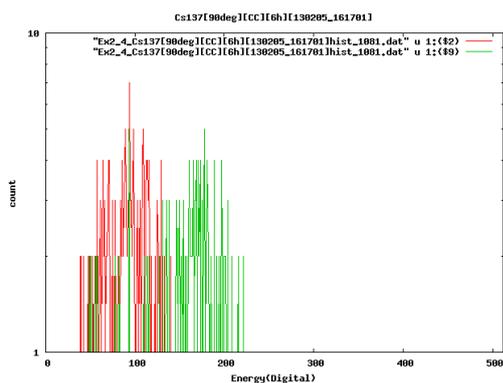


図 5.15: 実験 2 結果 (90 度)

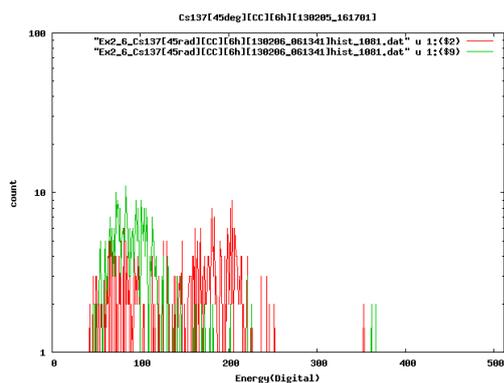


図 5.16: 実験 2 結果 (45 度)

検出数は 90 度が 229 個と 10[%]、45 度が 443 個と 25[%] 程度の取得となった。

取得エネルギーに関しては、最確値を最大カウント数周りで取得したものを較正直線 (5.8) を用いて構成した結果は以下のようになった。

散乱角度	$E_1 [keV]$	$E_2 [keV]$
90 度	397.5[keV] (理論値:372[keV])	231.2[keV] (理論値:288[keV])
45 度	193.6[keV] (理論値:182[keV])	476.8[keV] (理論値:479[keV])

表 5.11: コンプトン散乱エネルギー取得結果

5.3.5 実験 2: 考察

実験 2 では、コンプトン散乱のエネルギーを取得に成功した。

第6章 まとめと今後

本研究では携帯型放射線検出器のロジック回路を PoGOLino 用ロジック回路を元に開発を行い、FADC ボードを用いた 8 チャンネルでの波形取得やヒストグラム生成が可能になった。その際に、コインシデンス設定を CoincidenceGroup と CoincidenceMask の 2 つの導入によりコインシデンス条件を付けることが出来るようになり、 ^{22}Na の放出する 1274keV の γ 線と 511keV の放射線のコインシデンスイベントの取捨選択の一部や、コンプトン散乱角によって変化するコンプトン散乱と吸収でのエネルギーを検出することが出来た。

今回の開発したロジック回路ではデータ取得に PC から命令を送ったが、今後は PC の無い環境でも使用可能な携帯型検出器として、PC の代わりに小型マイコンを用いることで携帯中でも線源からの波形やスペクトルの取得を行える様になって行く事を目指す。また、PC 上で用いることも念頭にし、簡易的なものとは別に全取得コインシデンスパターンとを切り替えられる様に開発していく事を目指す。

付録 A 多チャンネルの制御プログラム

8ch プログラムを動かすには以下の 4 つのプログラム、データを表示するのに 1 つの 5 つのプログラム (+gnuplot) を用いる。各節では詳しいコマンドの入力と設定方法を解説していく。

- checkFADC : ADC 出力値が正常であることをチェック
- trigstartup_8ch : 閾値、上限値などの基本的な取得条件を設定する。
- coincidence_setting : コインシデンス条件設定を行う。
- Read_wf_8histo512_8ch : 波形とヒストグラムの取得を行う。
- CompressedWave : 取得した波形を表示する。

アルゴリズムの表現の関係で 2 進数、10 進数、16 進数が織り交ざって表現している。そのため、これらが混ざらないように各数字の頭に 0b(2 進数:binary)、0x(16 進数 hexadecimal) をつける。また、10 進数(10 進数:decimal) は 0d を付けるが、特に表記がなければ基本省略する。

(例 : 0b100000 = 32 (= 0d0032) = 0x0020)

A.1 checkFADC

checkFADC プログラムでは、FADC が正常に起動しているかどうかを 3 つのモードで検証する。[checking deskew][checking sync] で 2 種類の固定値が正常に出力されるかを検証し、正常な値が取得できていれば OK、値が異なっていれば NG と出力される。また、[reset and dump] で実際に入力から ADC が読み取っている値を出力する。正常であれば以下の様な値を表示する。

1. checking deskew :0x0aaa(=0b101010101010)
2. checking sync :0x003f(=0b000000111111)
3. reset and dump :ADC 値 (0x0300 前後 = 768 前後)

値が異常な場合は FADC ボードを再起動して再度適正な値が出るまで checkFADC を行う。また、何らかの拍子に ADC 値が突然悪くなる場合があるので、プログラムの調子が少しでも変だと思ったらチェックすると良い。

```
[matsuoka@maihama:hirano_8ch_Program]8:43$ ./checkFADC
SpaceWireIF OverIP is used.
Input Hostname or IP address of SpaceWire-to-TCP/IP converter.
note : IP port number can be provided separated by ':'; 133.11.165.117:10030
note : If no port number is given, the default 10030 will be used.
Hostname or IP address > 192.168.0.131 ←(使う GbE 等のアドレス)

Hostname : 192.168.0.131
TCP/IP Portnumber : 10030
SSDTPModule : SSDTP2 Protocol is used.
Destination
Logical Address : 30
Path Address : none
Destination Key : 02
CRC Version : Draft F
Word Width : 1bytes-1Word

Source
Logical Address : f1
Path Address : none
Path Address Len. : 00

Please set Destination information
checkFADC
checking deskew ←(0x0aaa を出力するモード)
ch0 is OK. 0x0aaa
ch1 is OK. 0x0aaa
ch2 is OK. 0x0aaa
ch3 is OK. 0x0aaa
ch4 is OK. 0x0aaa
ch5 is OK. 0x0aaa
ch6 is OK. 0x0aaa
ch7 is OK. 0x0aaa
checking sync ←(0x003f を出力するモード)
ch0 is OK. 0x003f
ch1 is OK. 0x003f
ch2 is OK. 0x003f
```

(続く)

```

ch3 is OK. 0x003f
ch4 is OK. 0x003f
ch5 is OK. 0x003f
ch6 is OK. 0x003f
ch7 is OK. 0x003f
reset and dump    ←(実際の ADC 値を出力 0x0300 前後)
ch0 : 0x030d
ch1 : 0x033b
ch2 : 0x0302
ch3 : 0x0310
ch4 : 0x032b
ch5 : 0x030c
ch6 : 0x0308
ch7 : 0x0303

```

A.2 trigstartup_8ch

trigstartup_8ch ではコインシデンス以外の波形取得の基本的な閾値、上限値、間引く、ディレイの 4 つを設定する。また、この設定を一括で全てのチャンネルに当てはめるか、各チャンネル毎に個別に設定することも出来る。

1. Threshold : 閾値 (default:50)(最大 4096)
2. UD.Threshold : 上限値 (default:3500)(最大 4096)
3. MABIKU : Waveform の出力間隔を間引く (default:0)(最大 9)
4. DelayLength : Delay を引く clock の長さ (default:1)(0:ディレイなし , 1:4clock , 2:16clock , 3:32clock)

各入力値を図で関係を表すと、Threshold , UD.Threshold , MABIKU は図 A.1 に対応し、Threshold は閾値を超えたものを入力波形と判定する値、UD.Threshold は波形の上限、MABIKU は波形の立ち上がりが遅くて波形のピークが見られない (50clock に収まらない) 場合 (図 A.1 左) に取得する波形を数個飛ばしごとにする設定 (図 A.1 右) を行う。

また、DelayLength は図 A.2 の様に数 clock 前の自身の波形を発生させてその差分を取る。それぞれ Delay1 が 4clock、Delay2 が 16clock、Delay3 が 32clock の波形を用いて実波形との差分を取りオフセットを取り除く。

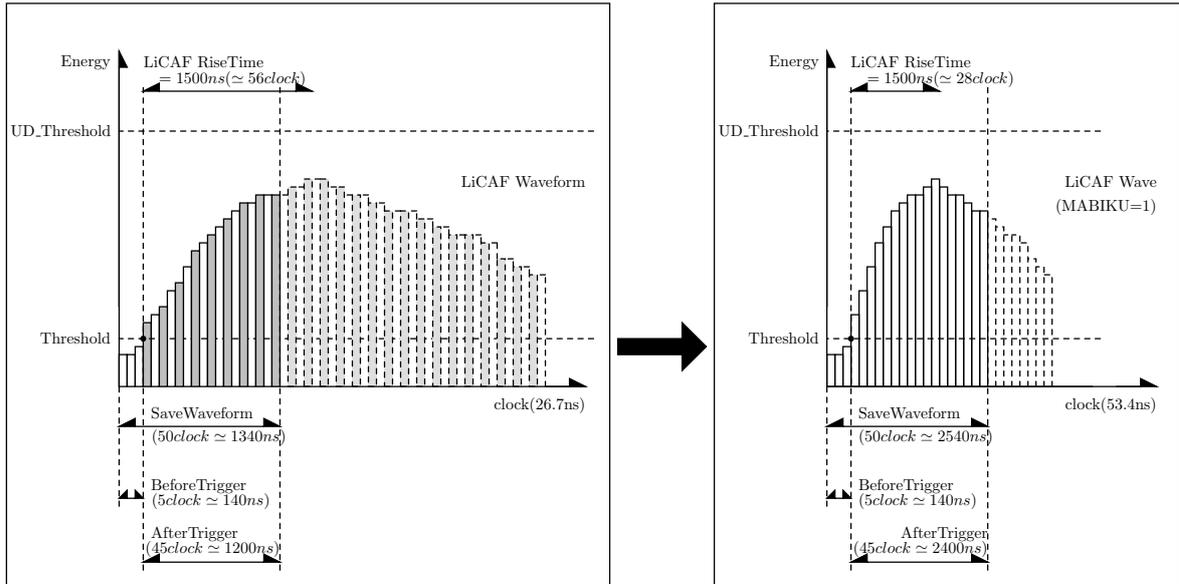


図 A.1: Threshold,UD_Threshold,MABIKU のイメージ図 (左:MABIKU=0)(右:MABIKU=1)

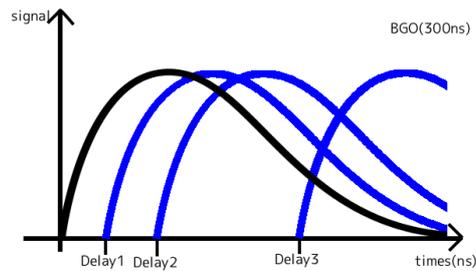


図 A.2: Delay の波形

trigstartup_8ch - terminal 画面

```
[matsuoka@maihama:hirano_8ch_Program]8:45$ ./srtrigstartup
(... 中略...)
Hostname or IP address > 192.168.0.131  ←(使う GbE 等のアドレス)
(... 中略...)
Input All(0) or not(1)  ←(一括設定 (0) か各チャンネル個別設定 (1))
> 0
Input Threshold Data  ←(Threshold の一括設定)(範囲 0-4096)
> 100
Input UD_Threshold Data  ←(UD_Threshold の一括設定)(範囲 0-4096)
> 3500
```

(続く)

```

Input Mabiku Data  ←(Mabiku の一括設定)(範囲 0-9)
> 0
Input Delay Data  ←(Delay の一括設定)(範囲 0-3)
> 2
setting trigger Threshold  ←(各 Threshold[16 進数表示])
Address Data
0101-0050 0x0064
0101-0052 0x0064
(... 中略...)
0101-005e 0x0064

setting trigger UD_Threshold  ←(各 UD_Threshold[16 進数表示])
Address Data
0101-0090 0x0DAC
0101-0092 0x0DAC
(... 中略...)
0101-009e 0x0DAC

setting trigger Mabiku  ←(各 Mabiku 値)
Address Data
0101-e100 0x0000
0101-e102 0x0000
(... 中略...)
0101-e10e 0x0000

setting trigger DelayLength  ←(各 DelayLength 値)
Address Data
0101-e100 0x0002
0101-e102 0x0002
(... 中略...)
0101-e10e 0x0002

```

A.3 coincidence_setting

coincidence_setting はコインシデンス (同時発生) 時の取得の可否を決めるプログラムである。詳しい設定方法は 2012 年度平野卒論を参考にして分かると嬉しい。

行う設定は Coincidence_Group と Coincidence_Mask を設定する。各チャンネルを Coincidence_Group で A-H,Z のグループに所属させ、所属させたグループのトリガー数の状況に応じて出力の可否を決定する

Coincidence_Mask を設定する。ただし、グループを入力する際は A-H ではなく 1-8 とする。グループ Z は入力があると取得を行わないため、バックグラウンドなどに使うと良い。

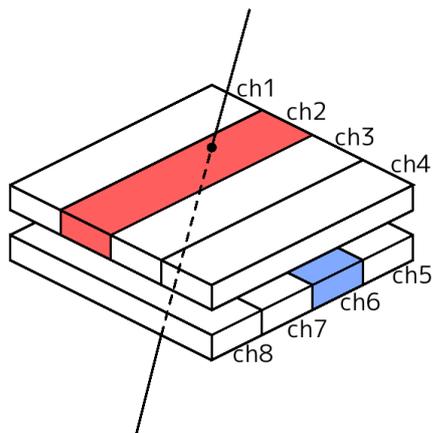


図 A.3: ミューオン入射位置検出器

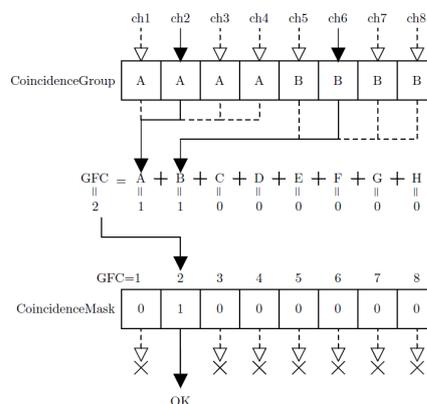


図 A.4: コインシデンスロジック

```

----- coincidence_setting - terminal 画面 -----
[matsuoka@maihama:hirano_8ch_Program]8:42$ ./coincidence_setting
(... 中略...)
Hostname or IP address > 192.168.0.131  <=(使う GbE 等のアドレス)
(... 中略...)
Input Coincidence is [Normal(0)] or [HIRANO(1)] Mode > 1  <=(“1”で HIRANO モードでの設定)
Now is the HIRANO Mode!

Input Coincidence_Group(1-8ch) Data
<12345678 ch>
> 11112222  <=(各チャンネルのグループ分け [1~8=A-H, 0=Z])

Input Coincidence_Mask Data
<12345678 GFC>
> 11000000  <=(出力するマスク値を入力)

setting CoincidenceGroup
Address Data
0101-00b0 0x0001  <=(各チャンネルのグループ分けデータ)
0101-00b2 0x0001

(続く)

```

```

0101-00b4 0x0001
0101-00b6 0x0001
0101-00b8 0x0002
0101-00ba 0x0002
0101-00bc 0x0002
0101-00be 0x0002
setting CoincidenceMask
Address Data
0101-00d0 0x0003 ←(Coincidence_Mask[8bit の 16 進数表示])

```

A.4 Read_wf_8histo512_8ch

Read_wf_8histo512_8ch プログラムでは波形保存とヒストグラム生成の同時を行う。このプログラムを走らせると以下のログファイル、波形データ、ヒストグラムデータが出力される。

- (日付_時間)wave_(番号).dat
- (日付_時間)hist_(番号).dat
- (日付_時間).log

(日付_時間) はプログラムを走らせた日時と時間が付き、プログラムを走らせた日付が例えば 2013 年 3 月 12 日 17 時 1 分 55 秒の場合は 130312_170155 となる。(番号) はアウトプットした順番に番号がつく。

取得した波形 (wave.dat) や取得したヒストグラム (hist.dat) は後述のプログラム CompressedWave、gnu-plot を用いることで表示できる (表示方法は第 A.5 節, 第 A.6 節)。ログはテキストエディタで表示できる。

```

[matsuoka@maihama:posix]12:29$ ./Read_wf_histo_8ch
(... 中略...)
Hostname or IP address > 192.168.0.131 ←(使う GbE 等のアドレス)
(... 中略...)

Now Time : 130313_122919
Output Histogram File is 130313_122919hist_1.dat
Output SaveWave File is 130313_122919wave_1.dat
How long ? (sec) —> 50 ←(測定時間 (秒))

```

(続く)

```
DATA Acquisition Start!
Start : 130313_122922
Reading initial Histo content...
Done.
wait 50 sec...
new hist file is created.
Done.

1 / 50 sec. | events : 0
new hist file is created.
Done.
21 / 50 sec. | events : 0
new hist file is created.
Done.
41 / 50 sec. | events : 0
Data acquisition finished.
Stop : 130313_123012
Dead Time (s) is 0.00010922667
Start Time : 130313_122919
END Time : 130313_123012

Output Histogram File is 130313_122919hist_4.dat  ←(ヒストグラムデータ)
Output SaveWave File is 130313_122919wave_1.dat  ←(波形データ)
```

A.5 CompressedWave

CompressedWave では、A.4 で取得した wave.dat の波形を表示することが出来る。

terminal 画面

```
> ./compressedwaveform (見たい波形データ)
```

と入力することで起動する。起動後は数字で制御する。

- 0 : 現在の波形を再表示
- 1 : 次の波形を表示する
- 2 : 前の波形を表示する
- 999 : 終わる。

A.5.1 wave.dat の保存形式

役に立つかわからないけど、VHDL を構成する時にメモとなる表記を行う。wave.dat というデータの構成は以下のコマンドで内容を見ることが出来る。詳しい詳細は od コマンドを参照。

```
terminal 画面
> od -x -w110 (ファイル名) |less
```

実際に取得した wave.dat のデータを od コマンドによって表示させると、以下の表のような数字の羅列が表示される。wave.dat に格納されている保存データの例は図 A.1 の様となる。

行番号	保存データ (110Byte)												
0000000	0000	0f51	0402	7066	966c	028f	0291	0291	0296	...(中略)...	02cc	02d2	02d5
0000156	0000	0f52	0402	7976	966c	036b	0369	0368	0366	...(中略)...	0332	0331	0332
0000334	0000	0f53	0402	7b8e	966c	02cc	02c7	02c5	02c4	...(中略)...	0289	0287	0289
0000512	0000	0f54	2005	7f16	966c	0280	0285	0283	0284	...(中略)...	0279	0278	0276
0000670	0000	0f55	2005	af29	966c	02cb	02cb	02cb	02cd	...(中略)...	02db	02de	02dc

表 A.1: wave.dat の保存データ (例)

これらの数字には構成が決まっており、波形の保存は 1 つの波形に付き図 A.5 の構成が連なって保存される。

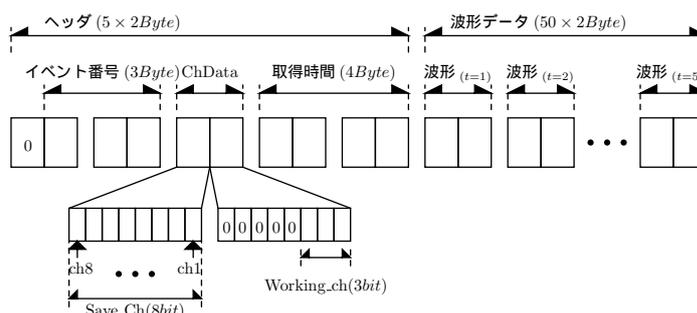


図 A.5: 一波形の保存データ構成

したがって、各対応表として見てやると図 A.2 の様に成る。このデータを読み取って CompressedWave は動いている。

保存データ													
0	イベント番号	save_Ch	Working_ch	所得時間	波形データ								
0000000	00	00 0f51	04	02	7066 966c	028f	0291	0291	0296	...(中略)...	02cc	02d2	02d5
0000156	00	00 0f52	04	02	7976 966c	036b	0369	0368	0366	...(中略)...	0332	0331	0332
0000334	00	00 0f53	04	02	7b8e 966c	02cc	02c7	02c5	02c4	...(中略)...	0289	0287	0289
0000512	00	00 0f54	20	05	7f16 966c	0280	0285	0283	0284	...(中略)...	0279	0278	0276
0000670	00	00 0f55	20	05	af29 966c	02cb	02cb	02cb	02cd	...(中略)...	02db	02de	02dc

表 A.2: wave.dat の保存データの構成 (例)

A.6 gnuplot によるヒストグラム表示

Read_wf_8histo512_8ch を取得したヒストグラムデータをグラフに表示する。ヒストグラムデータは番号の一番大きいものが最新の取得データとなるため、表示に用いるのは (番号) の一番大きい hist データを用いる。

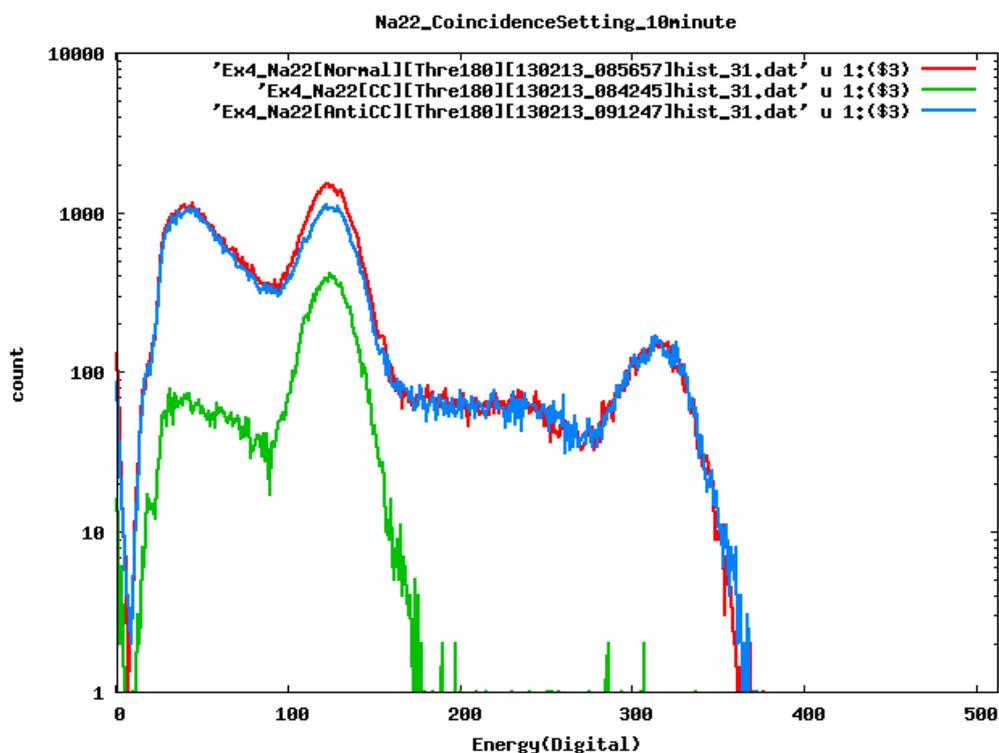


図 A.6: gnuplot 表示例

gnuplot を起動したものと話を進める (起動してない、インストールしてない場合は周りの先輩などに聞くか別の物を導入して自力で表示する)。表示するだけの場合は以下のように表示を行えば良い。

gnuplot 画面 (とにかくプロット)

```
gnuplot> plot '130210_200543hist_16.dat' u 1:($4) w lp
```

内訳は、ファイル (130210_200543hist_16.dat) のチャンネル 2 (データの行 (4)) を読み取ってライン (l) とプロット (p) を表示するとなっている。データの行 (\$4) はチャンネルとは対応していないため、エラーが出る場合は一回データを Excel 等の表で表示して何列目であるかを確認してから表示を行うと良い (表 A.3 参照)。後半の (l) と (p) はどちらかだけでも表示できる。

チャンネル 2 がデータの行 (4) となる理由は保存データ形式を Excel などで表示すると分かる。実際に表示した例は以下の表 A.3 となる。

ch:	0	Time:	351.823	DeadTime:	0	Exposure:	281	
ch:	1	Time:	351.829	DeadTime:	0	Exposure:	281	
ch:	2	Time:	351.837	DeadTime:	0.174435	Exposure:	281	
ch:	3	Time:	351.862	DeadTime:	0	Exposure:	281	
ch:	4	Time:	351.867	DeadTime:	0	Exposure:	281	
ch:	5	Time:	351.874	DeadTime:	0.00895659	Exposure:	281	
ch:	6	Time:	351.88	DeadTime:	0	Exposure:	281	
ch:	7	Time:	351.887	DeadTime:	0	Exposure:	281	
0	0	0	3521	0	0	3524	0	0
1	0	0	5	0	0	4	0	0
2	0	0	3	0	0	4	0	0
3	0	0	3	0	0	2	0	0
4	0	0	0	0	0	1	0	0
(中略)								
510	0	0	0	0	0	0	0	0
511	0	0	0	0	0	0	0	0

表 A.3: hist.dat の全体構成

データの上部が各チャンネルにおける測定時間 (Time)、デッドタイム (DeadTime)、波形の取得時間 (Exposure) を表示し、下部が各チャンネルでのヒストグラムデータになる。下部のデータは 1 列目がエネルギー値が 1 から 512 まで書いており、2 列目から 9 列目がそれぞれ ch0 から ch7 のヒストグラムの値となる。そのため、チャンネル選択は実際のチャンネルに+2 した値を読みこませる必要がある。

エネルギー	ch0	ch1	ch2	ch3	ch4	ch5	ch6	ch7
0	0	0	3521	0	0	3524	0	0
1	0	0	5	0	0	4	0	0
2	0	0	3	0	0	4	0	0
3	0	0	3	0	0	2	0	0
4	0	0	0	0	0	1	0	0
(中略)								
510	0	0	0	0	0	0	0	0
511	0	0	0	0	0	0	0	0

表 A.4: hist.dat の構成 (下部)

ただ表示させるだけでは少々ブサイクなのでオプションとして、以下の物をプロットする前に打ち込むと卒論修論などで使えるようになるかもしれない。それぞれの内訳は実際に使って見たら分かるので試してみるのみ。

————— gnuplot 画面 (オプションを加えてプロット) —————

```
gnuplot> set title 'Na22_5minute'  ←(タイトルに「Na22_5minute」と表示)
gnuplot> set xlabel 'Energy(Digital)'  ←(X 軸に「Energy(Digital)」とラベルを貼る)
gnuplot> set ylabel 'count'  ←(Y 軸に「count」とラベルを貼る)
gnuplot> set xrange [0:512]  ←(X 軸の表示領域を 0-512 にする)
gnuplot> set logscale y  ←(Y 軸を Log スケールにする)
gnuplot> plot '130210_200543hist_16.dat' u 1:($4) w lp  ←(プロットする)
```

複数重ねて表示させる場合は、カンマ (,) で分けると複数のヒストグラムを表示することが出来る。以下

では 130210_200543hist_16.dat のチャンネル 2 とチャンネル 5 の波形を重ねて表示している。

————— gnuplot 画面 (複数の波形を重ねて表示) —————

```
gnuplot> plot '130210_200543hist_16.dat' u 1:($4) w lp, '130210_200543hist_16.dat' u 1:($7)
w lp
```

また、グラフや上記の表示を png などデータとしてグラフをアウトプットする際は、更に以下のオプションを用いることでファイルに出力することが出来る。これによりプロットしたグラフを **output.png** という名前で保存する。保存箇所は gnuplot を起動した際のプロンプトのディレクトリの位置。

————— gnuplot 画面 (png ファイルにアウトプット) —————

```
gnuplot> set terminal png
gnuplot> set output 'output.png'
gnuplot> plot '130210_200543hist_16.dat' u 1:($4) w lp
```

A.7 メモ (備忘録)

色々やりたいことがあったもののタイムリミットによりできなかった事を、誰かがやってくれるんじゃないかという希望的観測のもと、メモとして書き記す。引き継いだ方、よろしくお願ひします。

A.7.1 coincidence_setting(全取得モード)

Coincidence_setting では Normal モードという取得する全パターンを設定するものを実装しようと思ったが完成することができなかったのて、コインシデンスの全パターンの案をメモとして残す。

作ろうとしたものは、例えばチャンネル 1 のみの反応があった時にデータを取得する場合に

```
input data:( 1 0 0 0 0 0 0 0 )
             ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑
             ch1 ch2 ch3 ch4 ch5 ch6 ch7 ch8
```

と順番にビット値を入力する (1 が入力有りて 0 が入力無し)。この入力により ch1 のみに反応があったパターンで正常な波形であると判定するが、ch1 と ch2 の同時反応など、ch1 以外に同時反応がある場合は取得を行わないので注意が必要である。ch1 と ch2 との同時反応時も取得する場合は

```
input data:( 1 1 0 0 0 0 0 0 )
             ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑
             ch1 ch2 ch3 ch4 ch5 ch6 ch7 ch8
```

と入力することで追加で ch1 と ch2 の同時反応のパターン時にも取得を行う。入力を終える場合は 00000000 と入力することでコインシデンスの設定を終了する。

その結果、入力パターンは 0b00000000 から 0b11111111(0-255) までの 256 パターンがあるので、256bit のメモリに全てのパターンを入力を行えばあらゆるパターンに対応することが出来る。例えば、チャンネル 1 だけの反応の場合は 10000000 となるので、この値を 2 進数として読んだものを 10 進数に直すと 128 となるので、以下の表の 128 が 1(出力可) か 0(出力不可) かを読み出す。同様にチャンネル 2,7 の同時反応の場合は 01000010←66 より、表から 66 のメモリの bit 値を読み、出力の可否を決定する。

メモリ (16bit)	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]
メモリ 1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
メモリ 2	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
メモリ 3	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
メモリ 4	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
メモリ 5	79	78	77	76	75	74	73	72	71	70	69	68	67	66	65	64
メモリ 6	95	94	93	92	91	90	89	88	87	86	85	84	83	82	81	80
メモリ 7	111	110	109	108	107	106	105	104	103	102	101	100	99	98	97	96
メモリ 8	127	126	125	124	123	122	121	120	119	118	117	116	115	114	113	112
メモリ 9	143	142	141	140	139	138	137	136	135	134	133	132	131	130	129	128
メモリ 10	159	158	157	156	155	154	153	152	151	150	149	148	147	146	145	144
メモリ 11	175	174	173	172	171	170	169	168	167	166	165	164	163	162	161	160
メモリ 12	191	190	189	188	187	186	185	184	183	182	181	180	179	178	177	176
メモリ 13	207	206	205	204	203	202	201	200	199	198	197	196	195	194	193	192
メモリ 14	223	222	221	220	219	218	217	216	215	214	213	212	211	210	209	208
メモリ 15	239	238	237	236	235	234	233	232	231	230	229	228	227	226	225	224
メモリ 16	255	254	253	252	251	250	249	248	247	246	245	244	243	242	241	240

表 A.5: 全パターンコインシデンスの取得設定の格納例

謝辞

この卒業論文を書くまでに色々な方にお世話になりました。ひろたか先生にはロジック回路の制作にあたり PoGOLino 用のプログラムを親身になって教えていただいたり、実験で多くの助言を頂きました。おかげで完成させることができ、本当に感謝しております。深澤先生には卒論を書くにあたって添削をして頂き、提出日ギリギリまで修正して頂きまともな論文を書き上げることができました。大野先生には PC 関係のトラブルや FADC からの信号読み取りプログラムの設定などで助けて頂きました。PC 関係は出来る方だと自負していましたが、謙虚になります。M2 の木村さんには客観的な側からの意見や相談に乗って頂きました。良く脱線することがありましたがためになりました。M1 の中川さんには FADC ボードの修理や実験の際に沢山のアドバイスを頂きました。放射線実験での電源やパルサーなどの使い方を教えていただいたり、MPPC 用の読み取り回路の制作を教えていただいたりして勉強になりました。M1 の河野さん、Merlin、PoGOLite の性能テストの時にお世話になりました。英会話ががんばります。

1 年と言う短い間でしたが大変お世話になりました。今まで有難う御座いました。

関連図書

- [1] 岡村定矩「人類の住む宇宙 (シリーズ現代の天文学 第1巻)」
- [2] 長谷川裕恭「VHDLによるハードウェア設計入門 言語入力によるロジック回路設計手法を身につけよう」
- [3] 井上一, 高橋忠幸, 水本好彦, 小山勝二「宇宙の観測高エネルギー天文学 (シリーズ現代の天文学) 第17巻」
- [4] 核物理学講座 第9「放射線と物質」
- [5] 中 幸政「VHDLとCPLDによるロジック設計入門」
- [6] 2003年中本達也 修論「多層シリコンストリップとBGO・フォトダイオードを用いた宇宙軟ガンマ線コンプトンカメラの開発」
- [7] 2011年中川崇之 卒論「NaI(Tl)シンチレータとアレイ型APDを組み合わせた携帯型放射線モニターの構築」
- [8] 2010年宇井崇紘 卒論「アレイ型ガイガーAPDを用いたX線衛星搭載アクティブシールド用BGOの光読み出しの試み」
- [9] 2010年木村太輔 卒論「衛星搭載軟ガンマ線検出器用ファインコリメータの放射化特性」
- [10] 2010松岡正之 修論「宇宙軟ガンマ線の気球・衛星観測におけるシンチレータ信号の波形デジタル処理の実証」
- [11] 2007松岡正之 卒論「新衛星通信規格SpaceWireを用いたデータ収集システムの開発」
- [12] 2005田中琢也 卒論「宇宙X線観測用放射線検出器多チャンネル読み出しシステムの開発」
- [13] PoGOLite project < <http://www.particle.kth.se/pogolite/> >
- [14] 宇宙研 X線天文グループ <<http://www.astro.isas.jaxa.jp/>>
- [15] 宇宙船物理学研究室 < <http://alpha.sci.osaka-cu.ac.jp/grapes3/detectors-j.html> >
- [16] Xilinx <<http://japan.xilinx.com/>>
- [17] Spartan-3(XC3S1000-FT256) SUZAKU <<http://suzaku.atmark-techno.com/dev/dictionary/XC3S1000>>
- [18] SpaceWire とは? <<http://shimafuji.co.jp/spacewire/explanation/spacewire.html>>