

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ィジタル処理の実証

広島大学 理学研究科 物理科学専攻

M083451

松岡正之

高エネルギー宇宙・可視赤外線天文学研究室

主査：深澤泰司 副査：杉立徹

2011年2月10日

概要

超新星残骸や銀河団、ブラックホールなどの高エネルギー天体の物理状態を研究する上で、硬 X 線・ガンマ線の観測は強力な手段となる。しかしこの帯域は今まで検出器の感度が十分ではなかったため他波長に比べると観測が進んでいない。これは天体からの信号が弱い上に、バックグラウンドが高いからである。宇宙空間では宇宙線に由来する荷電粒子、中性子、X 線やガンマ線がバックグラウンドとなり、そのフラックスは天体信号の 1000 倍にも達する。さらに 10 keV から MeV 以下の硬 X 線・軟ガンマ線帯域は、10 keV 以下の X 線帯域では技術が確立している望遠鏡による集光が難しく、また光電吸収とコンプトン散乱が混交するために S/N 比を向上させるのが難しい。

この硬 X 線・軟ガンマ線帯域において良い感度を達成するには徹底したバックグラウンド除去が不可欠である。そのため、2011 年度に放球予定である気球搭載の宇宙硬 X 線偏光検出器 PoGOLite と 2013 年度に打上げ予定の次期 X 線天文衛星 ASTRO-H の硬 X 線撮像検出器 (HXI) と軟ガンマ線検出器 (SGD) は「すぐ」衛星で実績のある BGO アクティブシールドを搭載する。アクティブシールドとは主検出部を視線方向以外シンチレータで囲ったもので、シールドそのものを検出器とすることにより、例えばシールドと主検出部の同時にイベントが観測された場合、その反同時計数を取ることでバックグラウンドを除去することが出来る。

PoGOLite は 25 - 80 keV における偏光観測が目的であり、そのために主検出器 61 ユニット、アクティブシールド 30 ユニット搭載に加え、世界で初めてバックグラウンドモニタのための中性子シンチレータ検出器を搭載する。宇宙線と大気が反応して生成される中性子はバックグラウンドの主要な成分であり、このフラックスをモニタできればバックグラウンドの変動を見積もれるためより良い感度を達成できる可能性がある。私は主検出部とアクティブシールド用の 91 ユニットの信号処理とは別に、中性子シンチレータ用に独立に動く信号処理システムを開発し、正しく動作することを確認した。

ASTRO-H の HXI は 5 - 80 keV での撮像を、SGD は 10 - 600 keV における広帯域の観測を行う。衛星搭載の検出器であるためスペースが限られることから、HXI/SGD の BGO シンチレータの読み出しにはアバランシェフォトダイオード (APD) を用いる。APD は量子効率が高く小型であるが半導体検出器であるため光電子増倍管よりも信号ノイズが大きい。そこでアクティブシールド部で微弱なバックグラウンド信号に対しても反同時係数信号を出力するために後段での回路フィルタが重要になる。限られた回路面積でアナログフィルタを実装することは難しいため、ASTRO-H では FPGA によるディジタルフィルタを採用している。これにより複雑な周波数特性を実装、そのパラメタを変更することも可能となる。本研究では実際に HXI/SGD 検出器のディジタルフィルタに搭載することを想定した FIR フィルタを作成した。実機同様の条件で BGO+APD 信号にどのような影響を与えるか調べ、フィルタを加えることで閾値をおよそ 40 % 下げることに成功した。

目 次

第 1 章 序論	9
1.1 背景	9
1.2 研究の目的	10
第 2 章 SpaceWire	12
2.1 SpaceWire の特徴	12
2.2 SpaceWire IF 搭載 Board	14
2.2.1 DIO Board	15
2.2.2 FADC Board	15
2.2.3 Router Board	16
2.3 FPGA	16
2.3.1 VHDL	17
2.4 SpaceCube と Gigabit Ethernet	18
2.4.1 SpaceCube	18
2.4.2 SpaceWire Gigabit Ethernet	19
2.5 Remote Memory Access Protocol(RMAP)	20
2.6 SpaceWire を用いた観測器器 1:PoGOLite	21
2.6.1 Phoswitch Detector-Cells (PDC)	23
2.6.2 Side Anticoincidence Shield (SAS)	26
2.7 SpaceWire を用いた観測器器 2:ASTRO-H	26
2.7.1 HXI/SGD	28
2.7.2 ASTRO-H における BGO アクティブシールド	32
2.7.3 BGO 信号処理系回路	34
第 3 章 PoGOLite 用データ収集システム開発	36
3.1 研究目的	36
3.2 PoGOLite で実装すべき機能とその開発	38
3.2.1 波形保存プロセス	38
3.2.2 ヒストグラム生成	43
3.2.3 FADC Board と DIO Board での開発	46
3.3 中性子シンチレータとそれに向けた開発	49
3.3.1 大気バックグラウンド	49
3.3.2 中性子シンチレータ	51
3.3.3 中性子シンチレータ用のデータ取得システム構築	54

第4章 Astro-H HXI/SGD用BGO処理回路の開発	58
4.1 研究目的	58
4.2 ディジタルフィルタ	59
4.2.1 移動平均フィルタと差分フィルタの表現	60
4.3 ディジタルフィルタを用いたTrigger試験	62
4.4 高度なFIRフィルタの適用	69
4.4.1 FIRフィルタ	69
4.4.2 BPFをFIRフィルタで実現	70
4.4.3 FIRフィルタ遅延器数・係数精度	71
4.4.4 波形整形アンプを模擬したFIRフィルタの設計	74
4.4.5 ソフトウェアによるFIRの効果検証	80
4.5 ADC素子の選定	84
4.5.1 ADC128S102	85
4.5.2 ADC128S102シリアル通信方式	85
4.6 ADC128S102を用いたFIRフィルタの性能評価	88
4.6.1 ADC128S102評価用基板	88
4.6.2 FPGAへのFIRフィルタ組込み	90
第5章まとめと今後	99
付録A ディジタルフィルタ(Digital Filter)	100
A.1 アナログフィルタとディジタルフィルタ	101
A.2 ディジタル信号とディジタルフィルタ	104
A.2.1 ディジタル信号とは	104
A.2.2 フィルタブロック図、差分方程式から伝達関数と周波数特性まで	105
A.3 ディジタルフィルタの例	108
A.3.1 移動平均フィルタ(LPF)	108
A.3.2 差分フィルタ(HPF)	109
A.3.3 共振器	111
A.3.4 ノッチ・フィルタ	112
A.4 FIRフィルタ	113
A.4.1 FIRフィルタ	114
A.4.2 FIRフィルタの係数導出	116
A.5 代表的なフィルタをFIRフィルタで実現	120
A.5.1 HPF	120
A.5.2 BPF	121
付録B ADC128S102評価用基板設計・製作	123
B.1 ADC128S102評価用基板回路アナログ部	123
B.1.1 アナログ部回路設計	123
B.1.2 LTspice	123

B.1.3 LTspice によるアナログ回路シミュレート	125
B.2 ADC128S102 評価用基板回路 デジタル部	126

図 目 次

2.1	Data-Strobe 信号	12
2.2	ケーブルの構造 [10]	13
2.3	コネクタ・ピンアサイン [10]	13
2.4	SpaceWire における特定ノードへの迂回パス	13
2.5	SpaceWire FPGA Memory Map の例 [26][27]	14
2.6	UserFPGA Internal Bus と SpaceWire FPGA、SpaceCube の関係 [9][3]	15
2.7	SpaceWire IF 搭載 DIO Board	15
2.8	DIO Board ブロック図	15
2.9	SpaceWire IF 搭載 FADC Board	16
2.10	FADC Board ブロック図	16
2.11	SpaceWire IF 搭載 Router	16
2.12	Router ブロック図	16
2.13	VHDL によるロジック・ゲート例	18
2.14	SpaceCube の写真	19
2.15	SpaceWire Gigabit Ethernet	20
2.16	メモリマップド I/O	20
2.17	PoGOLite 検出器が搭載される気球ゴンドラ [3]	22
2.18	PoGOLite による偏光測定の概念 [3]	23
2.19	PoGOLite 検出器の模式図 [3]	23
2.20	パスファインダーフライトにおける PDC ユニットの写真	23
2.21	PDC の 1 ユニット図 [4]	24
2.23	PDC の各シンチレータでの波形 [4]	25
2.22	PoGOLite 検出器の断面図 [4]	25
2.24	パスファインダーフライトにおける SAS ユニットの写真	26
2.25	次期 X 線天文衛星 ASTRO-H 完成予想イラスト [25]	27
2.26	ASTRO-H 開発スケジュール [25]	27
2.27	ASTRO-H の各観測装置に要求された観測エネルギー帯域 [25]	28
2.28	HXI の構造 [3]	30
2.29	SGD の構造 [7]	31
2.31	コンプトンカメラの原理 [3]	31
2.30	SGD 観測帯域における Si と CdTe の光電吸収、コンプトン散乱反応断面積 [33]	32
2.32	すざく衛星 HXD で実証された井戸型 BGO アクティブシールドによるバックグラウンドの低減手法。[21][22]	33
2.33	HXI における信号処理部コンポーネント [23]	34

2.34	HXI/SGD の APD 信号処理	35
3.1	PoGOLite 全体接続図	37
3.2	PoGOLite PDC 部におけるトリガ信号の動向 [17]	37
3.3	FADC Board で取得した波形の例	38
3.4	PoGOLite 波形保存ブロック図	39
3.5	波形による FastPeak と SlowPeak の違い	40
3.6	PoGOLite における保存した波形のデータフォーマット (1 波形:110 byte)	41
3.7	保存した波形をイベント選択した 2 次元ヒストグラム [4]	42
3.8	1 光子におけるコンプトン散乱と光電吸収の波形 [17]	43
3.10	図 3.3 の ch.0 波形を拡大。スレッシュルド判定、ヒストグラム波高値はこの Pulse Height から求める。	43
3.9	PoGOLite 波形保存クロックタイミング	44
3.11	PoGOLite ヒストグラム生成ブロック図	45
3.12	PoGOLite での一つの PMT におけるノイズヒストグラム。700 と 850 ch. に見られるピークは ADC の上限値に対応している。	46
3.13	PoGOLite ヒストグラム生成クロックタイミング	47
3.14	PoGOLite 開発の初期段階におけるセットアップ図	48
3.15	FADC Board における PDC/SAS 共用回路	48
3.16	PoGOLite での PDC/SAS/中性子シンチレータを全て含む計 92 本の PMT のノイズヒストグラム	50
3.17	シミュレーションによるバックグラウンドスペクトル [16][4]	51
3.18	高速中性子領域における $^3\text{He}(n, p)$ と $^6\text{Li}(n, \alpha)$ の反応断面積の図。横軸に中性子のエネルギー [eV]、縦軸は反応断面積 (バーン) である。[8]	53
3.19	BGO 結晶、LiCAF(Eu/Ce) 結晶写真 [20] と中性子シンチレータ構造図。フォスイッチ構造のため、BGO 結晶と LiCAF 結晶で波形弁別を行う必要がある。	54
3.20	BGO と LiCAF(Eu) シンチレーション光波形 (FADC Board のプリアンプ出力)[20]	54
3.21	BGO と LiCAF(Ce) シンチレーション光波形 (FADC Board のプリアンプ出力)[20]	54
3.22	中性子シンチレータのための FPGA ブロック図	55
3.23	同じ波形を入力した場合の ch. 別波形処理の様子	56
3.24	BGO と中性子シンチレータ LiCAF(Eu) のブランチの図 [20]	57
3.25	BGO と中性子シンチレータ LiCAF(Eu) のイベントセレクションヒストグラム [20]	57
4.1	APMU での VETO 信号、ヒストグラム生成ブロック図	58
4.2	移動平均フィルタのブロック図	60
4.3	一般的な移動平均フィルタの遅延器数を変えた場合の周波数特性 ($f_s = 1\text{MHz}$)	61
4.4	差分フィルタのブロック図	61
4.5	差分フィルタの遅延器数を変えた周波数特性 ($f_s = 1\text{MHz}$)	62
4.6	VETO 信号出力効率の測定で用いるテストパルス波形 ($T_{\text{rise}} = 0.5\text{usec}$, $T_{\text{fall}} = 50\text{usec}$, プリアンプ 580K 出力, $f_s = 2.35\text{MHz}$)	63
4.7	VETO 信号出力効率の試験における Trigger Module の全体図	63

4.8 図 4.7 の Discri 部における Trigger Module スレッショルド判定プロセス	64
4.9 差分フィルタ (Tap=3) 周波数特性	64
4.10 差分フィルタを通した波形	64
4.11 差分+移動平均フィルタ周波数特性(赤:移動平均フィルタ、緑:差分フィルタ、青:移動平均と差分フィルタの合成)	65
4.12 図 4.11 の赤線フィルタを通した波形	65
4.13 図 4.11 の青線フィルタを通した波形	65
4.14 テストパルスを用いた VETO 信号の出力効率測定セットアップ	65
4.15 500 Hz リピートにおけるスレッショルドを変えた場合の出力効率	66
4.16 10 kHz リピートにおけるスレッショルドを変えた場合の出力効率	66
4.17 移動平均 (Tap=3) フィルタ有無でのランダムパルス出力効率	67
4.18 連続パルス時の検出効率の測定セットアップ	67
4.19 移動平均 (Tap=3) フィルタ有無での連続パルス検出効率	68
4.20 FIR フィルタのブロック図	70
4.21 FIR BPF(透過帯域 100-300 kHz) 遅延器の数を変えたときの周波数特性 ($f_s = 1.17\text{MHz}$)	71
4.22 FIR BPF(通過帯域 100-300 kHz) の遅延器数を変えた場合の周波数特性	72
4.23 FIR BPF 遅延器数の違いによる残差二乗和	72
4.24 FIR BPF(通過帯域 100-300 kHz) の係数精度を変えた場合の周波数特性	73
4.25 FIR BPF 係数精度の違いによる残差二乗和(横軸:フィルタ係数精度 2^n の n、遅延器数別にプロットしている)	74
4.26 BGO+APD(Gain100,-15deg) HICPreAmp,ADC128S102 による ^{241}Am 信号(緑)とノイズ(赤)の周波数成分	75
4.27 ClearPulse Shaper 4417 ShapingTime 別 周波数特性	76
4.28 ORTEC Shaper 571 ShapingTime 別 周波数特性	76
4.29 CP と ORTEC Shaper 周波数特性 比較	76
4.30 係数の対称性の違いによる波形の変化(赤:元となるプリアンプ信号、緑:偶対称係数フィルタに通したときの波形、青:奇対称係数フィルタに通したときの波形)	77
4.31 係数が偶対称の場合(緑)と奇対称の場合(青)	78
4.32 係数が偶対称と奇対称の場合による周波数特性(BPF 100-300 kHz)	78
4.33 FIR フィルタで Shaper(式 4.15) を再現し、ピーク位置を変化させた場合の周波数特性	79
4.34 周波数特性にオフセットを与えるための FIR フィルタの係数 g_n	80
4.35 周波数特性にオフセットをつけるための係数 h_n (色によってオフセットの度合が異なる)	80
4.36 図 4.35 の各係数における周波数特性(色は図 4.35 と対応している)	80
4.37 FPGA での波形取得時のデータフォーマット	81
4.38 ^{241}Am を BGO+APD(Gain100,-15deg) に照射、HICPreAmp 出力信号を FADC Board で取得した波形 ($f_s = 1.17\text{MHz}$)	81
4.39 ソフトウェアでの FIR フィルタの検証に用いる各フィルタ係数における周波数特性	82
4.40 図 4.38 に FIR2bit フィルタを通した波形	83

4.41 図 4.38 に FIR BPF 25-150kHz を通した波形	83
4.42 図 4.38 に FIR Analog1 60kHz を通した波形	83
4.43 図 4.38 に FIR Analog2 60kHz を通した波形	83
4.44 ソフトウェアによる FIR フィルタを用いたヒストグラム比較 (BGO+APD ^{241}Am -15deg)	84
4.45 ADC128S102 SerialTiming Diagram[29]	87
4.46 SPI 通信ブロック図	87
4.47 SPI Master In Slave Out Mode	88
4.49 ADC128S102 評価基板表面 (コネクタ、受動素子実装側)	88
4.50 ADC128S102 評価基板裏面 (パターン、能動素子実装側)	88
4.48 ADC128S102 評価基板回路図。ADC 素子の前段にゲイン 1 の反転アンプを、ディジタル入出力ポートにはバッファを配置。	89
4.51 ADC128S102 によって取得した正弦波 (10 kHz) 信号波形	90
4.52 $26=b'11010'$ のビットシフト	91
4.53 ビットシフトを利用した乗算方法	91
4.54 三角波と FPGA 上で FIR フィルタ [係数 1,0,-1] を通した後の波形	92
4.55 FIRAnalogue2 Peak:60kHz Offset150% Tap16 で係数が 2^{20} 小数 (赤) と 2^8 整数 (緑) での周波数特性	93
4.56 BGO+APD PreAmpOut ^{241}Am 波形 (下) とソフトウェアで FIR8bit フィルタを通した波形 (上)	93
4.57 図 4.56 の波形から得られた ^{241}Am 59.5 keV ヒストグラム (左:フィルタ前、右:FIR8bit フィルタ後)	94
4.58 ADC128S102 評価基板を用いた線源データ取得環境セットアップ [BGO+APD(Gain 100,-15 度)]	95
4.59 BGO+APD(Gain100,-15 度) ADC128S102 を用いた ^{241}Am 波形 [フィルタ無し] . .	95
4.60 HICPreAmp 信号を ADC128S102(赤) とオシロスコープ (緑) で取得した波形	95
4.61 BGO+APD(Gain100,-15 度) ADC128S102 を用いた ^{241}Am 波形 [FIR 2bit フィルタ] .	96
4.62 BGO+APD(Gain100,-15 度) ADC128S102 を用いた ^{241}Am 波形 [FIR 8bit フィルタ] .	96
4.63 FIR8bit フィルタ (緑) と Ortec Shaper ($\tau=2\text{usec}$)(赤) の周波数特性	96
4.64 FIR8bit フィルタ (緑) と Ortec Shaper ($\tau=2\text{usec}$)(赤) による HICPreAmp 波形の整形	96
4.65 BGO+APD(Gain100,-15 度) ADC128S102 を用いて FPGA 上で取得した ^{241}Am ヒストグラム	97
4.66 BGO+APD(Gain100,-15 度) ADC128S102 を用いて FPGA 上で取得した ^{137}Cs ヒストグラム	97
A.1 各フィルタの通過帯域、除去帯域のようす	101
A.2 アナログ Low Pass Filter	101
A.3 図 A.2 において $C=500 \text{ pF}$ 、 $R=500 \Omega$ の場合の周波数特性	102
A.4 ディジタル フィルタの例:移動平均フィルタ	103
A.5 2010 年東広島市気温 (1 点/1 時間) [32]	104

A.6	2010 年東広島市の気温(1日平均)	104
A.7	アナログ信号の AD 変換の様子	104
A.8	デジタルフィルタブロック図基本コンポーネント	105
A.9	移動平均フィルタのブロック図(図 A.4)	105
A.10	移動平均フィルタ周波数特性 (Tap=1, $f_s = 1\text{MHz}$)	108
A.11	移動平均フィルタの一般的なブロック図	108
A.12	一般的な移動平均フィルタ周波数特性 ($f_s = 1\text{MHz}$)	109
A.13	差分フィルタの一般的なブロック図	110
A.14	一般的な差分フィルタ周波数特性 ($f_s = 1\text{MHz}$)	110
A.15	共振器ブロック図	111
A.16	共振器の周波数特性 ($f_s = 1\text{MHz}, F_0 = f_s/6$)	112
A.17	ノッチフィルタブロック図	112
A.18	ノッチフィルタの周波数特性 ($f_s = 1\text{MHz}, F_0 = f_s/6$)	113
A.19	FIR フィルタブロック図(直接形)	115
A.20	FIR BandPassFilter (透過域 100 - 300 kHz, $f_s = 1.17\text{MHz}$)	116
A.21	理想的な Low Pass Filter	117
A.22	FIR フィルタ係数と窓関数との関係	118
A.23	FIR LPF 遅延器の数を変えたときの周波数特性	119
A.24	FIR LPF 遅延器の数を変えたときの周波数特性 ($f_s = 1.17\text{MHz}$)	120
A.25	FIR HPF 遅延器の数を変えたときの周波数特性 ($f_s = 1.17\text{MHz}$)	121
A.26	FIR BPF 遅延器の数を変えたときの周波数特性 ($f_s = 1.17\text{MHz}$)	122
B.1	NS 社による ADC128S102 典型回路 [29]	123
B.2	ADC 評価用基板 アナログ部回路	123
B.3	LTspice による ADC Board アナログ回路シミュレーション	124
B.4	ADC Board アナログ部波形シミュレート(緑:Vin、赤:Vout)	125
B.5	過大入力時における反転アンプの抵抗別波形	125
B.6	ADC Board アナログ部周波数特性シミュレート	126
B.7	ADC Board デジタル部参考 FADC CMOS(In/Out) [28]	127
B.8	ADC128S102 評価基板 PCBE	127

第1章 序論

1.1 背景

宇宙では我々の想像もつかないような現象が多く存在する。例えば超新星残骸や銀河団、ブラックホール、ガンマ線バーストなどであり、これら高エネルギー天体からは硬X線、ガンマ線領域のエネルギーを持つ放射が観測されている。このような高いエネルギーを持つ放射は逆コンプトン散乱、シンクロトロン放射、制動放射など、非熱的な過程で相対論的エネルギーまで加速された粒子からの放射であることが分かってきている。我々の住む環境では粒子が十分に相互作用することでエネルギーは熱的平衡状態に至るが、宇宙では粒子密度が希薄なため粒子に加速メカニズムが働いたときに高いエネルギーを粒子が得、非熱的現象の放射を行う。しかしながら加速メカニズムについてははっきりと分かっておらず、さらに詳しい観測が求められている。

非熱的な成分は熱成分が弱くなる 10 keV 以上の領域で観測を行うことが望ましい。10 keV 程度からそれ以下では集光が可能であり、天体信号も大きいため S/N 比が高くなる。過去に打ち上げられた ASCA, Chandra, Newton などの X 線天文衛星では 10 keV 以下の軟 X 線を観測するうえでその効果を発揮した。TeV 以上の帯域では天体信号は小さいが二次宇宙線である大気チレンコフ光の検出技術の向上によってその大気シャワーの広がりの違いから陽子と TeV ガンマ線を区別することができる。さらに MeV、GeV 帯域では電子・陽電子対生成を利用した信号選択が可能な Fermi 衛星などにより、高い S/N 比を実現している。一方、10 keV 以上、MeV 程度までの硬 X 線・軟ガンマ線帯域では集光が難しく (S が小さい)、コンプトン散乱と光電吸収が混交するため (N が大きい) S/N 比が悪い。そのため今まで十分な観測が行われてこなかった。

未だに調査の余地が多く残るこの硬 X 線・軟ガンマ線帯域を観測するため、2011 年夏にパスファインダーが放球予定の気球搭載型の硬 X 線偏光検出器 PoGOLite と 2013 年度に打ち上げ予定の次期 X 線天文衛星 ASTRO-H(旧称 NeXT) 計画が進行中である。PoGOLite は日欧米の国際協力のもとで開発が行われており、日本からは広島大学、東京工業大学、山形大学、宇宙航空研究開発機構・宇宙科学研究所 (JAXA/ISAS) などが参加している。硬 X 線・軟ガンマ線帯域での偏光観測は高エネルギー天体の幾何学的な構造や磁場構造を調べるために重要であるが X 線の粒子性が強くなるために観測が難しく、これまでも 2 例しか有意に観測されていない。PoGOLite では 25 - 80 keV という今まで困難であった硬 X 線・軟ガンマ線帯域での偏光を検出する。ASTRO-H 計画には JAXA/ISAS を中心に、国内外の多数の大学や研究機関が参加している。ASTRO-H は (1) 硬 X 線における世界初の撮像分光、(2) マイクロカロリメータによる 1 eV まで分解可能な超高分解能分光観測、(3) 0.3 - 600 keV までと 3 枠以上の広帯域において従来の 1 枠程度感度の良い観測を目標としている。これを実現するために硬 X 線撮像分光検出器 (HXI)、軟ガンマ線検出器 (SGD)、軟 X 線精密分光検出器 (SXS)、軟 X 線撮像分光検出器 (SXI) の 4 つの検出器を搭載している。

S/N 比の低いこの帯域でより良い観測を行う上で重要なことは宇宙背景放射、陽子、電子・陽電子などの一次宇宙線、中性子などの二次宇宙線、SAA(South Atlantic Anomaly) など、バックグラ

ウンドを徹底して除去することである。そのため、PoGOLite と ASTRO-H では「すざく」衛星で実績のある、BGO 結晶を用いたアクティブシールドを用いる。アクティブシールドとは視野外以外をシンチレータで覆ったもので鉛などを用いるパッシブシールドとは違ってシールドそのものを検出器としており、シールドを突き抜けて主検出部へと入射するような強い信号の場合はシールド検出器の VETO 信号と主検出部との反同時係数を取ることでデータの選別が可能となる。またシールドそのものが検出器であるため、シールドに入射した信号のモニタも可能となる。ASTRO-H ではアクティブシールドでガンマ線バーストなどの突発天体にも対処できるようにしている。また ASTRO-H では BGO 結晶シンチレーション信号の処理にデジタルフィルタを用いることで効率の良い VETO 信号の出力を行う。このためアクティブシールドに合わせたデータ処理、デジタル処理など衛星に搭載することを考慮した読み出しシステムの開発が進められている。

天体信号が常に変動する衛星では多量のデータを高速で処理する必要があり、それに伴ってデータを衛星上で他の装置に高速に転送するための通信インターフェイスが必要となる。衛星では観測機器、処理装置、通信機器などがそれぞれ個別に製作される。従来、これらの機器間での通信規格は衛星ごと、機器ごとに開発されていたため異なったものが混在していた。このことは衛星開発の長期化、コストの増大、信頼性の確保が難しくなるなどといった問題を生じさせていた。このような状況を打開するべく、現在衛星上の各種処理装置等を結ぶ通信規格の統一が進められている。統一通信規格として SpaceWire が提唱され、世界的にその規格策定が進められている。日本からは JAXA/ISAS を中心に SpaceWire 規格の策定に参加している。SpaceWire は高速通信を行える一方で、可変ビットレートに対応するなど、様々なスケールの装置間での柔軟な接続を実現する。さらに通信インターフェイスを SpaceWire に統一していれば、我々は観測装置の開発だけに集中することができる。PoGOLite や ASTRO-H ではこの SpaceWire を搭載することが決定しており、データ処理、通信、コマンド転送などのようなシステムを SpaceWire へと対応させることが要求される。

1.2 研究の目的

PoGOLite では SpaceWire IF 搭載の FADC Board 12 枚と DIO Board を用いて偏光を観測する。偏光はコンプトン散乱の異方性を調べるものであり、これを実現するためには天体信号が主検出部に入力されれば全ボードを同時に動作させてどの方向に散乱されたかを調べる必要がある。またアクティブシールドでは VETO 信号を出力させ、バックグラウンドのモニタとしてパルスハイヒストグラムを取得する。これらは先行研究においてハードウェア記述言語 VHDL を用いて独立に開発された。ただし、主検出部では波形を保存する機能、ヒストグラムを取得する機能、波形弁別信号を出力する機能が必要であり、アクティブシールド部ではヒストグラム取得機能、波形弁別信号を出力する機能があればよいため、両者の共用回路を作り、アクティブシールド部において波形保存機能をオフにすれば、一つの回路で両者に対応することができる。こうすることで回路のデバッグも一回路分ですみ、信頼性も向上する。そこで私はこの共用回路を作成し、実機でも問題無く動くかどうか確認した。

また PoGOLite パスファインダーフライトでは世界で初めて LiCAF 結晶による中性子シンチレータを搭載する。バックグラウンドとなる宇宙線のうち、中性子は荷電粒子ではないため電磁気的な相互作用を起こしにくい。そのためアクティブシールドでは十分防げず、重大なバックグラウ

ンドとなる。しかし、そのフラックスをモニタすることができれば中性子バックグラウンドの見積りを行える可能性がある。中性子シンチレータは他のシンチレータとは減衰時定数が異なり、またこのユニットは偏光観測とは異なるため他のユニットとは独立に動作する必要がある。よって私は他のボードとは独立に動作し、かつ中性子シンチレータの波形に合わせたシステムを構築して、中性子照射実験において正常に動作するか確認した。

ASTRO-H のアクティブシールドではシールドとなる BGO 結晶に対してアバランシェフォトダイオード (APD) 素子で検出する。この信号は 1 MHz でサンプリングされ、後段の FPGA でデジタル処理を行われる。FPGA では主検出部への VETO 信号の出力とバックグラウンドのモニタのためヒストグラム生成などが行われる。より良い感度で観測を行うためにはこの VETO 信号がより低いスレッショルドまで感度を持つとよい。そのために FPGA でのデジタルフィルタを用いることが決定している。デジタルフィルタを用いることで従来のアナログフィルタに比べて回路の実装面積を抑えることができ、またフィルタをパラメタ化することでフィルタの種類を変えることができる。しかし VETO 信号出力のためには限られた時間での処理を行わなければならず、加えて FPGA の容量の問題によって複雑なフィルタを用いることができない。さらに衛星観測で BGO+APD 信号に対してデジタルフィルタを実装したことなく、どのようなフィルタを用いればよいのか、ということが分からぬ。そのため私は実機に近い環境を構築した上でノイズと信号成分の周波数成分を取得し、どのようなフィルタを通せばよいのかを調べてそれに合わせたフィルタを構築して実際に効果があるのかを VETO 信号出力、パルスハイトヒストグラム取得の両方で検証した。

第2章 SpaceWire

科学衛星では観測装置をはじめ、姿勢制御装置、通信装置、各種センサ、データ処理装置など、様々な機器の間で情報をやりとりする必要がある。現在、衛星上の装置間通信のためのハードウェア開発は、それぞれの装置の機能設計と同時に行われることが多い。こうすることでその装置に適した通信装置を開発することができるからである。しかし、それぞれの装置が独自の通信規格を採用するために各機器間通信のためのハードウェアを新たに作る必要がある。そのため時間がかかり、コストも増大する。さらに、通信のためのハードウェアとそれぞれの装置の機能を実現するためのハードウェアが互いに影響を及ぼし合い、信頼性の低下にもつながる。また衛星ごとに設計をし直すため、過去の資産も継承されない。このような問題を打開すべく、従来より衛星上の装置間通信規格の共通化が考えられてきた。

衛星上の装置間通信の統一規格として、現在最も有力な規格が SpaceWire である。SpaceWire はもともと、欧州宇宙機関 (ESA) により IEEE1355 をベースに宇宙用の標準規格として提案された。現在、さらに使いやすいうように改良、あいまいさの明確化を行い、組み込み機器間の柔軟な接続を目指している。SpaceWire の規格策定には、ヨーロッパをはじめ世界中の宇宙機関や企業が参加しており、日本からも JAXA/ISAS や大阪大学を中心に参加している。

2.1 SpaceWire の特徴

SpaceWire はデータ転送レートが可変なため、さまざまな機器に柔軟に対応できる。1 ラインあたり、2~400 Mbps という高速転送をサポートする。また、伝送には LVDS(Low Voltage Differential Signaling) 方式を採用している。低電圧での差動伝送方式のため、高速通信でも低消費電力、低ノイズでの伝送が可能である。また、DS-LINK を採用し、Data 信号の読み取りに必要な Clock 信号を伝送する Clock ラインを持たないが、Data 信号と一緒に Strobe 信号を伝送することで受信側で Clock 信号を再現することが可能である。図 2.1 にあるように Data と Strobe の XOR をとることで Clock を再現できる。

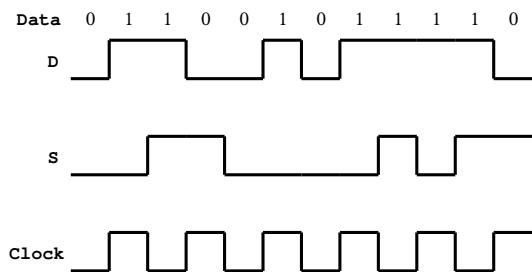


図 2.1: Data-Strobe 信号

SpaceWire は全二重方式を採用しているため、入力と出力を同時に使うことができる。またケーブルは Data、Strobe が入出力合わせて 2 組で構成されている。それぞれの信号線は強くより合わせたツイストペアを構成しており、ノイズに強い構造をとる。また差動信号ペア間や Data-Strobe 信号間の歪みも小さい。さらに LVDS 方式よりケーブルの長さを最大 10 m 以上の長さまで延長することが可能となっている。使用されるケーブルコネクタは 9 ピン D-sub コネクタと同じ形である。このコネクタの型は宇宙空間で使用するのに適任である。

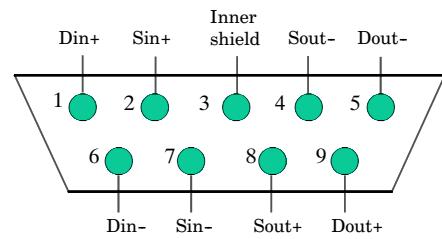
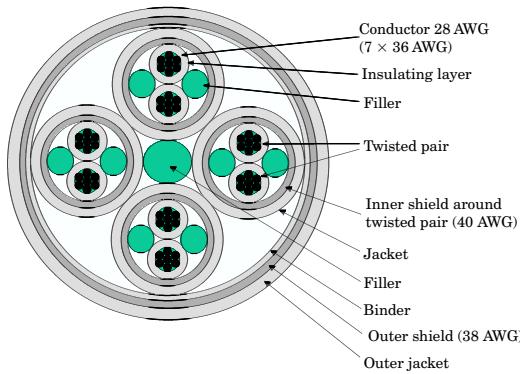


図 2.2: ケーブルの構造 [10]

図 2.3: コネクタ・ピンアサイン [10]

SpaceWire はコンセントに挿すようにノードを組み合わせることができる。そのためそのトポロジはスター型、リング型など様々な形を取ることが可能である。これにより、データネットワークが切れた場合でも別経由に切替えることでデータの転送を可能にする。図 2.4 にあるネットワークが切断された際の特定ノードへの迂回パスを記載する。

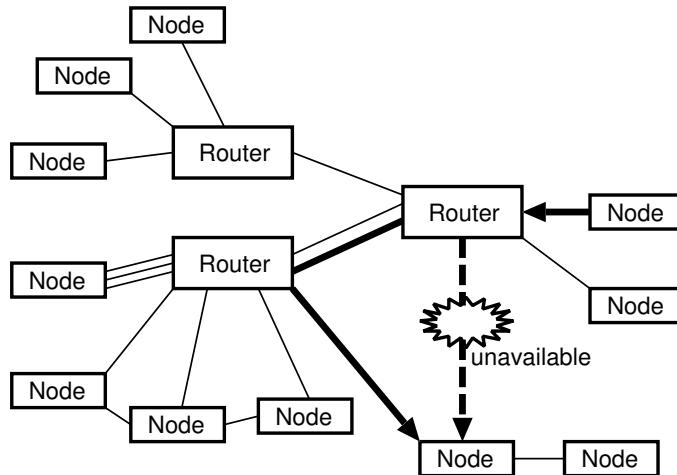


図 2.4: SpaceWire における特定ノードへの迂回パス

2.2 SpaceWire IF 搭載 Board

現在我々は SpaceWire プロトコルを搭載し、プログラマブルな LSI である FPGA(Field Programmable Gate Array)を実装したボードを用いて開発・実験を行っている。このボードは大きく分けて DIO Board と FADC Board の二種類に分かれる。各ボードでの詳しい説明は後に行う。DIO Board と FADC Board に共通してボード上に FPGA が二つ搭載されており、一つが SpaceWire 規格を用いた通信を司るためのチップである SpaceWireFPGA(Xilinx 社製 XC3S1000)で、もう一つがユーザの任意の回路を製作するためのチップである UserFPGA (XC3S400 か XC3S1000) である。SpaceWireFPGA の IP コアはボードを製作しているシマフジ電機によって開発されている。UserFPGA のレジスタへの Read/Write は SpaceWireFPGA を通じて行う。SpaceWireFPGA における UserFPGA のメモリ配分は以下の図 2.5 のように x'0101-0000' から x'0101-FFFF' までの 64 kB の領域が与えられている。

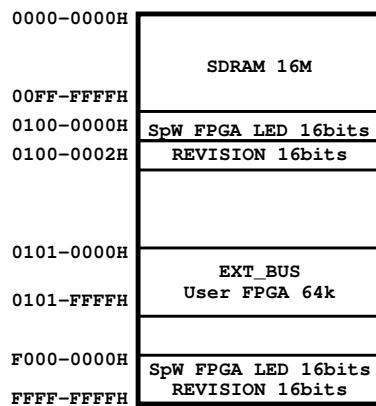


図 2.5: SpaceWire FPGA Memory Map の例 [26][27]

ボード上には 16 MB の SD-RAM も搭載されている。SpaceWireFPGA との双方向通信を利用してアクセスが可能であり、データの Read/Write が行える。各ボードには CMOS デジタルポート、LVDS デジタルポートが実装されており、FPGA へ任意の回路を構築することであらゆる実験への対応が可能である。その FPGA への回路実装はチップ開発元の Xilinx 社提供である統合開発環境 ISE にてハードウェア記述言語である VHDL を用いて行う。ISE は回路の論理合成、配置配線、バイナリファイル生成まで一貫して行うことが可能である。FADC Board や DIO Board の UserFPGA における SpaceWireFPGA とのバス通信、ADC 制御などは東京大学の湯浅氏、宇宙航空研究開発機構 (JAXA) の小高氏、広島大学の田中氏によって雛型が完成されているため、我々は自身の作りたいモジュールのみを作成すればよい。図 2.6 に UserFPGA におけるブロック図を示す。ユーザが作りたいモジュールはここでは UserModule1、UserModule2 と示している。SpaceWireFPGA との通信は ExternalBusIFtoSpWFPGA が行い、ユーザは各 UserModule に BusIFModule を接続するだけで容易に SpaceWireFPGA との通信が可能である。

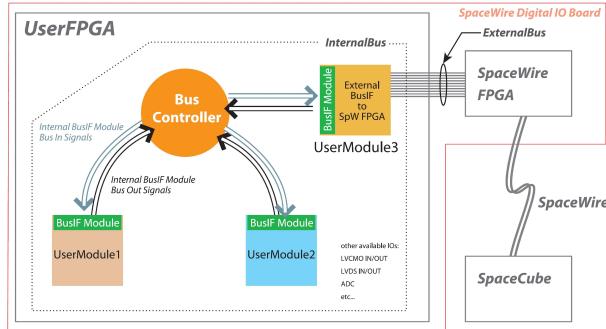


図 2.6: UserFPGA Internal Bus と SpaceWire FPGA、SpaceCube の関係 [9][3]

ボード上には 2 つの SpaceWire Port が与えられている。各ポートにはロジカルアドレスと呼ばれる 0~255 までの番号が割り振られており、SpaceWire 通信の際にはこのアドレスを用いることで任意のノードへのアクセスが可能である。

2.2.1 DIO Board

DIO Board は汎用ディジタルボードとして CMOS ディジタルポートが In/Out それぞれ 8 ポート、LVDS ディジタルポートが In/Out それぞれ 12 ポート搭載されている。以下の図 2.7 に DIO Board の写真を、図 2.8 に簡単なブロック図を搭載する。



図 2.7: SpaceWire IF 搭載 DIO Board

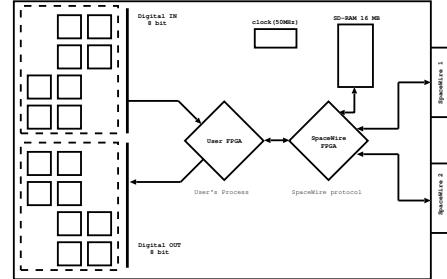


図 2.8: DIO Board ブロック図

2.2.2 FADC Board

FADC Board には CMOS ディジタルポートが In/Out それぞれ 3 ポート、LVDS ディジタルポートが In/Out それぞれ 4 ポート搭載されている。またアナログ入力部が 8 ポートあり、ADC には Texas Instruments 社の ADS5271 が実装されている。この ADC は 8 チャンネル入力で最高 50 MSPS を達成できる。FADC Board では上述したように UserFPGA で ADC 制御を行う。以下の図 2.9 に FADC Board の写真を、図 2.10 に簡単なブロック図を搭載する。



図 2.9: SpaceWire IF 搭載 FADC Board

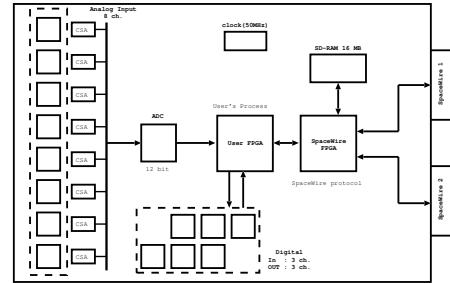


図 2.10: FADC Board ブロック図

2.2.3 Router Board

Router Board とは複数の SpaceWire Board を扱うときにそのネットワーク間の相互接続のためのものである。DIO Board や FADC Board と違い、FPGA には Xilinx 社製 XC4VLX60 が搭載されている。一つのルータで 8 つの SpaceWire Port が搭載されている。以下の図 2.11 に Router Board の写真を、図 2.12 に簡単なブロック図を搭載する。

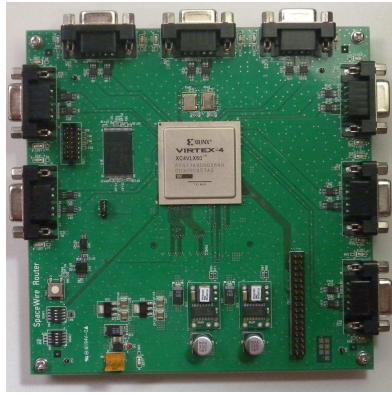


図 2.11: SpaceWire IF 搭載 Router

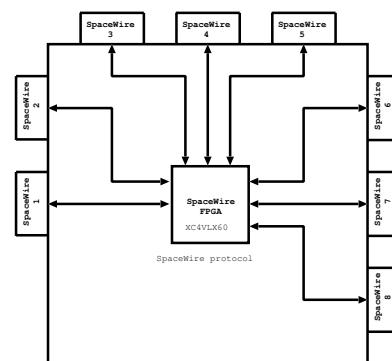


図 2.12: Router ブロック図

2.3 FPGA

システムを開発する際、そのおおまかな仕様設計から始め、より詳細部分へと設計を行う。つまり、抽象度の高いアーキテクチャ・レベルから詳細なゲート・レベルへと段階を踏んで設計を行う。初期のころは真理値表、論理式などを用い、システムの目的に合わせて標準ロジック IC と呼ばれる NOT や AND 回路を含んだ IC を並べていた。しかし、この手法だと実装面積が大きくなる、時間がかかる、他者には分かりづらい、設計回路が正しく動作するかも基板上に組み上げた回路で実際に動作させてみるしか方法がないなどといった様々な問題があった。そこで、特定用途向けの集積回路である ASIC (Application Specific IntegratedCircuit) が使われる。ASIC を作ることで、回路を一つの集積回路に納めてしまい、省面積化、高速化を実現することができる。しかし、このような ASIC は半導体工場で製造されるため、開発に時間がかかり、工場の設備を使用するために莫大なコストもかかる。そのため、ASIC の開発段階での試作や、少量の ASIC しか使用しない場

合、その度に半導体工場で LSI を生産するのは非現実的である。また、製造された LSI の中身は書き換えることができないため、後から回路を修正したりすることは不可能である。そこでユーザが自由に回路を書き換えることができるデバイスとして開発されたのがFPGAである。

FPGA は ASIC のように工場に注文して作成する形式ではなく、回路を設計する設計者の手元で作成した回路を書き込むことができる LSI である。書き込むことをダウンロードと呼ぶが、ダウンロードに必要な機材は FPGA の他にダウンロードケーブルと PC ぐらいなので簡単に実装することができる。似たようなもので回路を書き込むことが出来るデバイスで CPLD(Complex Programmable Logic Device) というものがあるが、一般に FPGA の方が大規模回路に適している。FPGA への回路作成には後述する VHDL を用いている。

2.3.1 VHDL

少し前まではハードウェアを目的の動作に向けて記述する際、真理値表、論理式などを用いて AND や OR 回路を組み合わせたゲートレベルでの設計が主流だった。つまり、ある処理を作動させようと思ったらどのように処理をするかを'0' と '1' を用い、更に何かしらの条件を考えるときでさえ AND や OR などの論理式を考えなくてはならなかった。これでは高度な処理を記述しようとすればするほど回路が複雑になり、設計にかなりの時間がかかる。誤作動が起きたときにエラーの判別がしにくくなり、更に設計者以外にはその回路の意図を掴むのが難しくなる。

これを解消するために、著しく発達したコンピュータを利用して回路の接続関係を記述する言語、ハードウェア記述言語 (HDL:Hardware Description Language) が利用されるようになった。HDL ではゲートレベルでの記述以上に抽象度の高い記述を行うことができる。つまり C 言語のようなコンピュータ言語を操るかのように回路を記述することが出来る。これにより回路を動作に合わせて設計しやすくなり、他人にも動作内容が分かりやすくなる。誤作動が起きたときも対処しやすく、回路変更も容易となる。

現在最も普及している HDL には VHDL と Verilog HDL がある。本研究では文法的には難しいが高い記述能力を持つ VHDL(VHSIC HDL) を用いて回路設計を行った。VHDL は、米国国防総省の VHSIC(Very High Speed Integrated Circuit) 委員会によって提唱されて開発された。当時、国防総省向けの ASIC を開発する際ゲートレベルでの開発を行っていたため開発に長い時間がかかり、設計を開始したころの最新半導体部品を基準に開発していたとしても設計が完了するころには既にそれ以上の部品が出来ているため完成品が時代遅れとなってしまうという問題があった。そのため、より抽象度の高く、開発終了時にその時点での最新の部品が使えるような設計手法を開発する必要があった。これより、1981 年に提唱、1986 年公開されたのが VHDL である。その後文法の改訂作業を経て、IEEE(米国電気電子技術者協会) により標準化され、VHDL は世界の標準 HDL として認識されている。現在、LSI のような大規模設計を行うことができるようになったのも VHDL などの高度のハードウェア記述言語のおかげとも言える。

VHDL は様々なレベルでの記述が可能である。抽象度の高いシステムの概要を記述したり、下のレベルである装置間信号の記述、モデル化を行ったモジュールの記述といったことも可能である。更に下のゲートレベルでも記述を行うことが出来る。VHDL でロジック回路の生成を行うためには RTL(Register Transfer Level) で記述する。このレベルで記述された場合、ロジック回路を簡単に論理合成して作成することが可能である。以下に VHDL によるロジック・ゲートの記述、

図 2.13 にその論理合成の結果を載せる。

```

library IEEE;
use IEEE.std_logic_1164.all;

entity sample is
    port (    A : in std_logic;
              B : in std_logic;
            Out1 : out std_logic;
            Out2 : out std_logic);
end sample;

architecture RTL of sample is
    signal sig : std_logic;

begin
    sig <= A nor B;
    Out1 <= not sig;
    Out2 <= A and sig;
end RTL;

```

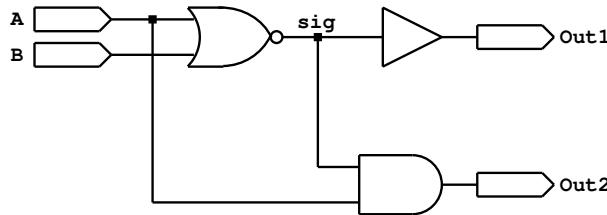


図 2.13: VHDL によるロジック・ゲート例

2.4 SpaceCube と Gigabit Ethernet

2.4.1 SpaceCube

SpaceWire 通信規格を用いた実験・装置開発を行う上で SpaceWire 規格に対応したコンピュータが開発された。それが SpaceCube と呼ばれるもので本体の大きさはおよそ 5 cm 角と小型化されている。SpaceCube には外部接続で SpaceWire ポートを三つ実装されており、CPU として MIPS アーキテクチャを持つ NEC 製の VR5701(300MHz) を搭載、PCI バスに LAN、USB を搭載したボードを内蔵する。OS は Linux と T-Engine が対応する。通常はシリアルケーブルを用いてシリアルコンソールで操作を行うが、キーボード、マウス、ディスプレイを接続すれば普通の PC としても起動することもできる。小型ながら高機能、低コストでの SpaceWire 動作検証を行うことが可能。以下の表 2.1 に SpaceCube の性能表を、図 2.14 に SpaceCube の概観写真を載せる。

CPU	VR5701 200 MHz / 250 MHz / 300 MHz
Flash ROM	16 MB
DRAM I/F	DDR SDRAM 64 MB
INPUT/ OUTPUT	IEEE1355(SpaceWire),RTC,CF(True IDE),XGA(1024×768),USB1.1, LAN(100BASE),Audio(Stereo), 入出力 RS232C,JTAG I/F(Debug)
POWER	+5 V
SIZE	52 mm × 52 mm × 55 mm

表 2.1: SpaceCube の仕様



図 2.14: SpaceCube の写真

宇宙のような突然の現象などによるシステムの割り込みなどの場合、リアルタイム性に優れた OS の方が対処しやすくなる。これはリアルタイム OS が機器制御を目的としているため、個々の処理内容に優先度がつけられており常に優先度が高い処理から行われるように設計されているからである。人間が相手となる OS ではなく、機器制御という OS より、T-Kernel は衛星に有利といえる。しかし Linux の場合ではこれまで PC で行ってきた開発環境が有効活用できるため、全面的に T-Kernel が良いというわけではない。すなわち目的、用途に適した OS を選ぶ必要がある。

2.4.2 SpaceWire Gigabit Ethernet

SpaceWire は規格として 2~400 Mbps までサポートされているが、SpaceCube を用いる場合はそれを最大限まで利用することはできない。これは SpaceCube の CPU クロックがそもそも早くなく (200 - 300 MHz)、それによって SpaceCube 内で SpaceWire-To-TCP/IP 変換を行うことで最高転送速度に制限がかかってしまうためである。SpaceWire 通信クロックを 100 MHz にしていたとしても SpaceCube 内での処理を含めると 4 Mbps ほどしか達成できない。大量のデータを処理する際はこの転送速度では十分ではない。それを打開するため東京大学の湯浅氏とシマフジ電機が共同で開発したものが図 2.15 の SpaceWire Gigabit Ethernet(SpW-GbE) である。SpaceWire-To-TCP/IP 変換の専用機とすることで SpaceWire 通信クロックが 100 MHz のときに ~70 Mbps を達成する。SpaceCube にて操作をする際はドライバが必要だが、SpW-GbE は SpaceWire と TCP/IP の仲介役として特化しているためネットワーク IP を設定するだけですぐに使用が可能である。



図 2.15: SpaceWire Gigabit Ethernet

2.5 Remote Memory Access Protocol(RMAP)

PC 上のプログラムで何かのデータを参照したいというときは、CPU がデータが格納されたメモリからデータを取り出して実行する。このとき CPU はメモリのアドレス空間を参照してデータを取り出す。同じように外部機器との I/O 機能を参照するためにメモリアドレスとは別に I/O アドレスというものがあるが、この I/O アドレスはメモリアドレスよりも空間範囲が狭いため、メモリアドレス上に配置して疑似的に I/O アドレスのようにすることができる。このことをメモリマップド I/O と呼ぶ。

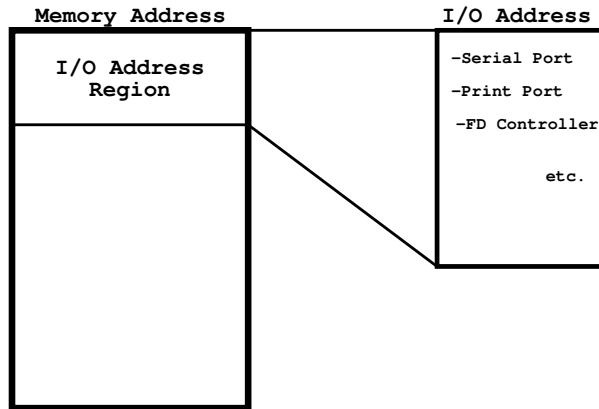


図 2.16: メモリマップド I/O

SpaceWire に接続された機器はこのメモリマップド I/O 機能が使える。そのため SpaceCube 側から見れば自身のメモリにアクセスするかのごとく SpaceWire の先の機器 (本研究では SpaceWire Board 上の SpaceWireFPGA メモリ) にアクセスすることが可能となる。これにより SpaceWire に繋がった機器に CPU が実装されていなくても制御通信が出来る。このような仕組みを定めたプロトコルのことを RMAP(Remote Memory Access Protocol) と呼ぶ。RMAP を用いれば CPU を持たない基板のメモリ情報を簡単にネットワークに組み込むことができる。そのため機器間での複雑なデータ変換のためのハードウェア、ソフトウェアを省略することができ、機器のハードウェアの省スペース、干渉によるデータの損失といった心配が減ることになる。

以上から簡単にネットワーク構成を組み立てることができ、また統一的な規格にすることでデータの信頼性、開発期間の短縮、機器間での通信プロトコルの変換機器を作成しなくてもよいといった様々なメリットがある。つまり開発を全く別々に行っていても通信規格を SpaceWire に統一しておけば簡単に接続でき、プロセッサが設置されていなくても SpaceWire と接続されていればその機器はリモートメモリアクセスによってその装置のレジスタをさながら自身のメモリを読み取るがのごとく知ることができる。このような特徴から、車載機器や通信機器業界からの関心も高く、民生機器への応用も期待される。

2.6 SpaceWire を用いた観測器器 1:PoGOLite

宇宙 X 線源が発見された 1960 年代以降、宇宙における X 線源やガンマ線源はそのスペクトルや光度の時間変化、X 線源の撮像を用いることでその特性が解明されてきた。X 線帯域における偏光の観測は、回転駆動型バルサの放射機構やブラックホール連星近傍の膠着円盤の磁場・幾何学的構造、高磁場中性子星・活動銀河核の特性の解明に貢献する。しかし、X 線・ガンマ線領域において電場(磁場)ベクトルの方向の偏りなどに起因する偏光の観測はほとんど行われていない。それは X 線領域になると光の粒子性が強くなり、電波や可視光のように電場ベクトルの観測が困難だからである。実際これまでには、かに星雲を対象に、1970 年代に OSO-8 衛星によるプラグ偏光計を利用して測定した偏光度 19.2 % (2.6 keV)/19.5 % (5.2 keV)、2008 年 INTEGRAL 衛星による 0.1 - 1 MeV における偏光度 $46 \pm 10\%$ の二例しか有意な偏光の検出例はない。

the Polarized Gamma-ray Observer Lite-weight version(PoGOLite) は、25 - 80 keV の硬 X 線・軟ガンマ線帯域における偏光観測を目的とする実験である。データ収集システムには FADC Board や DIO Board を用いるなど SpaceWire がベースとなっており、SpaceWire の実証試験も兼ねている。25 - 80 keV の帯域は大気による散乱・吸収の影響のため地上からの観測が行えないため、気球により上空 40 - 41 km に放球する。1/3 サイズのパスファインダーフライトが 2011 年にスウェーデンのキルナから放球予定である。波形弁別と徹底したバックグラウンドの除去により 100 mCrab で 10.5 % の偏光度の検出を目指している。PoGOLite の主な特徴を以下の表 2.2 に記載する。また図 2.17 に PoGOLite が搭載された気球ゴンドラ図を記載する。

重量:~1200 kg; 電力:200 W; サイズ:3.5m × 2.5m × 2.5m; 視野: 2.0° × 2.0°; 幾何学的面積:994cm ²						
	25 keV	30 keV	40 keV	50 keV	60 keV	80 keV
有効面積	93 cm ²	167 cm ²	228 cm ²	198 cm ²	172 cm ²	158 cm ²
6 時間観測で期待される最小偏光度 for 100 mCrab			10.5 %			
for 200 mCrab			6.5 %			
100 % 偏光ビームを用いた偏光度	33 %	29 %	26 %	27 %	32 %	40 %

表 2.2: PoGOLite の主な仕様 [18][19]

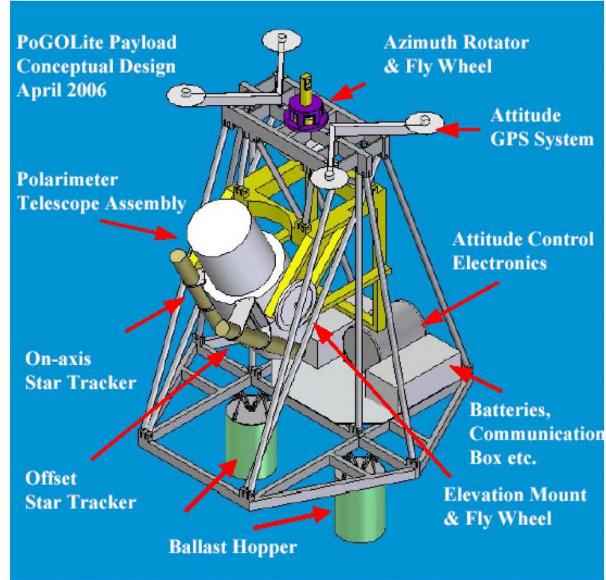


図 2.17: PoGOLite 検出器が搭載される気球ゴンドラ [3]

X 線帯域における偏光観測は OSO-8 で採用されたブレーグ反射を用いるもの、ガス検出器による光電吸収を用いたトラッキングの方向を測定するもの、コンプトン散乱の異方性を検出するものがある。主に 10 keV 以下の帯域では光電吸収により、25 keV 以上 1 MeV 以下ではコンプトン散乱により偏光が観測可能である。PoGOLite ではコンプトン散乱の異方性を用いて偏光を検出する。コンプトン散乱は図 2.18 のように X 線の偏光ベクトルに対し垂直方向に起こりやすい [3]。そこでコンプトン散乱の方向を同定するために六角形の井戸型シンチレータをハニカム構造に並べる。六角形にすることでき間無く配置することができ、四角形より正確にコンプトン散乱の散乱角が測定できる。本観測ではこのシンチレータユニットを 217 本、パスファインダーフライトでは 61 本搭載する。この主検出部を PDC 部 (2.6.1 節) と呼ぶ。目的となる 100 mCrab で偏光度 10.5 % を検出するためには X 線やガンマ線などのバックグラウンドを徹底的に除去する必要がある。このために BGO 結晶を用いた PDC 部のアクティブシールドを周囲に設置する。アクティブシールドとはシールドそのものを検出器として、観測した天体以外からのバックグラウンド外部からの信号の場合はシールド部と主検出部の二つが同時に反応する。このとき反同時係数法によりこのバックグラウンド信号のデータを破棄することができ、天体からの信号のみを選別することが可能となる。またシールドそのものが検出器なためバックグラウンドの見積りが可能であり、PoGOLite の他にも後述する ASTRO-H での採用も決定している。シールドは一枚で覆っているのではなく、本観測では 54 本、パスファインダーフライトでは 30 本のユニットに分割されており、それぞれのユニットでデータ処理を行うことで、デッドタイムを減らすとともにより正確にバックグラウンドを見積もることができる。このアクティブシールドを SAS 部 (2.6.2 節) と呼ぶ。図 2.19 に PDC 部と SAS 部を記載した PoGOLite 検出器の模式図を載せる。

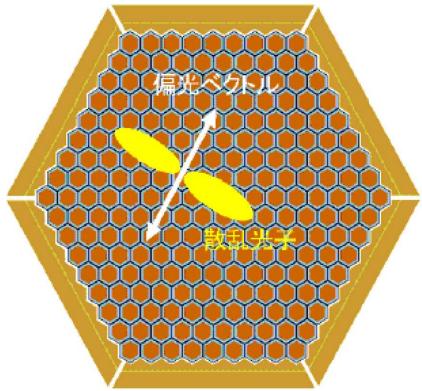


図 2.18: PoGOLite による偏光測定の概念 [3]

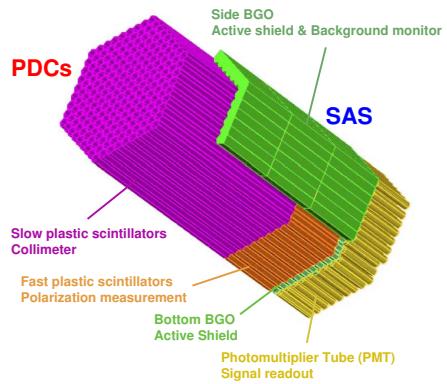


図 2.19: PoGOLite 検出器の模式図 [3]

2.6.1 Phoswitch Detector-Cells (PDC)

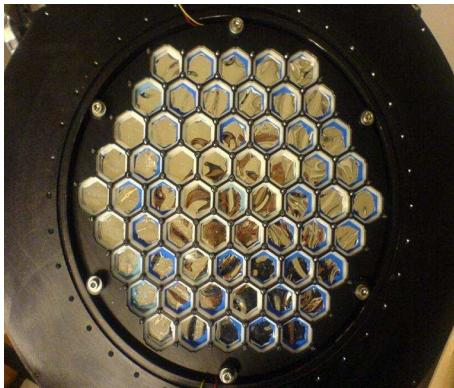


図 2.20: パスファインダーフライトにおける PDC ユニットの写真

PoGOLiteにおいて主検出部は Phosswitch Detector-Cells(PDC) と呼ばれている。1つのユニットが 1 m ほどのハニカム構造型をしており、中空構造の Slow プラスチックシンチレータ、Fast プラスチックシンチレータ、BGO 結晶が鉛直方向に重なって配置されている。この構造をフォスイッチ構造と呼ぶ。PDC 部においてこの 600 mm の長さを持つ Slow シンチレータはアクティブコリメータの役割を行い、Fast シンチレータでは光子を検出、BGO 結晶は底部でのアクティブシールドの役割をする。このアクティブコリメータによって視野を $2.0^\circ \times 2.0^\circ$ まで制限し、またアクティブシールドでバックグラウンドを除去することで高感度観測を実現する。図 2.21 に PDC ユニットにおける各シンチレータの配置とそれらの写真を記載する。一つ一つのユニットがフォスイッチ構造をしている。このシンチレータ信号を浜松フォトニクス社製の光電子増倍管(PMT)R7899EGKNP で読み取る。2つのプラスチックシンチレータは反射材(3M 製 VM2000)で、BGO は反射材として BaSO₄ がコーティングされている。X 線に対する阻止能を上げるために PDC 部の各ユニットは 50 um の鉛、スズの薄い膜で覆われている。鉛は阻止能が高く高エネルギーのガンマ線も防ぐが、74 keV の特性 X 線を発生させる。この X 線を防ぐためにスズが内側にまかれているのである。ス

ズも 20 keV の特性 X 線を発生するが PoGOLite の観測エネルギー域 (25 - 80 keV) ではほとんど影響を与えない。全てのユニットは PMT にて信号を検出する。PMT は半導体検出器に比べ、低ノイズで受光面も大きくすることが容易などの利点がある。PMT で読み出された信号は FADC Board に入力され、37.5 MHz サンプリングで AD 変換されて後段の FPGA でデータ処理が行われる。

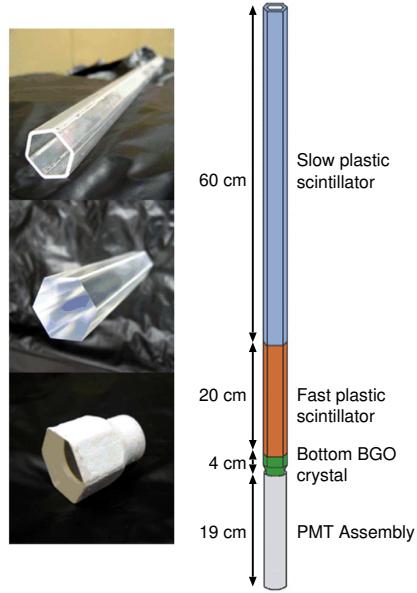


図 2.21: PDC の 1 ユニット図 [4]

目的とするガンマ線がユニットの視線方向から到来してきた場合、Slow シンチレータの空洞部分を通り抜け、Fast シンチレータが光り、視野角 $2.0^\circ \times 2.0^\circ$ 外からのイベントの場合は Slow シンチレータや底の BGO が光る。真横からのイベントの場合は周囲のアクティブシールドが反応する。よって視線方向から観測したい天体からの信号を検出するためには Fast シンチレータのみが反応したデータを選択すれば良い。図 2.22 に様々な入射方向パターンにおける散乱の様子を示す。このうち緑字で表された、Fast シンチレータのみが反応した信号が取得するデータである。

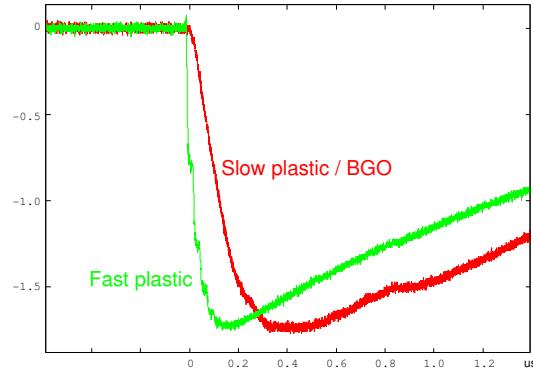


図 2.23: PDC の各シンチレータでの波形 [4]

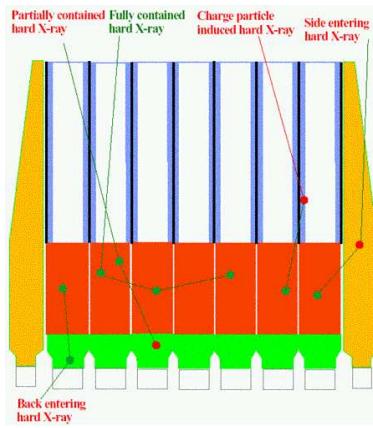


図 2.22: PoGOLite 検出器の断面図 [4]

PDC 部では一つの PMT で 3 つのシンチレータを読み出すため、Fast シンチレータでの信号か、それ以外のシンチレータによる信号化を区別するために波形弁別を用いる。波形弁別は時定数の違いを利用して特定のシンチレータの信号を取り出す方法である。各シンチレータの減衰時定数 τ は Slow シンチレータが ~ 300 ns、Fast シンチレータが $\tau \sim 2$ ns、BGO 結晶が $\tau \sim 300$ ns である。各シンチレータの波形を図 2.23 に記載する。波形を弁別するための Discri 信号を PulseShapeDiscri(PSD) 信号と呼び、Slow/BGO シンチレータからの信号であると判断されたときはこの PDS 信号が出力される。これにより純粋な Fast シンチレータの信号のみを取り出すことができる。

コンプトン散乱の異方性を調べるために散乱光子の方向、およびエネルギー値を調べる必要がある。そのため PoGOLite ではあるユニットで光子が検出されたら素早く全ユニットで同時にヒットがあったかどうかスキャンを行い、データ処理をする。観測時のトリガは 0.5 kHz/s と予測されているが、この全スキャンから保存までのデッドタイムは $\sim \text{us}(0.6\%)$ と十分に短い。こうして Fast シンチレータに入射した信号を地上で詳細に解析し、コンプトン散乱の異方性を調べることで入射した硬 X 線・軟ガンマ線の偏光度を観測できる。

2.6.2 Side Anticoincidence Shield (SAS)

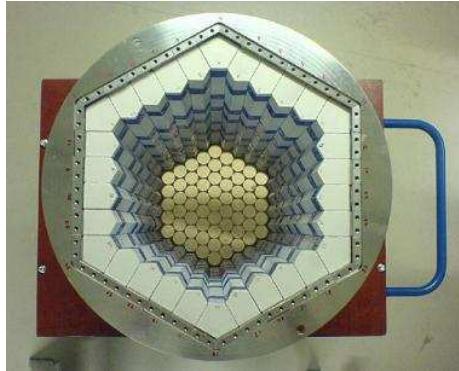


図 2.24: パスファインダーフライトにおける SAS ユニットの写真

PoGOLiteにおいてアクティブシールド部は Side Anticoincidence Shield(SAS) と呼ばれている。PDC ユニットの周囲を本観測では 54 本、パスファインダーフライトでは 30 本の BGO 結晶で覆っている。高さは PDC ユニットの 2/3 ほどであり、断面は五角形型である。反射材として各ユニットに BaSO₄ がコーティングされており、ユニット間の隙間は～100 μm と小さい。また中性子バックグラウンドを防ぐため、SAS ユニット周囲は厚さ 15 cm のポリエチレンで覆われている。バックグラウンドは全 SAS ユニットで 100 kHz と見積もられており、PDC 部で検出されたイベントのうち 6 % は偶然同時にバックグラウンド信号とみなされてしまうが、SAS ユニットの位置を区分けすることで有効なイベントが除去される率を減らすことができる可能性がある。SAS 部の信号も PDC 部と同じく PMT(R7899EGKNP) で読み取られ、SAS ユニットによる VETO 信号出力スレッショルドは 75 keV ほどになる。

2.7 SpaceWire を用いた観測器器 2:ASTRO-H

我々が開発に参加しているもう一つの計画に次期 X 線天文衛星 ASTRO-H(旧称 NeXT) がある。ASTRO-H は 2013 年度打ち上げであり(図 2.26)、10 keV から数 100 keV の帯域においてこれまでのいかなる検出器よりも一桁優れた感度を得ることが要求されている。図 2.25 に ASTRO-H の完成予想イラストを、表 2.3 に ASTRO-H の基本情報を載せる。

打ち上げ年	2013 年度
打ち上げ場所	種子島宇宙センター
打ち上げロケット	JAXA H-II-A ロケット
重量	2.4 t
全長	14 m
軌道高度	550 km
軌道種類	円軌道
軌道周期	96 分

表 2.3: ASTRO-H の機体データ

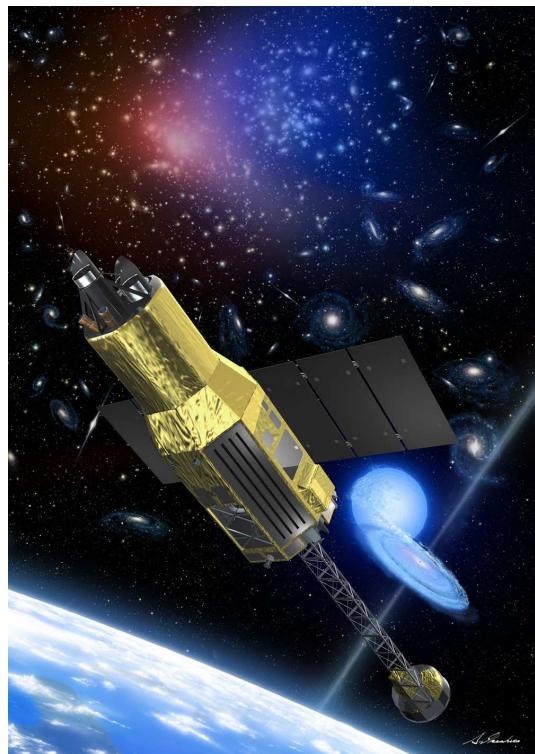


図 2.25: 次期 X 線天文衛星 ASTRO-H 完成予想イラスト [25]



図 2.26: ASTRO-H 開発スケジュール [25]

ASTRO-H 計画には宇宙航空研究開発機構・宇宙科学研究所本部 (JAXA/ISAS) を中心に、国内外の多数の大学・研究機関が開発に参加している。ASTRO-H では (1) 硬 X 線望遠鏡による初めての撮像分光観測、(2) 初めてのマイクロカロリメータによる超高分解能分光観測、(3) 0.3 keV から 600 keV と 3 枠以上にもおよぶ過去最高の高感度広帯域観測を開発の目的とする。これによりブラックホールの周囲や超新星残骸など高エネルギー天体や銀河団を観測することで宇宙の進化や大規模構造の解明、宇宙の極限状態の理解、シンクトロン放射など多種多様な非熱的現象の解明、ダークマターやダークエネルギーの解明が期待される。これらを考慮して、ASTRO-H 衛星は次のような機器を観測装置として設置する。

- 硬 X 線撮像分光検出器 (HXI)

硬 X 線望遠鏡 (Hard X-ray Telescope:HXT) と組合せ、10 - 80 keVにおいてこれまでより 1 枠以上に優れた感度を達成する。

- 軟ガンマ線検出器 (SGD)

10 - 600 keV で過去最高感度を得る。特に 100 keV 帯域において現存の検出器を 1 枠以上上回る感度を得る。

- 軟 X 線精密分光検出器 (SXS)

マイクロカロリメータを用いて 0.3 - 12 keV における 5 eV のエネルギー分解能、エネルギー決定精度 1 eV の分光を行う。

- 軟 X 線撮像分光検出器 (SXI)

0.4 - 12 keV において軟 X 線の撮像分光観測を行う。

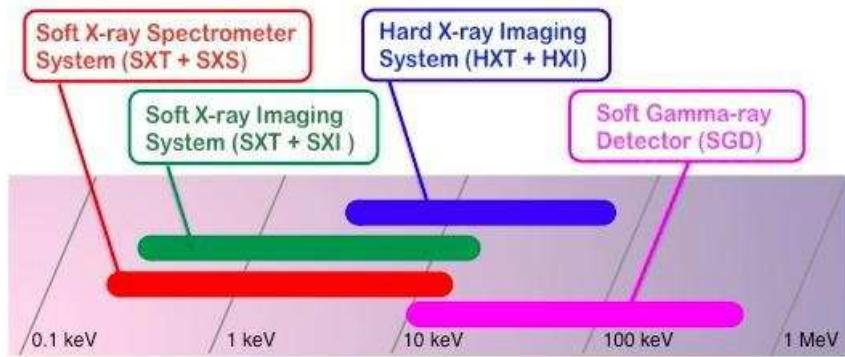


図 2.27: ASTRO-H の各観測装置に要求された観測エネルギー帯域 [25]

各検出器の観測エネルギー帯域を図 2.27 に示す。これら検出器を用いて従来より 1 枠程度良い感度の観測を実現する。特に 10 - 80 keV においては従来の衛星と比較して数十倍優れていると期待されている。ASTRO-H では衛星内の観測装置、データ処理装置間でのデータ転送やコマンド送受信のプロトコルとして SpaceWire が本格採用され、各種搭載機器には SpaceWire インターフェイスが搭載されている。

2.7.1 HXI/SGD

HXI

ASTRO-H に搭載される硬 X 線撮像検出器 HXI は硬 X 線スーパーミラー (HXT) が集光・結像した光子を 5 - 80 keV の帯域にて撮像する。5 - 80 keV までの X 線に対して高い(50 %以上)検出効率を持ち、徹底したバックグラウンドの除去が求められている。表 2.5 に HXT の特長を示す。

搭載台数	2
焦点面検出器	Hard X-ray Imager (HXI)
有効エネルギー領域	80 keV 以下
有効面積	~ 300cm ² at 30 keV
結像性能	~ 60 arcsec
有効視野	~ 8arcmin

表 2.4: 硬 X 線望遠鏡 (HXT) の特長

図 2.28 に HXI の構造図を載せる。 HXI は主検出部に 10 - 80 keV までを担当する CdTe ピクセル検出器が下段、 5 - 30 keV までの両面シリコンストリップ検出器 (Double-sided Silicon Strip Detector : DSSD) が上段に 4 層配置される。これによって軟 X 線は DSSD で検出されるのに対し、硬 X 線は DSSD を突き抜けて HXI で検出されるため、極めて広いエネルギー範囲における高感度撮像が可能となる。 CdTe は $Z_{\text{Te}} = 52$ 、 $Z_{\text{Cd}} = 48$ と比較的大きな原始番号を持つため高い阻止能を実現する。 CdTe では 0.5 mm 厚で 60 keV の硬 X 線を 80 % 以上止めることが可能である (GaAs で 40 %、シリコンで数%)。また半導体検出器であるゆえの優れたエネルギー分解能、位置分解能を持つ。これだけ優れたものであるが、大面積化が難しく、近年の技術革新によって実用に用いるまでの大きさになるまでなかなか現実のものとならなかった。 DSSD は n 型半導体基板上に短冊上の n+ 領域と p+ 領域が直行するように配置された半導体検出器である。細長い多数のダイオードが並んだ構造をしており、バルクで電子生成対反応が起こった際は正孔と電子を n+ と p+ にて収集するため、CCD や Si-Pad 検出器のようにパッド分のチャンネルを読まず、ストリップ分のチャンネルを読み出すだけで良いという特長がある。開発は浜松ホトニクス社で行われ、2.6 cm 角の DSSD に対して 1.3 keV、4 cm 角の DSSD で 1.6 keV(FWHM) を達成した。表 2.5 に HXI への要求性能を記載する。

	要求値
エネルギー帯域	5 - 80 keV
エネルギー分解能	< 2keV (FWHM, at 60 keV)
有効面積	300cm ² at 30 keV
視野	9 × 9arcmin ²

表 2.5: HXI への要求性能

この帯域では天体信号が小さいことに加え、コンプトン散乱と光電吸収が混交しているため他の波長帯に比べて S/N 比が小さく、観測には徹底したバックグラウンドの除去が不可欠である。そのため、PoGOLite 同様、「すざく」衛星硬 X 線検出器 (HXD) で実績のある BGO 結晶による井戸型アクティブシールドで覆われている。BGO アクティブシールドのシンチレーション光の読み出しには従来用いられた PMT ではなく、大面積化の進むアバランシェ増幅型フォトダイオード (Avalanche Photo Diode : APD) を用いる。APD は PMT と比べると信号ノイズが大きいが、サイズが大幅に小さくなり、量子効率が高い。この BGO 結晶と APD 素子によるアクティブシールドは SGD にも採用されている。

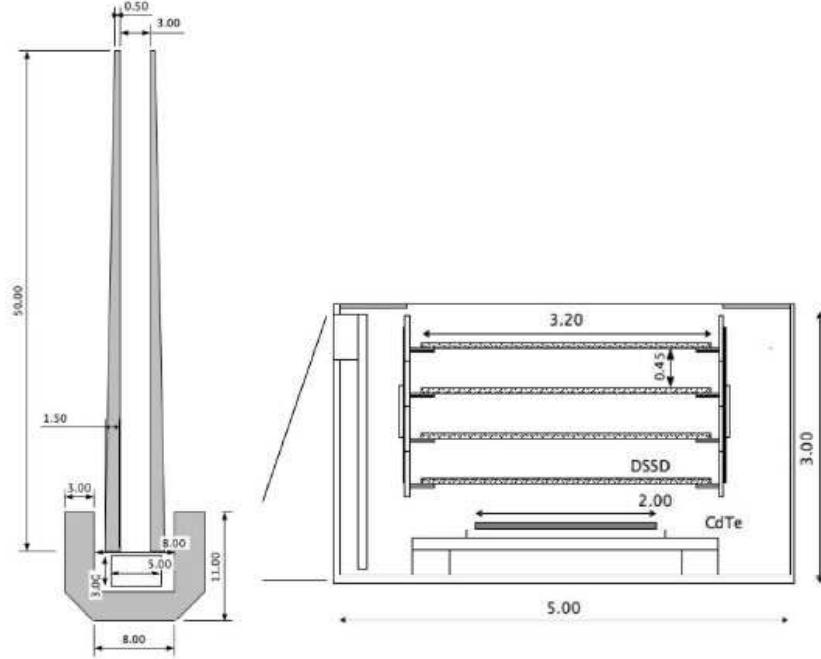


図 2.28: HXI の構造 [3]

SGD

スーパーミラーによる集光・結像技術の現在の上限は～80keV ほどであるが、加速された粒子の最高エネルギーを示すスペクトルの折れ曲がりを観測することも解析の上では重要な指標となる。そこで軟ガンマ線検出器 SGD は BGO アクティブシールドに加えて、Si/CdTe コンプトンカメラを用いることで徹底したバックグラウンドの除去を行った上で 10 - 600 keV の広帯域観測を目標としている。このバックグラウンドの除去によって、特に 100 keV での帯域における感度を従来の性能から 1 衝上回ることを目標としている。図 2.29 に SGD の構造を、表 2.6 に要求性能を載せる。

	要求値
エネルギー帯域	10 - 600 keV
エネルギー分解能	< 2keV (FWHM, at 40 keV)
有効面積	150cm ² at 30 keV
視野	33 × 33arcmin ² (< 150keV) 600 × 600arcmin ² (> 150)

表 2.6: SGD への要求性能

以下簡単にコンプトンカメラの原理を示す。SGD で用いられるコンプトンカメラは大きさが 5 cm 角ほどで、ピクセル状の Si 検出器 (Si-Pad) と CdTe 検出器が複数重ねられた構造をしており、周囲も CdTe 検出器で覆われている。各物質における反応断面積を図 2.30 に載せる。ガソマ線が検出器に入射するとコンプトン散乱によりエネルギーの一部を落としながら次の検出器へと進む。

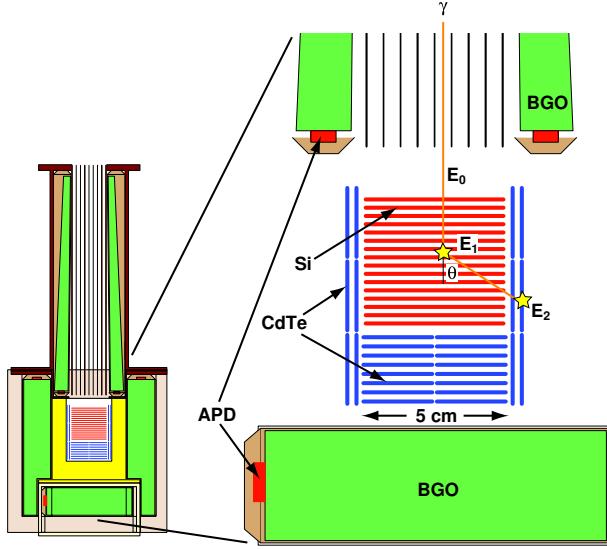


図 2.29: SGD の構造 [7]

このときコンプトン散乱で電子に与えたエネルギーを E_1 とする。散乱されたガンマ線はまたコンプトン散乱を繰り返し、最後には検出器にて光電吸収される。この散乱されたガンマ線のエネルギーを E_2 とする。このときエネルギー保存則と運動量保存則から散乱角 θ は以下の式 2.1 で与えられる。

$$\cos \theta = 1 - m_e c^2 \left(\frac{1}{E_2} - \frac{1}{E_1 + E_2} \right) \quad (2.1)$$

これより、正確なエネルギー値と光子の散乱・吸収箇所が分かれば光子の到来角度が θ で制限される。これは天空では円錐形で与えられ、複数の光子によって円錐を重ねればその交点からガンマ線源の位置を求めることが可能となる。

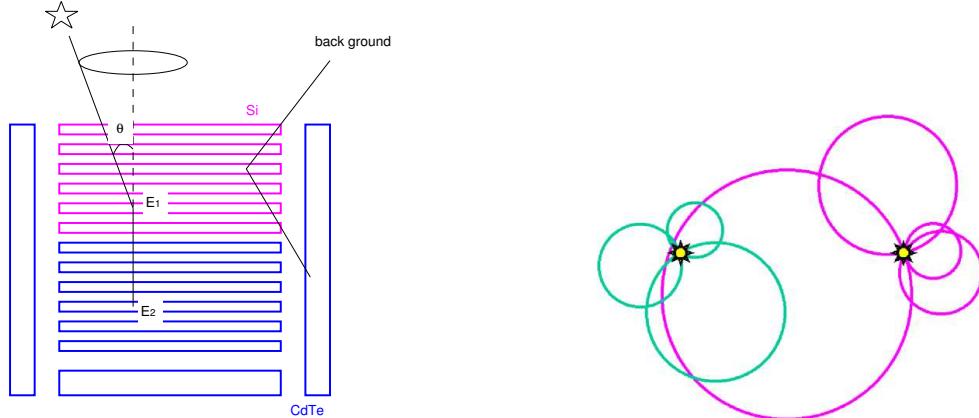


図 2.31: コンプトンカメラの原理 [3]

入射されたガンマ線が視野外からの信号であると判断されればバックグラウンドとみなすこと

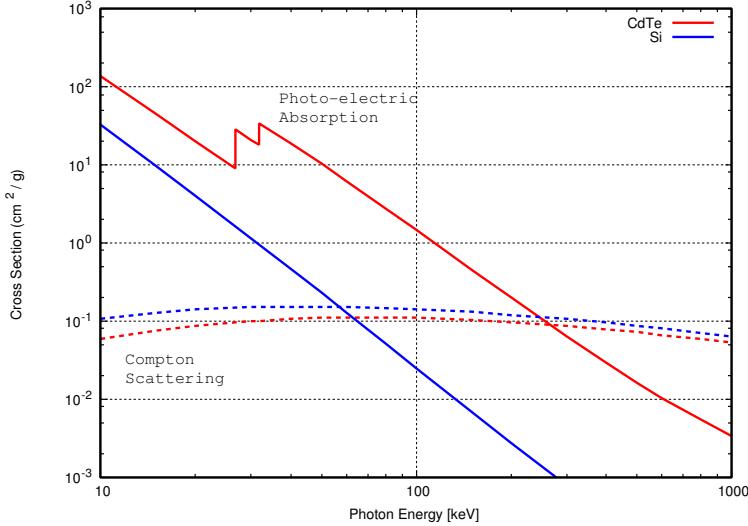


図 2.30: SGD 観測帯域における Si と CdTe の光電吸収、コンプトン散乱反応断面積 [33]

き、これによって視野を制限して S/N 比を向上することができる。このためには Si-Pad 検出器、CdTe 検出器において優れたエネルギー分解能、位置分解能を達成する必要がある。現在、5 cm 角のコンプトンカメラでは各検出器に対してエネルギー分解能が 2 keV 以下、また 1 - 2 mm の位置分解能を持つことが要求されている。

2.7.2 ASTRO-H における BGO アクティブシールド

宇宙線は地上においては大気で散乱・吸収されるため日常においてその影響は意識しないが、宇宙空間の衛星ではその宇宙線が降り注ぐ。そのフラックスは $\sim 1/\text{s}/\text{cm}^2$ であり、エネルギーは 100 GeV 以上にも及ぶ。宇宙線が入射されて検出器などを突き抜けると二次放射が発生し、周囲の検出器にも影響が及ぶ。結果として重大なバックグラウンドとなる。そのため ASTRO-H では「すぐ」衛星で多大な実績のある BGO アクティブシールドを採用する。図 2.32 にすぐ衛星硬 X 線検出器 (HXD) でのアクティブシールドバックグラウンドの効果を表した図を載せる。すぐ衛星ではアクティブシールドを用いることで、30 - 40 keV の帯域では $\sim 3 \times 10^{-4} \text{ cnts}/\text{s}/\text{keV}/\text{cm}^2$ から $\sim 3 \times 10^{-5} \text{ cnts}/\text{s}/\text{keV}/\text{cm}^2$ までと、1 衝程度バックグラウンドが異なることわかる。ASTRO-H におけるアクティブシールドも PoGOLite 同様、主検出部を視線方向以外シンチレータでシールドしたもので、宇宙線の入射を最低限にまで抑えると同時に、シールドを突き抜けるようなバックグラウンドも主検出部と反同時計数を取ることでバックグラウンドを除去することができる。シールド自体が検出器であるため、ガンマ線バーストなどの突発天体のモニタを行うことも可能である。ASTRO-H における BGO 結晶のユニット数は HXI が 9 本、SGD が 25 本である。

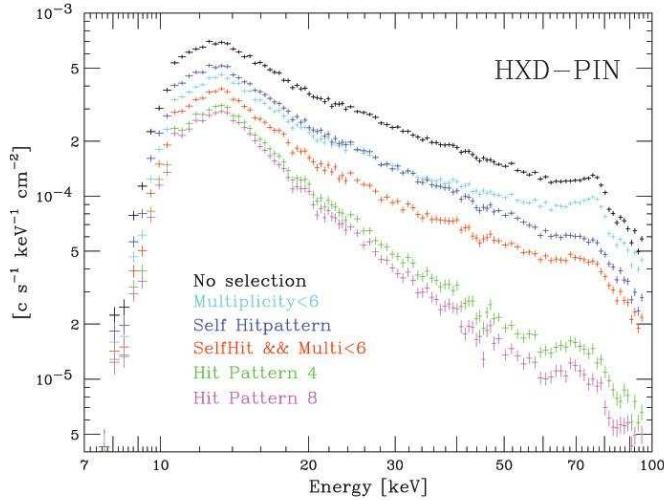


図 2.32: すぐ衛星 HXD で実証された井戸型 BGO アクティブシールドによるバックグラウンドの低減手法。[21][22]

BGO ユニットが多数あるので衛星という限られた電力、スペースでは従来のシンチレータ読み出しに用いていた光電子増倍管を用いることはできない。そのため ASTRO-H では BGO 結晶の読み出しにアバランシェフォトダイオード (APD) を用いる。HXI/SGD は APD 信号から後段回路によって反同時係数信号を出力するが、その信号も FastVETO 信号と HitPattern 信号に分かれ。FastVETO 信号は BGO で明らかにイベントである信号を検出した場合、5 usec 以内に主検出部へと出力され、それを受け取った主検出部はこの時のイベントデータ処理を中止する。これによってバックグラウンドを処理するデッドタイムを無くすことができる。一方、HitPattern 信号は FastVETO で拾いきれなかったノイズに埋もれるほどの低エネルギーイベントを検出し、それを約 30 usec 後に主検出部へと出力する。このとき主検出部はデータ処理を行ってしまうが、アクティブシールドから受け取った HitPattern 信号を照合することにより、地上でのデータ処理において FastVETO では落とせなかったバックグラウンドを除去することができるのでより高感度のデータとなる。

BGO 結晶の読み出しに用いる APD は小型で量子効率が高く增幅率も大きいが半導体検出器であるがゆえに光電子増倍管に比べてノイズ信号が大きい。さらに BGO は阻止能は高いが発光量が少ないという問題点もある。FastVETO や HitPattern のデータ処理も含め、いかにノイズに対してバックグラウンド信号の感度をあげるか、ということが課題になる。このような状況で最も効果があるのがノイズフィルタだが、HXI で 9 本、SGD で 25 本の APD 信号を全て従来のアナログフィルタ方式で実現するでカバーすることは限られた回路面積では難しい。そこで HXI/SGD では FPGA 上でディジタルフィルタを搭載することとしている。

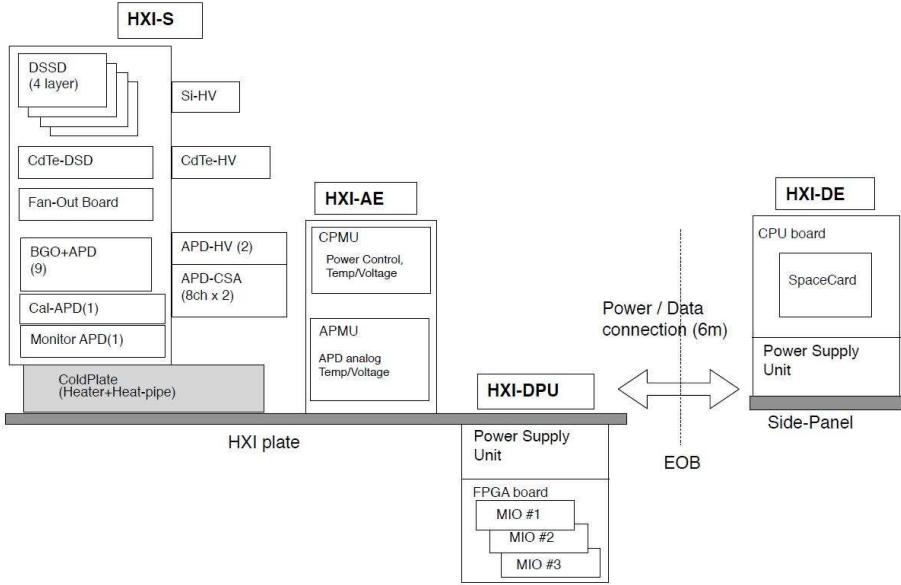


図 2.33: HXI における信号処理部コンポーネント [23]

2.7.3 BGO 信号処理系回路

図 2.33 のように HXI/SGD では信号処理回路系が 4 つのブロックに分かれている。 HXI/SGD-S (Sensor) 部、 HXI/SGD-AE (Analog Electronics) 部、 HXI/SGD-DPU (Data Processing Unit) 部、 HXI/SGD-DE(Digital Electronics) 部である。 BGO+APD 信号センサー部から前置増幅器 (CSA) で増幅された信号は AE 部の APMU(APD Processing and Management Unit) へと入力される。 APMU においてアナログ信号の波形増幅、整形が行われ、 AD 変換された後に FPGA においてデジタル処理される。処理はデジタルフィルタ、主検出部への VETO 信号出力、ヒストグラム生成、ガンマ線バースト判定などが行われる。 APMU では他にアナログ回路のゲイン調整、電流値、電圧値、温度などのデータ (House Keeping:HK) 取得も行う。 DPU 部は SpaceWire を通じた DE へのインターフェイスの役割、 APMU・CPMU からのデータの受け取り、 APMU・CPMU の制御などを行う。 DE 部はマイクロプロセッサを含み、イベント信号や HK データの処理を行う。また衛星から送られるコマンドを解析し、 DPU 部へコマンドを送信する。

APMU では APD 素子で検出された信号は主検出部への VETO 信号生成、ヒストグラム取得とともに、ガンマ線バースト現象 (GRB) などの突発現象のモニタも含まれる。 GRB はひじょうに大きなエネルギーを持つため、 APMU では 5 MeV までのエネルギーが観測できるように ADC のダイナミックレンジが 0 - 5.2 MeV となるように設計されている。 BGO 結晶を読み出すためのサンプリング周波数を 1 MHz とし、そのための ADC 素子が ADC128S102QML である。供給されたクロックの 1/16 サンプリングとなるため、 APMU-FPGA では ADC128S102QML に 15 MHz のクロックを与えている。図 2.34 にセンサー部から ADC 変換までの CSA、ゲイン調整、積分フィルタのブロック図を記載する。

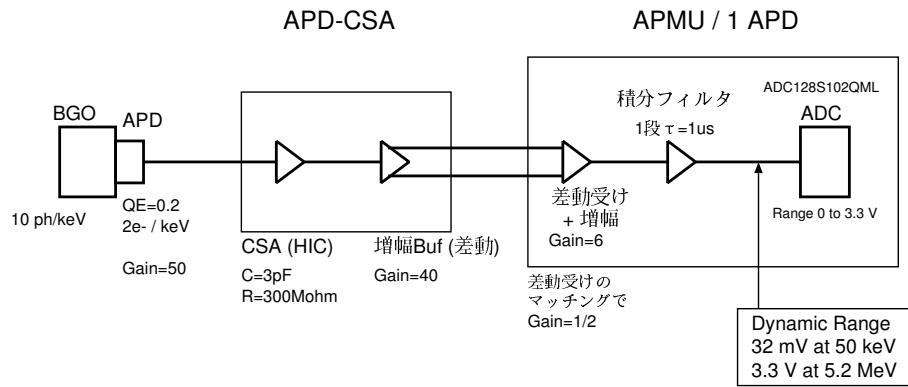


図 2.34: HXI/SGD の APD 信号処理

本研究では従来は実機の APMU での環境を想定し、信号整形をデジタルで行った上で、VETO 信号の出力効率の調査、図 2.34 に近い環境を実験室で構築したうえでのヒストグラムの効果を調べた。

第3章 PoGOLite用データ収集システム開発

3.1 研究目的

PoGOLiteは硬X線・軟ガンマ線の偏光を観測するためにPDC部(2.6.1節)と呼ばれるフォスイッチ型ユニットをハニカム構造で配置する。また上空におけるバックグラウンドを除去するためにSAS部(2.6.2節)と呼ばれるBGOアクティブシールドを周囲に搭載している。フォスイッチ構造であるPDC部はシンチレータの減衰時定数が異なることを利用して波形弁別を行い、取得すべき信号かどうかを取捨選択し、バックグラウンドは除去する。また地上にてデータ解析を行う際により詳細な偏光解析を行うために波形データを取得する。SAS部はアクティブシールドとして、VETO信号を出力する機能が必要である。更にPDC、SASともにバックグラウンドの見積りやユニットの状態管理のためパルスハイトヒストグラムを取得する機能を搭載する。硬X線偏光やVETO信号出力を行うために多数のシンチレータを扱い、その読み出しには低ノイズ高ゲインの光電子増倍管(PMT)を用いる。その本数はパスファインダーモデルで92本にもなり、それらを一括で処理する必要がある。

これまでに先人達の努力により、PDC部で最低限必要である波形弁別と波形取得機能、SAS部で必要であるVETO信号出力、パルスハイトヒストグラム生成機能は作成されていた。しかしこのままではPDC部用とSAS部用と二つの回路を作成しなくてはならないし、目標とするPDC部でのヒストグラム生成が達成できない。そこで私はPDC部とSAS部の必要な機能をまとめ、PDC/SAS部で共用の回路を作成した。これによって回路上ではPDC部とSAS部の区別が無くなりデバッグが容易になる。また上述のようにPDC部でのヒストグラム生成が可能になった。

PDC	波形弁別、波形取得、パルスハイトヒストグラム生成
SAS	VETO信号出力、パルスハイトヒストグラム生成
中性子シンチレータ	波形弁別、波形取得、パルスハイトヒストグラム生成

表 3.1: PoGOLite の各部において実装すべき機能

またPoGOLiteでは世界で初めて、硬X線・軟ガンマ線観測と同時に中性子シンチレータによる大気中性子バックグラウンドのリアルタイム計測を行う。中性子シンチレータとしては1熱中性子捕獲につき9000光子を放出するLiCAF(Eu 2%)をPDC部同様にBGO結晶で挟んだフォスイッチ構造で用いる。これによってPoGOLiteの主要なバックグラウンド源である中性子の変動を追え、より良い感度を達成できる可能性がある。PDC部と同じく波形弁別によって中性子由来かガンマ線などが由来かを判別できるが、LiCAF(Eu)による波形の減衰時定数($\tau \sim 1600$ ns)はBGO($\tau \sim 300$ ns)と比べてはるかに長く、PDC部のデータ処理システムをそのまま使うことは

できない。また他の PDC 部や SAS 部が DIO Board をブレインとしたツリー状にシステムが成立しているのと違い、この中性子シンチレータのデータ処理は独立に動く必要がある。そこで私は中性子シンチレータのために、LiCAF シンチレータの蛍光波形に合わせたプロセスでありながら、中性子シンチレータ部のみを独立に動作させるシステムを開発した。図 3.1 に PoGOLite バスファインダーフライトの検出器の概要を載せる。

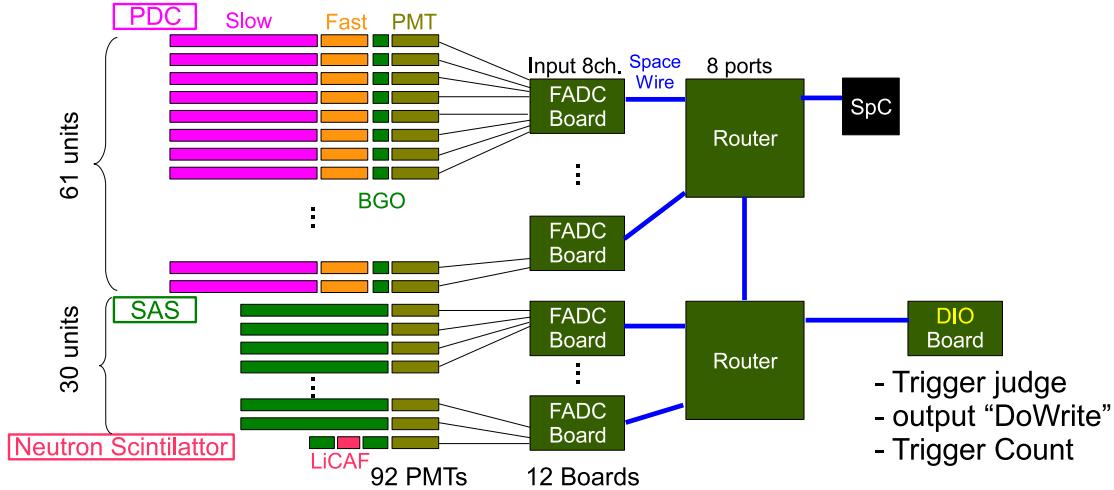


図 3.1: PoGOLite 全体接続図

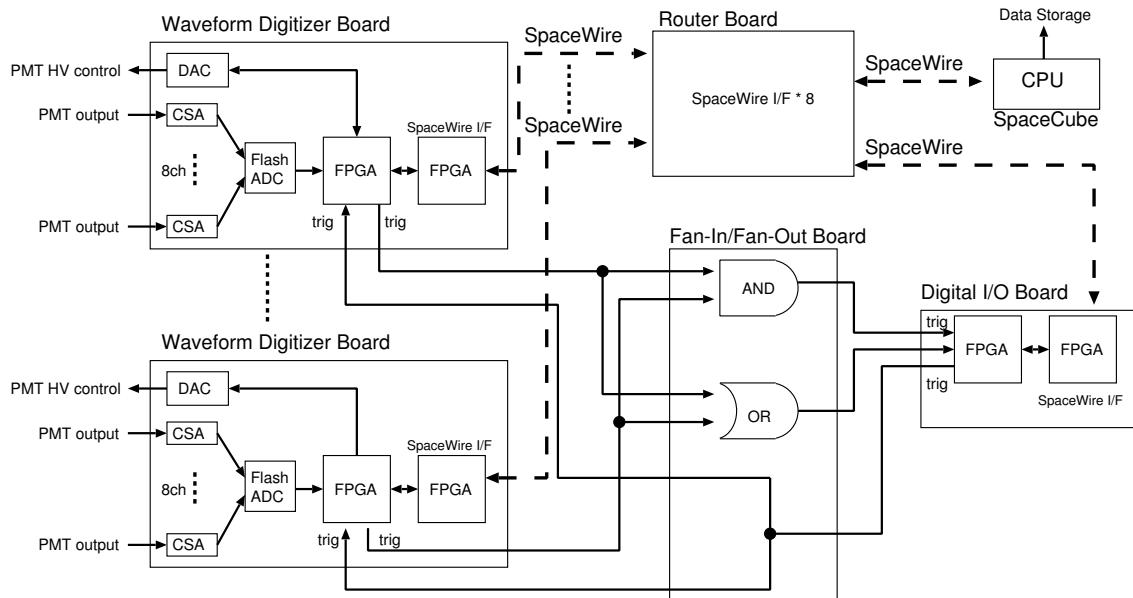


図 3.2: PoGOLite PDC 部におけるトリガ信号の動向 [17]

3.2 PoGOLite で実装すべき機能とその開発

3.2.1 波形保存プロセス

PoGOLite で用いる FADC Board は先行研究によって PMT の波形信号を以下の図 3.3 のように取得することが可能である。

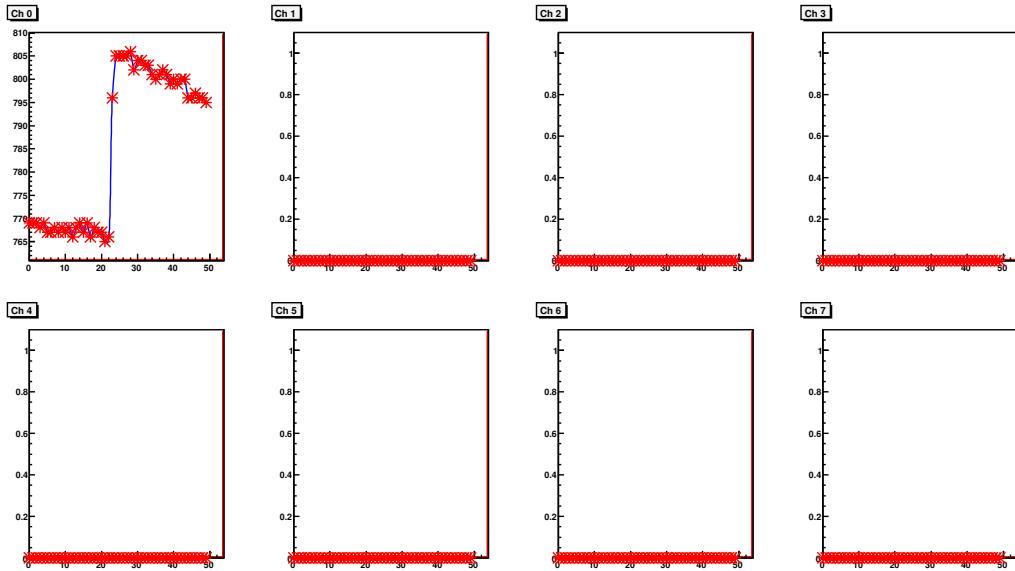


図 3.3: FADC Board で取得した波形の例

この波形は 37.5 MHz サンプリング周波数で取得している。FADC Board には 8 ch. のアナログ入力部があり、この図 3.3 では ch 0 にしか信号を入力していないため他のチャンネルでは波形データが存在していない。波形保存は図のように 37.5 MHz で 12 bit 分解能の信号を 50 クロック分保存している。また入力信号には 750 ほどのオフセットがはかせてある。PoGOLite では波形弁別、波形取得、ヒストグラム取得のプロセスでは、いずれもこの 12 bit 分解能、37.5 MHz サンプリング周波数の波形を用いている。

以下では図 3.4 のブロック図をもとにこの波形保存プロセスについて説明する。

PoGOLite では波形を保存する上で波高値があるトリガレベルを超えるとトリガ(Trigger Hit-Pattern)信号を出力するとともに Fast シンチレータか Slow シンチレータ/BGO 結晶シンチレータの波形を弁別する。各シンチレータは減衰時定数 τ が Fast シンチレータが $\tau \sim 2$ ns、Slow シンチレータ $\tau \sim 230$ ns、BGO 結晶 $\tau \sim 300$ ns と異なるため、信号は図 2.23 のように波形の立上り時間が異なる。Fast シンチレータの波形ではない遅い信号と判別された場合にはトリガとは別の Pulse Shape Discri 信号を出力し、波形の弁別を実現する。波形による弁別の様子を表した図が図 3.5 である。波形の時定数が短い場合は図中の PeakFast と PeakSlow がほぼ等しくなり、長い場合は PeakSlow > PeakFast となる。

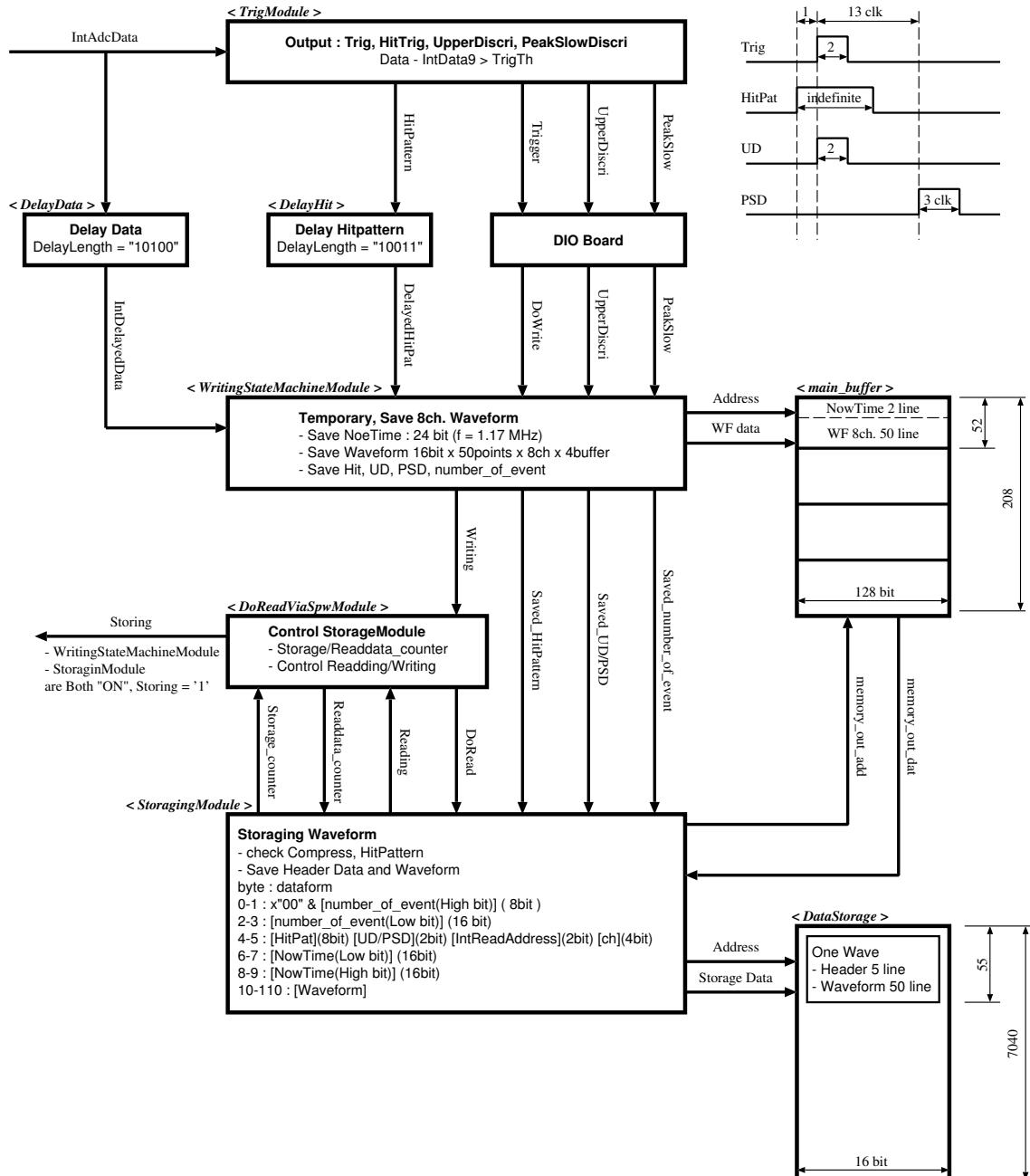


図 3.4: PoGOLite 波形保存ブロック図

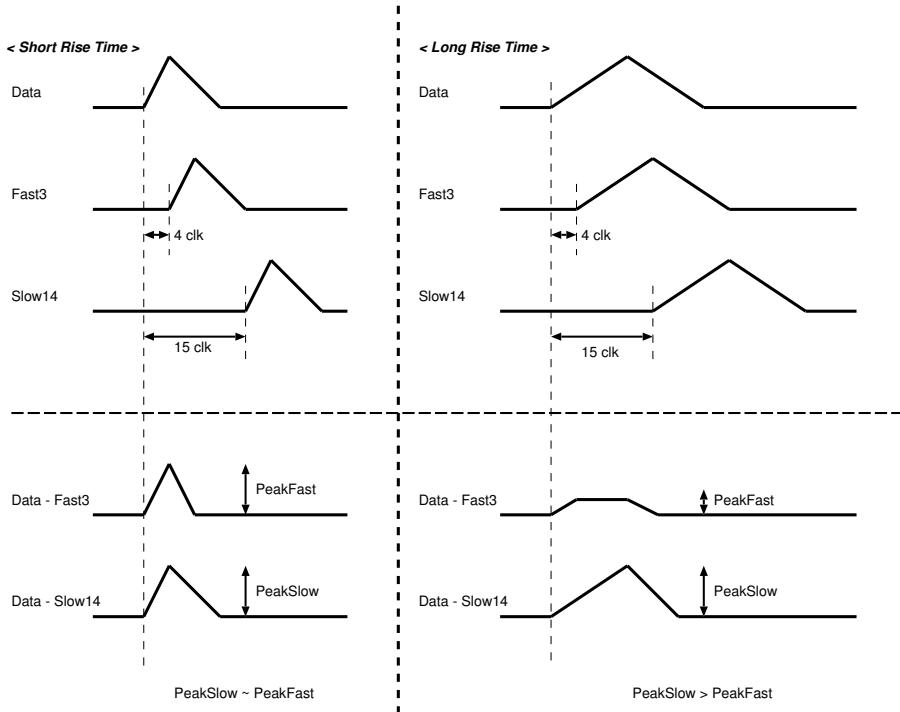


図 3.5: 波形による FastPeak と SlowPeak の違い

また Fast シンチレータ信号は 15 keV - 200 keV の帯域をカバーするが宇宙線によるバックグラウンドなどの高い信号が入力された場合はその信号を除去するために別途 Upper Discri 信号を出力する。

以上が波形弁別機能についてである。これらをまとめると一つのPDC ユニット毎に以下の表3.2のようなトリガ信号と波形弁別信号を出力する。

Trigger	波高値がスレッショルドを上回ったら出力
Pulse Shape Discri (PSD)	波形から弁別。これにより Fast シンチレータか Slow/BGO シンチレータかを区別
Upper Discri (UD)	波高値が上限値を上回ったかどうか。
Hit Pattern	どの PMT が反応しているのか。

表 3.2: PDC 部での弁別信号

PoGOLite ではこのトリガや弁別信号が 12 枚の FADC Board から独立に出力され、それをを DIO Board で集める。DIO Board はトリガが入力されてからある時間幅の間でトリガを出力したボードの PSD、UD を集め、そのトリガで検出された信号を保存するかどうか判定する。PSD によりこの信号が Slow/BGO シンチレータ信号であると判断されたときや、UD を上回った高い信号であれば保存させない。UD を下回り、Fast シンチレータからの信号であると判断されれば DIO Board から全 FADC Board へと波形を保存するための'DoWrite' 信号が出力される。

各 FADC Board は DoWrite 信号を受け取ったら波形の保存を開始する。トリガを検出してしばらく時間が経過しているため、そのときの PMT 信号を保存してもピークを過ぎた波形となるので

予め遅延させていた波形を保存する。トリガが立つ 0.4 usec 前から保存を開始し、50 クロック分およそ 1.4 usec 分保存する。これによりトリガ前の波形の様子や波形の減衰の様子も保存が可能である。DoWrite を受け取った FADC Board が全チャンネルの PMT 信号を保存していると 1 回の波形を保存するプロセスで ~20 usec ほどかかるてしまい、大きなデッドタイムになってしまう。それを解消するために HitPattern 信号を用いて信号が検出されたチャンネルのみを選択して保存する。仮りに 1 チャンネルのみの保存であれば 4.3 usec ほどに時間を抑えることができる。

波形は FADC Board の FPGA メモリに一時的に保存される。FPGA のメモリでは波形データは 128 セットしか保存できないので大容量の Flash Memory がセットされたブレインである SpaceCube に波形を転送する。波形データは以下の図 3.6 に沿ったフォーマットで保存されているため、後でオフラインで解析する際どのボードのどの時刻でどのチャンネルのデータなのか、またそのときのボードのヒットパターン、UD/PSD の情報も知ることができる。

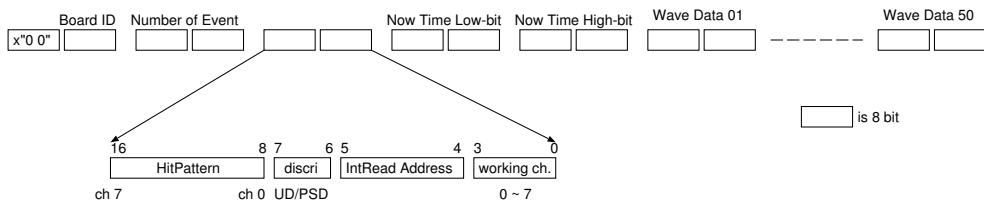


図 3.6: PoGOLite における保存した波形のデータフォーマット (1 波形:110 byte)

各ボードの時刻はボードのクロックに依存して時刻付けがされている。そのため時間がたてば時刻にズレが生じてくる。このズレは放球している間での対処はできないため、オフラインで解析を行う際にに対応できるように、一定間隔で DIO Board から pseudo 信号(偽信号)が出力される。この信号を受け取ったら各 FADC Board は波形を pseudo 信号に合わせたフォーマットで保存する。このとき時刻も図 3.6 同様に保存される。解析時にこの pseudo 信号を受け取った時刻を調整すれば全ての Board での時刻を合わせることが可能となる。

オフラインではこの波形を元に解析を行う。偏光度を調べるために Fast シンチレータで検出された信号のみで解析を行う必要があるが、DIO Board から DoWrite 信号を受け、HitPattern 信号によって保存された波形の中には Slow/BGO シンチレータによる波形も含まれる。そのためより純度の高いデータにするためにオフラインでも Fast シンチレータで検出された波形のみを選択する。上述したように Fast シンチレータと Slow/BGO シンチレータでは波形が異なる。そこで立上りの違いを考慮し、Fast と Slow の異なるゲート幅でピーク値を取得する (FastPH/SlowPH)。Fast シンチレータのように時定数が短ければどちらのゲート幅でも同じピーク値となり、Slow/BGO シンチレータのように時定数が長ければ Fast ゲートによるピーク値が低くなる。こうして得られる二次元ヒストグラムが図 3.7 である。

fastPH vs. slowPH of central unit

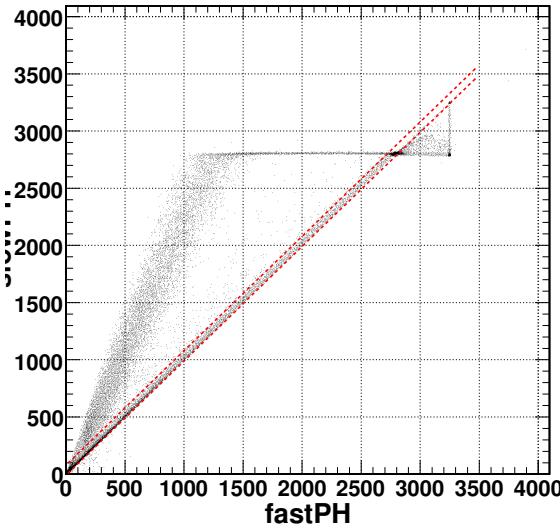


図 3.7: 保存した波形をイベント選択した 2 次元ヒストグラム [4]

ここから $\text{FastPH}=\text{SlowPH}$ となる直線上に存在するイベントのみを Fast シンチレータで得られた波形として取得する。図 3.6 のデータフォーマットからイベントの時刻、ユニット位置を知ることでどのユニットに天体信号が入射されてどの方向へコンプトン散乱が発生したのかを知ることができる。実際に組み上げた実機において 50 keV、90 % 偏光のペンシルビームを中心ユニットに照射してコンプトン散乱をさせ、その光電吸収とコンプトン散乱の波形を取得したものが図 3.8 である。ch.0 がコンプトン散乱で ch.4 が光電吸収である。このようにして検出された波形を解析することで天体信号のコンプトン散乱における散乱方位角の異方性を調べることで天体信号の偏光度を測定できる。最後に以上のプロセスにおけるクロックタイミングを図 3.9 に記載する。

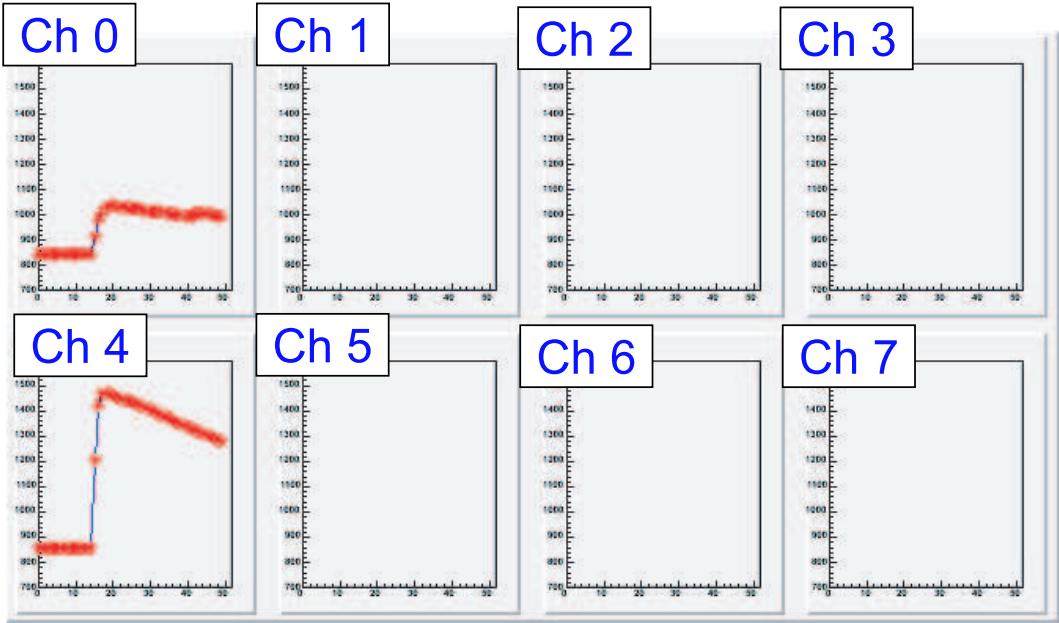


図 3.8: 1 光子におけるコンプトン散乱と光電吸収の波形 [17]

3.2.2 ヒストグラム生成

PDC/SAS 部ではバックグラウンドレートの測定、ユニットの健康状態の管理のためにパルスハイトイストグラムを生成する。パルスハイトイストグラムの波高値を求める際に絶対値を元にしてスレッショルドを決定してしまうとノイズの増加時やオフセットの異なる各チャンネルに合わせて細かくスレッショルドを設定しなくてはならなくなってしまう。そこで PoGOLite では図 3.10 のように差分データに対するスレッショルドを設定し、波高値も差分データを元に求める。

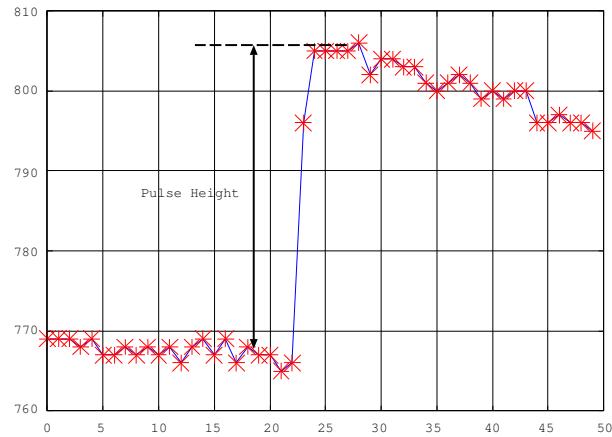


図 3.10: 図 3.3 の ch.0 波形を拡大。スレッショルド判定、ヒストグラム波高値はこの Pulse Height から求める。

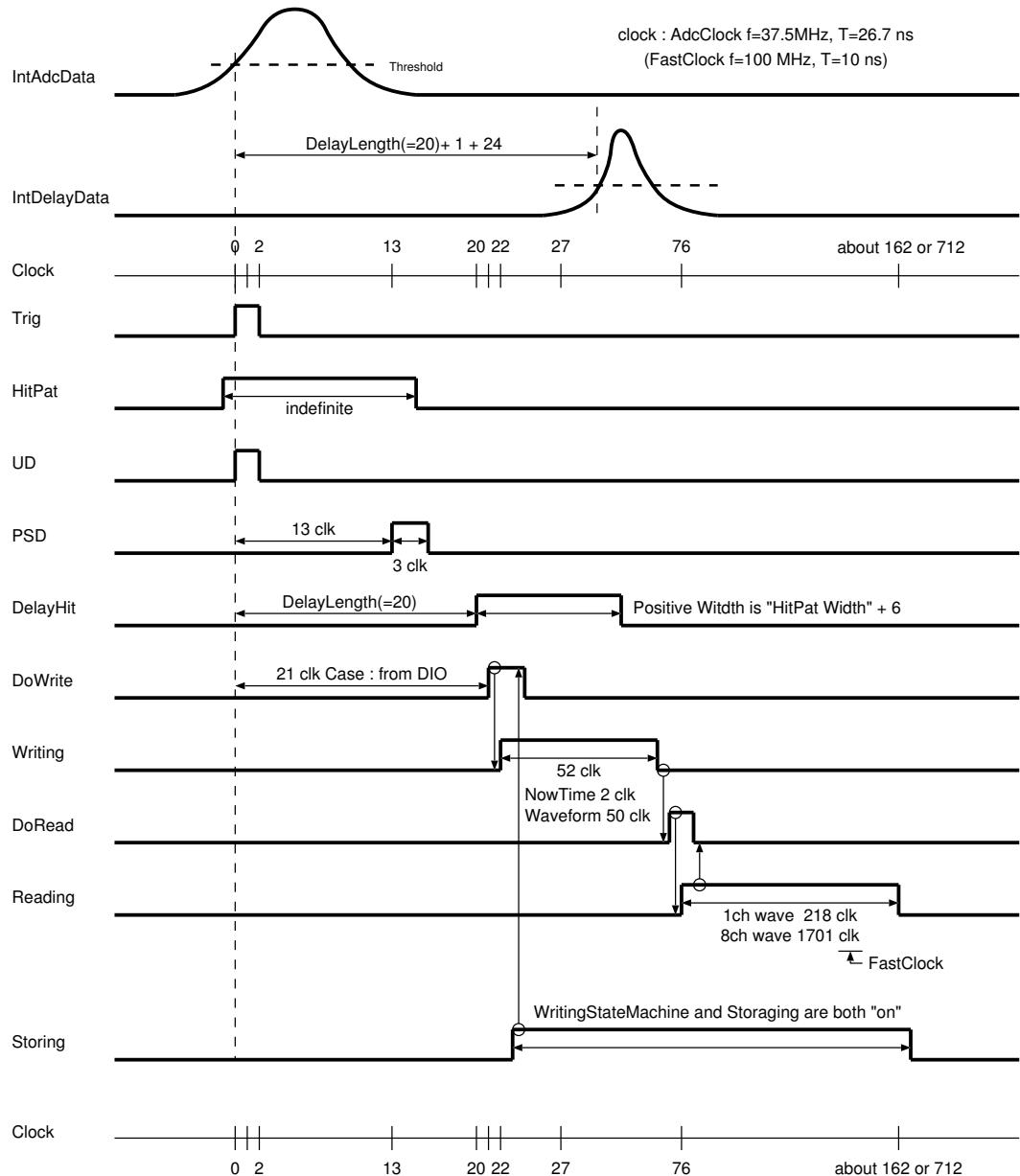


図 3.9: PoGOLite 波形保存クロックタイミング

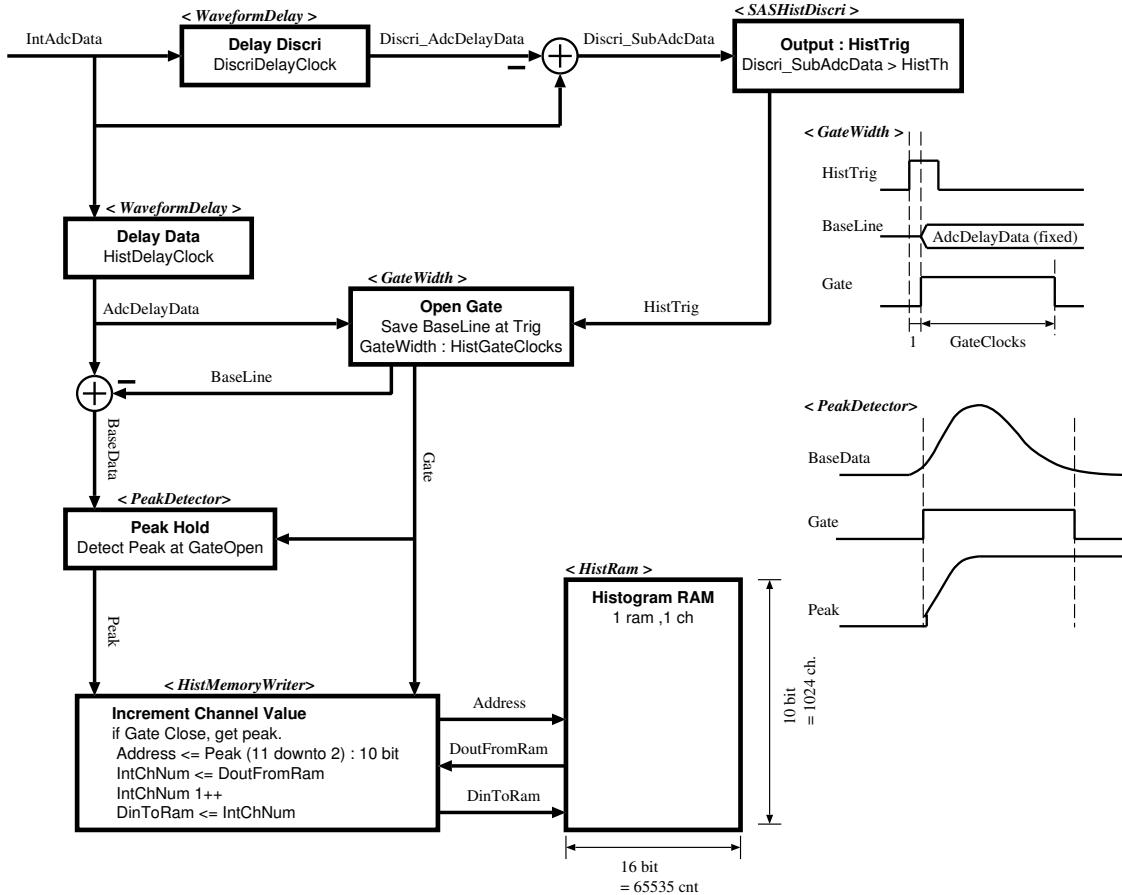


図 3.11: PoGOLite ヒストグラム生成ブロック図

パルスハイトヒストグラムは、波形保存のように複数チャンネルにまたがる同時イベントを取得する必要は無いので各ボード、各チャンネルで独立に動作する。そのためスレッシュルド判定、ピークサンプルホールド、メモリへの保存全てが各チャンネル毎に行われる。以下では図 3.11 のヒストグラムブロック図を元に説明する。

AD 変換された PMT 信号がヒストグラムモジュールに入力されると、まずモジュール内の **Discri** 部へと信号が伝えられる。このとき遅延した波形との差分をとり、オフセットは除去される。差分データがスレッシュルドを上回っていればヒストグラムを保存するよう **HistTrig** 信号が **GateWidth** コンポーネントへと出力される。**GateWidth** は **HistTrig** 信号を受け取ったら一定幅のゲートを出力する。ヒストグラムモジュールではこのゲート幅内でピーク値を決定する。ピーク値を決定するための波形データも遅延データを用いる。ゲートを開いたときの遅延データを **BaseLine** とし、遅延データから **BaseLine** を引いたデータをゲート幅内だけサンプリングホールドする。こうして得られたピーク値はメモリへと保存される。

スレッシュルド、トリガ出力のための遅延度合である **DiscriDelayClock**、ゲート幅である **HistGateClocks**、ゲート内に波形を収めるための遅延度合である **HistDelayClock** は SpaceCube で変更が可能である。初期設定では **DiscriDelayClock** が 3 clock、**HistGateClocks** が 10 clock、**HistDelayClock** が 5 clock である。37.5 MHz 動作の環境ではトリガが検出されてから波高値がヒスト

グラムに詰められるまでの全てのプロセスを 20 clock(~ 0.6 usec) ほどで完了する。

スウェーデンにある PoGOLite 検出器の実機の FADC Board に作成した FPGA プログラムを書き込み、ある PMT のノイズヒストグラムを取得した結果が図 3.12 である。以上までのヒストグラム生成におけるクロックタイミングを図 3.13 に記載する。

PoGOLite PMT Histogram

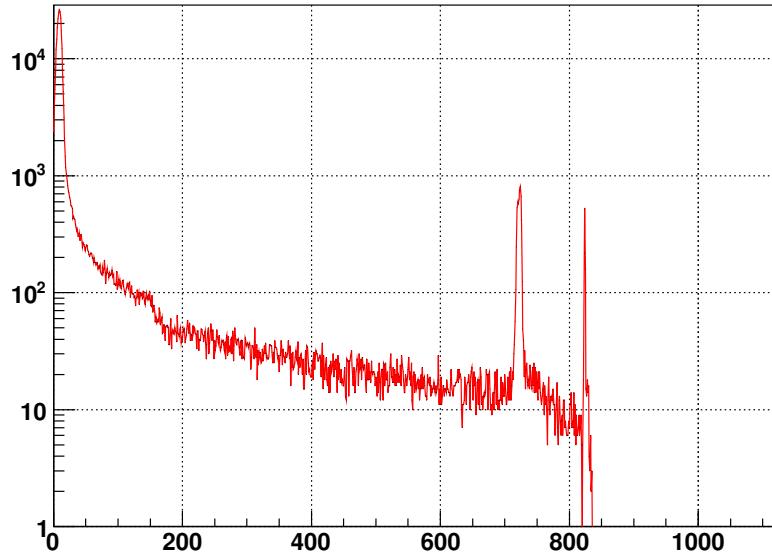


図 3.12: PoGOLite での一つの PMT におけるノイズヒストグラム。700 と 850 ch. に見られるピークは ADC の上限値に対応している。

3.2.3 FADC Board と DIO Board での開発

初期開発において、最低限必要とされてきたのが図 3.14 のように PDC 部では偏光観測が目的であったため波形の弁別、波形の取得であり、SAS 部ではアクティブシールド部としての役割である VETO 信号の出力、およびバックグラウンド変動の見積りのためのヒストグラム生成であった。しかしこのように PDC 部と SAS 部での独立した開発は手間がかかり、デバッグも二度手間となってしまう。また PDC 部でもヒストグラム生成を実装できれば主検出部におけるイベントデータの動向を追えることになる。これによってノイズの増加度などを知ることができ、各ユニットにおける状態管理が容易になる。また SAS 部の VETO 信号は PDC 部で用いる波形弁別モジュールの Trigger と HitPattern スレッシュルドを最大にすれば Trigger と HitPattern は出力されず、BGO 波形を検出する Pulse Shape Discri のスレッシュルドをほぼ 0 にすれば VETO 信号出力モジュールとして用いることができる。そのため PDC/SAS 部共通回路を作成する。

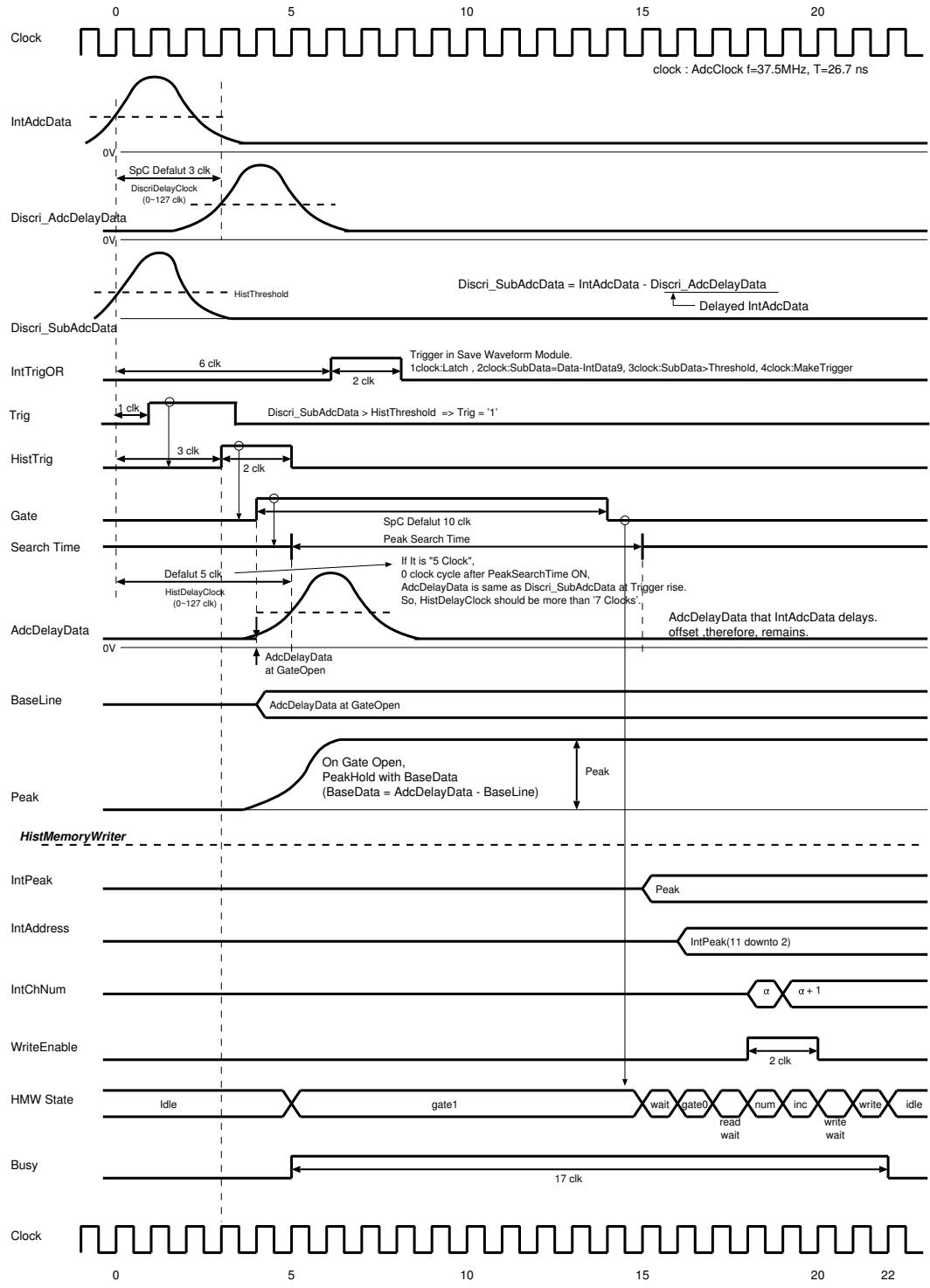


図 3.13: PoGOLite ヒストグラム生成クロックタイミング

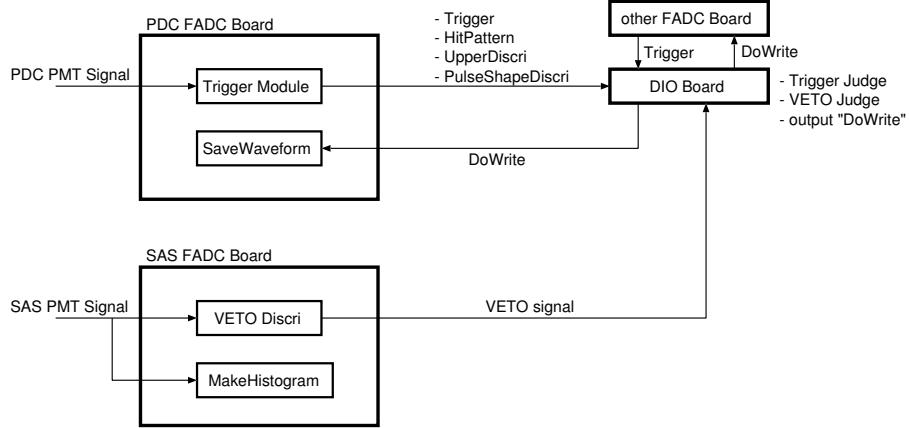


図 3.14: PoGOLite 開発の初期段階におけるセットアップ図

波形保存、ヒストグラム生成などのモジュールを実装する FPGA は PDC 部と SAS 部ともに Xilinx 社製の XC3S1000 であり、Slice 数は 7680、1 Slice 中に LUT は 2 個あるため LUT 数は 15360、BlockRAM 数が 24 である。PDC 専用の回路ではこの Slice 数が 4359 (56 %)、LUT 数が 5992 (39 %)、BlockRAM 数が 15 (62 %) を消費し、SAS 専用の回路では Slice 数 3875 (50 %)、LUT 数 5005 (36 %)、BlockRAM 数 24 (100 %) を消費している。LUT 数は両回路共に 50 % 到達していないため共用回路にしても容量の上では問題ない。BlockRAM は SAS 部だけで 100 % 消費しているが、RAM 機能は FPGA 既存の BlockRAM を用いるだけではなく、Slice を組み合わせて擬似的に RAM を作成する分散 RAM を用いることで SAS 部と PDC 部の独立した RAM を実現できると思われる。

私は SAS 部で用いる遅延用 RAM を分散 RAM にし、波形保存とヒストグラム生成が干渉せず、別用の各種トリガ信号出力モジュールを組み合わせた回路を開発した。作成した回路の概略図が図 3.15 である。

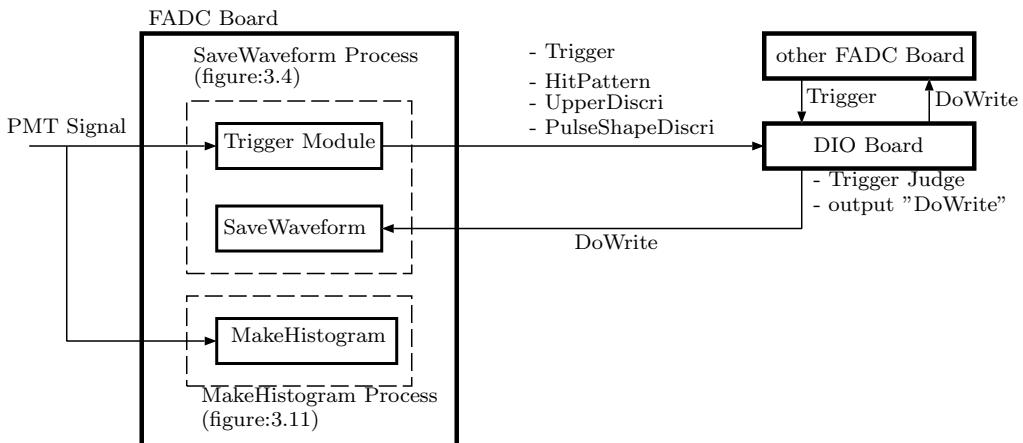


図 3.15: FADC Board における PDC/SAS 共用回路

Trigger Moduleにおいてトリガ、各種弁別用の信号を DIO Board へと出力し、DIO Board から DoWrite 信号を受け取ったのち、波形保存モジュールが稼動する。またそれとは独立にヒストグラム生成モジュールが動作する。PDC/SAS 共用回路および比較のため PDC 部・SAS 部専用回路における FPGA の使用容量を表 3.3 に記載する。SpaceWire での通信バスモジュールなど PDC と SAS の共通部分もあったため LUT は両回路の和よりも若干減ると思われたが分散 RAM を用いたことでそれ以上に増えている。私が作成したコードで 1 つの XC3S1000 内に、PDC と SAS で必要な機能を減らすこと無く全てセットすることができている。

	PDC	SAS	PDC/SAS
Slice	4359/7680 (56 %)	3875/7680 (50 %)	7030/7680 (91 %)
LUT	5992/15360 (39 %)	5005/15360 (36 %)	12145/15360 (79 %)
BRAM	15/24 (62 %)	24/24 (100 %)	23/24 (95 %)

表 3.3: PDC/SAS における FPGA(XC3S1000) での使用容量

この新しい共用回路をスウェーデンにある実際の PoGOLite FADC Board に書き込み、PDC/SAS の 91 ch、中性子シンチレータの 1 ch. の計 92 ch. ヒストグラムを取得した。図 3.16 がそのヒストグラムデータである。上部 8×8 が PDC 部で、下部 4×8 が SAS 部と中性子シンチレータである。用いる PMT 信号は 92 ch. なので 4 ch. 余る。そのため図の PDC 部で 3 箇所、SAS 部で 1 箇所にデータが全く無い場所が存在する。図 3.12 のような PMT のノイズヒストグラムが全 PMT において正常に取得できている。

3.3 中性子シンチレータとそれに向けた開発

3.3.1 大気バックグラウンド

PoGOLite の観測高度は～40km と高く、バックグラウンド源である宇宙線が大量に飛び交っている。宇宙線の起源はおよそ以下の 3 種類に分けられる。

1. 一次宇宙線
2. 二次宇宙線
3. SAA(South Atlantic Anomaly) 由来の荷電粒子

一次宇宙線とは地球磁気圏外から直接入射してくる宇宙線のこと、CXB(Cosmic X-ray Background: 宇宙 X 線背景放射)、陽子、電子・陽電子などが主な成分である。二次宇宙線はこの一次宇宙線と地球大気が相互反応することで新たに生みだされた粒子である。様々なものが生成されるが、特に注意すべきものはアクティブシールドでは除去しづらい中性子である。SAA とはブラジル上空にある南大西洋地磁気異常帯のこと、こここの帯域ではエネルギーは低いものの大量の荷電粒子(主として陽子)が流れ込んでくる。そのフラックスは他の領域に比べて 1000 倍ほどである。SAA の荷電粒子は衛星においては重大なバックグラウンドとなるが、PoGOLite は北極圏をフライトするためこの領域を通過しない。よってここでは深く述べないこととする。

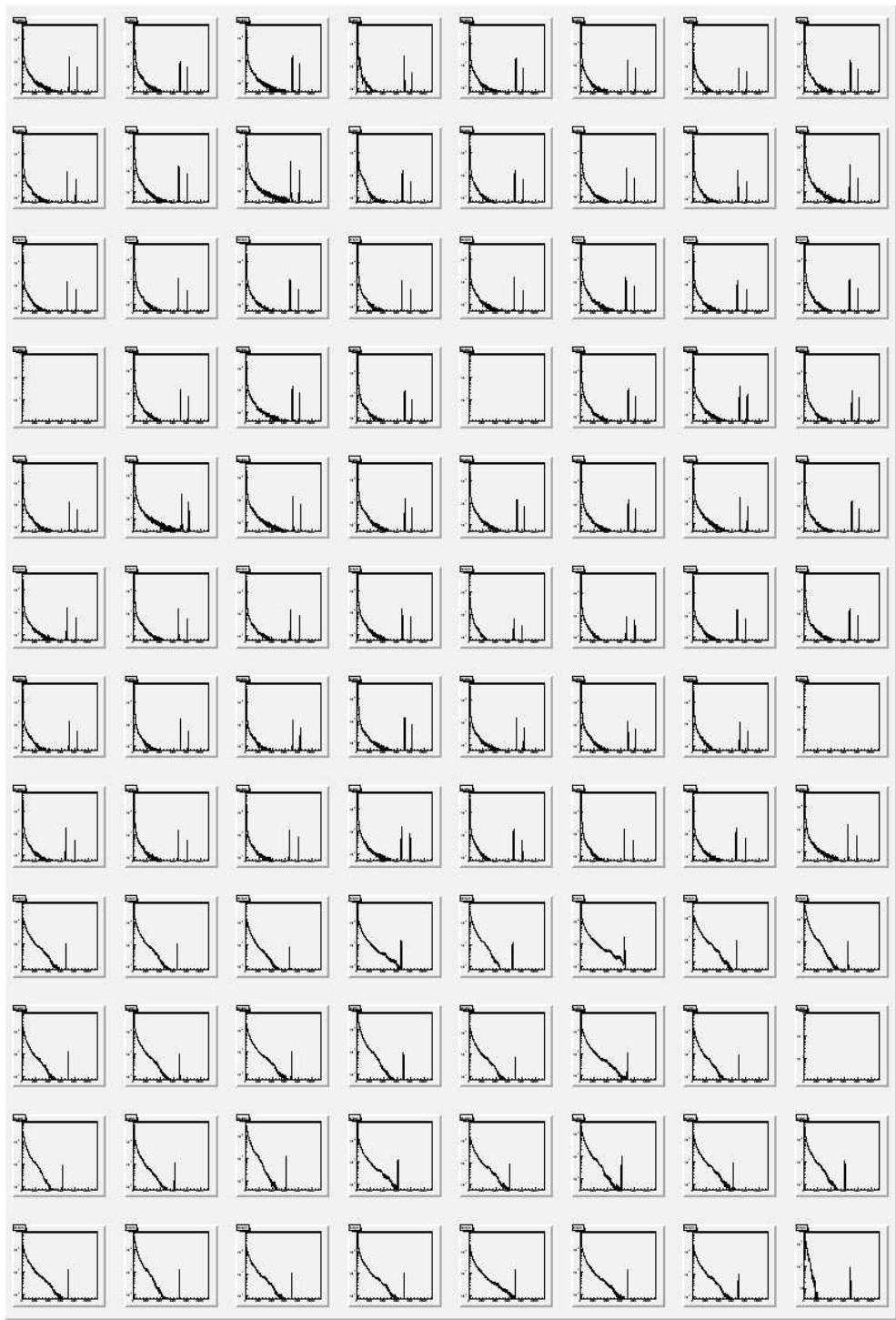


図 3.16: PoGOLite での PDC/SAS/中性子シンチレータを全て含む計 92 本の PMT のノイズヒストグラム

より良い感度の観測を行うためには回路系ノイズを落とすことに加え、このような宇宙線由來のバックグラウンドの除去が必須である。そのために前述したように、PoGOLite では Slow プラスチックシンチレータや BGO アクティブシールドによる反同時係数法を用いたバックグラウンドの除去を行い、 $2.0^\circ \times 2.0^\circ$ の視野角から入射し、Fast シンチレータでのみ反応した硬 X 線・軟ガソルマ線信号のみを選択することでバックグラウンドを大きく除去する。しかしバックグラウンドのうち、X 線・ガソルマ線や荷電粒子はアクティブシールドなどを用いることでほぼ 100 % 除去できるが、二次宇宙線の中性子は荷電粒子ではないため電磁気的な作用を受けず、他の放射線と比較しても物質の透過度が高いため有意に除去することは困難である。図 3.17 に、シミュレーションによる PoGOLite パスファインダー検出器の 20 - 100 keV の帯域における天体信号(かに星雲)とバックグラウンドの予想スペクトルを記載する。これによるとバックグラウンドは 100 mCrab 程度であり、天体信号の 1/2 にも達する。その中でも中性子バックグラウンドがバックグラウンドの主要成分であることから中性子バックグラウンドの変動を追えれば、検出器がより良い感度を達成できる可能性がある。このため、PoGOLite パスファインダーモデルには熱中性子に感度を持つ LiCAF(Eu) 中性子シンチレータ検出器を搭載している。

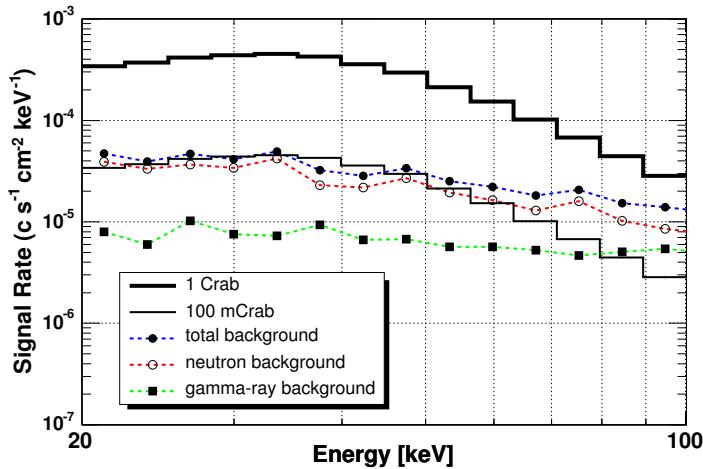


図 3.17: シミュレーションによるバックグラウンドスペクトル [16][4]

3.3.2 中性子シンチレータ

中性子とは質量が $1.6749 \times 10^{-24} \text{ g}$ でほぼ陽子と同じ質量であり、直徑が約 1 fm の非荷電粒子のことである。原子核外では例外を除いて不安定であり、平均寿命 886.7 ± 1.9 秒で陽子と電子および反電子ニュートリノに崩壊する。この中性子は速度によって表 3.4 のように分類される。

約 0.025 eV	熱中性子
0 ~ 1000 eV	低速中性子
1 ~ 500 eV	中速中性子
500 keV 以上	高速中性子

表 3.4: 中性子の速度による分類

中性子は上述のように電磁気的な作用は受けないため原子核と直接作用する反応を利用して検出する。この作用は散乱と吸収に分けられ、散乱は弹性散乱と非弹性散乱に、吸収は捕獲と核変換に分類される。高速中性子は弹性散乱、非弹性散乱が支配的であり、弹性散乱は物理的な衝突なため原子核が小さいほど信号を検出しやすい。逆に熱中性子・低速中性子は核捕獲、核変換反応が支配的である。

詳しくは米谷卒論[8]に書かれているため、ここでは吸収の核変換反応について述べる。核変換反応とは原子核が中性子を吸収し、複合核化、粒子を放出して新たな原子核に変換される反応のことである。原子番号の小さい原子核(軽核)の場合、ガンマ線があまり反応しないため、中性子のみを感度良く検出できる。また、クーロン障壁が低いために反応によるエネルギーQ値が正となって(発熱反応として)表れる。これは中性子検出の反応では重要となる。以下の式3.1から式3.3に代表的な核捕獲反応式を記載する。式右側にあるエネルギー値はこの中性子捕獲の結果起こる反応から放出されるエネルギーを決定する反応のQ値である。



このうち、式3.1の ${}_2^3 He$ によるものは ${}_5^{10} B$ に比べて反応断面積が大きいため、また式3.3の ${}_5^{10} B$ は天然存在比が20%と容易に入手することができることから、これらの中性子反応は大型の中性子検出器としてよく利用されてきた。しかしながら ${}_2^3 He$ や ${}_5^{10} B$ を含む物質はガス状か液体状であるためスペースに限りがあり、温度環境も厳しい観測機器に搭載することは難しい。最近になって ${}_3^6 Li$ がドープされた潮解性もない大型の結晶シンチレータであるLiCaAlF₆(LiCAF)が開発されたため、式3.2を用いた核変換作用でありながら固体のシンチレータを用いることが可能になった。 ${}_3^6 Li$ にはQ値が4.78 MeVと大きいという長所もある。これにより限られたスペースでありながら十分な中性子捕獲の断面積を確保できると期待され、PoGOLiteではこのLiCAFシンチレータを熱中性子検出器として使用する予定である。図3.18にこの核捕獲反応の高速中性子における核変換反応の反応断面積を ${}^3 He(n, p)$ と ${}^6 Li(n, \alpha)$ について記載する。図よりエネルギーが増加するにつれ反応断面積は小さくなっていることが分かる。よってPoGOLiteでは検出器全体を覆っている15 cm厚のポリエチレンシールドの中にこの中性子検出器も入れることにより、もともとは高速中性子だったものが熱化され熱中性子になったフラックスも検出する。

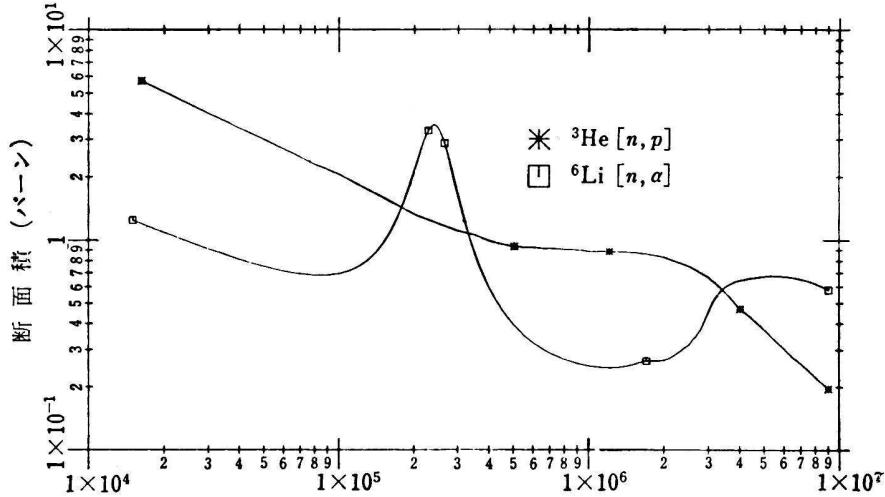


図 3.18: 高速中性子領域における ${}^3\text{He}(n, p)$ と ${}^6\text{Li}(n, \alpha)$ の反応断面積の図。横軸に中性子のエネルギー [eV]、縦軸は反応断面積 (バーン) である。[8]

PoGOLite における中性子シンチレータは光量の大きな LiCAF (Eu 2% ドープ) 結晶と時間応答性の高い LiCAF (Ce 2% ドープ) 結晶の二つが候補に上がっている。結晶の育成の都合により、厚さは LiCAF(Eu) が 5 mm、LiCAF(Ce) が 26 mm である。LiCAF(Eu) による時定数は ~1600 ns で、LiCAF(Ce) による時定数は ~40 ns である。図 3.20 と図 3.21 に LiCAF(Eu) と LiCAF(Ce) のシンチレーション光波形 (FADC Board のプリアンプ出力) を記載する。両方とも比較のため BGO 結晶のシンチレーション光波形を載せていている。

中性子シンチレータは熱中性子のみを効率的に検出したいため、PDC 部同様に BGO 結晶を用いたフォスイッチ構造を採用する。原子番号の小さい LiCAF 中性子シンチレータはガンマ線とは光電吸収反応をしくいが荷電粒子は防げない。荷電粒子は衝突距離によって蛍光量が異なるため、シンチレータへの入射方向により連続的なエネルギーを生成し、熱中性子を捕獲した際のピークがスペクトル上で埋もれてしまう。そこで時定数の異なる BGO 結晶で挟むことにより、BGO でも反応する荷電粒子を波形弁別で除去する。こうして熱中性子と反応したピークのみを感度良く検出できると考えられる。図 3.19 に各結晶の写真と中性子シンチレータユニットの構造図を載せる。様々な議論の末、PoGOLite パスファインダーフライトでは光量の大きい LiCAF(Eu) を用いたシステムとして構築する。

図 3.21: BGO と LiCAF(Ce) シンチレーション
光波形 (FADC Board のプリアンプ出力)[20]

3.3.3 中性子シンチレータ用のデータ取得システム構築

中性子シンチレータのデータ取得システム (DAQ) で重要なのは波形弁別により純粋な中性子由来のデータを取得することであり、これらのシステムは PDC 部専用回路で用いられていた Discri Module でほぼ流用することができる。しかし中性子シンチレータ DAQ を FADC Board 上で実装する上では以下の二つを改修しなくてはならない。(1) 一つ目が中性子シンチレータの DAQ を独自に動作させる必要があるということである。中性子シンチレータからの信号は他のユニットとの同時処理が必要な偏光観測には用いられず、中性子のカウント数のみをモニタすれば良いため、DIO Board を経由した DoWrite 信号を必要としない。ただし、8 チャンネル処理できる FADC

Board を 1 枚まるまる 1 チャンネルの中性子シンチレータ用にすることは FADC Board が余分に増え、電力・スペースが増加するため無駄である。そのため FADC Board 上で中性子シンチレータ用の回路と他のユニットと回路を共存させるように設計する必要がある。(2) 二つ目は中性子シンチレータ用に ADC のサンプリング周波数を変更する必要があることである。図 2.23 にあるように、BGO 結晶 ($\tau \sim 300$ ns) は Fast シンチレータ ($\tau \sim 2$ ns) と比べても時定数は長いが、LiCAF(Eu) は $\tau \sim 1600$ ns と更に長い。そのため従来のサンプリング周波数 37.5 MHz ではなく、その 1/6 である 6.25 MHz までサンプリング周波数を間引く必要がある。つまり中性子シンチレータ用には、37.5 MHz で取得した ADC 値を 6 回に 1 回のみ更新する。

中性子シンチレータ DAQ では PMT 信号が入力されればまず上述の信号の間引を行う。この間引いたデータは 6.25 MHz に最適化された Trigger Module へと運ばれる。ここで他のユニット同様スレッショルド判定を行い、トリガを出力する。他のユニットではこのトリガは DoWrite 信号を出力する判定のため DIO Board へと運ばれる。しかし中性子シンチレータは独立に動作させるために DIO Board からの DoWrite 信号は必要としていないので、自身の FPGA 内で中性子シンチレータによるトリガから中性子シンチレータについてのみ DoWrite 信号を即座に出力する。この DoWrite は中性子シンチレータのための DoWrite なため、DIO Board からの DoWrite とは別物である。こうして中性子シンチレータ DoWrite により中性子シンチレータの波形が他のユニットとは独立に保存される。この波形はトリガが出力される波形の 2.4 usec 前から 6.25 MHz で 50 clock 分 (8 usec) 保存される。波形のデータフォーマットは図 3.6 と全く同じである。どのボードのどのチャンネルが中性子シンチレータであるかを知っていればオフラインでの中性子シンチレータ解析が可能である。完成した回路のブロック図が図 3.22 である。

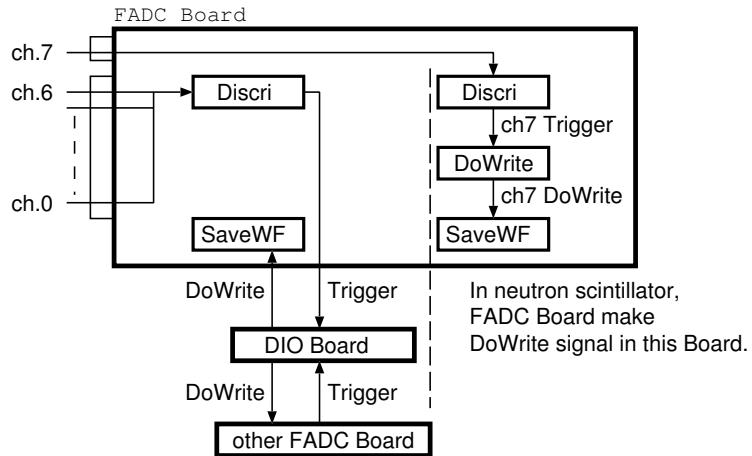


図 3.22: 中性子シンチレータのための FPGA ブロック図

中性子シンチレータは SAS 部の FADC Board と共にしている。ch.0 から 6 までが SAS 部であり、ch. 7だけが中性子シンチレータである。テストパルスを二つに分け、ch.0 と ch.7 に入力し、その波形を取得したものが図 3.23 である。この二つの波形は同じ信号を入力したものであるが、ch.0 と比べて ch.7 はクロックが 6.25 MHz と 1/6 に遅いため波形が広い時間にわたって保存されていることが分かる。

LiCAF(Eu) と BGO 結晶の波形から PDC 部における波形弁別 (図 3.5) とほぼ同様の弁別を行う

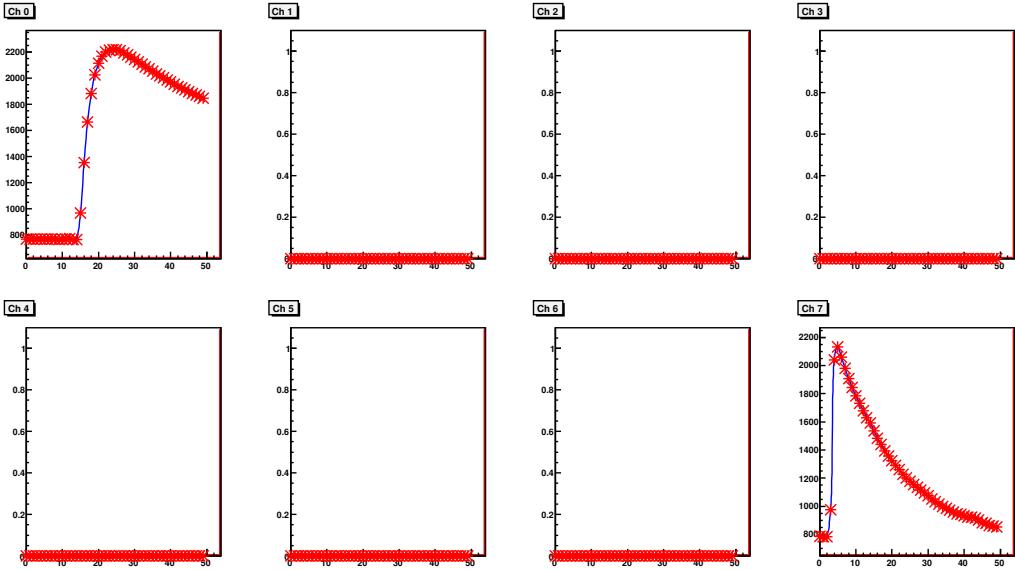


図 3.23: 同じ波形を入力した場合の ch. 別波形処理の様子

ここでシンチレータ別に波形を分けることができる。ゲートを短い Fast と長い Slow に分け、それぞれのピークを FastPeak、SlowPeak とする。こうして得られる中性子シンチレータの二次元ヒストグラムが図 3.25 である。中性子線源に ^{252}Cf 、ガンマ線源に ^{137}Cs を用いている。図の二次元ヒストグラムから LiCAF 由来と BGO 由来の二本の線に分かれていることが確認される。BGO 結晶は時定数が短いため傾きが小さいブランチにピークが含まり、逆に遅い LiCAF(Eu) 結晶は傾きが大きいブランチにピークが含まれる。これをブランチにそって各シンチレータごとに分け、一次元ヒストグラムに直したもののが図 3.25 である。BGO でも検出された信号は除去され、LiCAF によって熱中性子が効率良く検出していることが確認できる。

図 3.25: BGO と中性子シンチレータ LiCAF(Eu) のイベントセレクションヒストグラム [20]

第4章 Astro-H HXI/SGD用BGO処理回路の開発

4.1 研究目的

HXI/SGD では信号ノイズを軽減してより低いスレッショルドを実現するために、APD 信号をデジタルフィルタに通して APD の信号成分を通す。デジタルフィルタにはノイズ成分をカットする有限インパルス応答フィルタ (FIR フィルタ) を利用することを予定している。HXI/SGD では一つの APD 信号に対してヒストグラム取得と主検出部への VETO 信号出力の役割がある (図 4.1)。VETO 信号出力は処理は粗いが早い時間応答を示す FastVETO と、処理は細かいが少し時間がかかる HitPattern の二つが存在する。

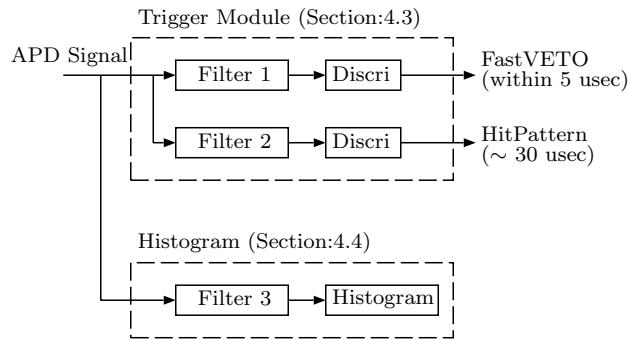


図 4.1: APMU での VETO 信号、ヒストグラム生成ブロック図

FIR フィルタは遅延器が多いほど、期待する周波数特性を再現できる。ただし人工衛星に搭載する装置であるため、使用できる資源はそもそも限られており、その中でもなるべく必要最低限に抑えたい。また手本となるアナログフィルタの周波数特性は矩形波の形ではなく、滑らかな関数をしているため、それほど多数の遅延器がなくてもある程度の形を再現することが可能である。よって、現時点では HXI/SGD 検出器では表 4.1 のように遅延器数が 8 ないし 16 での衛星処理を考えている。FIR フィルタはその特性上、遅延器数の真ん中で最も感度を持つ。つまり遅延器数が 8 ならば 4 番目の遅延器におけるフィルタ係数が最も重みがあるということである。その最も重みがある係数にピーク値が辿り着くまで遅延を重ねなければならないので、信号が入力されて出力結果にそのフィルタの影響が現れるまで遅延器数の半分のクロック数は待たないといけないということになる。HXI/SGD の BGO アクティブシールドは APD 素子がバックグラウンド信号を検出した場合、主検出部でバックグラウンド信号かどうかを判定する必要があるため、なるべく早く VETO 信号を出力しなくてはならない。つまり遅延器数は時間応答の観点からも少ない方が良い。FIR フィルタの係数もソフトウェアで開発を行うときは誤差を気にしなくてすむほど桁数の

多い小数を用いているが、HXI/SGD 検出器の場合には一つの FPGA で最大 13 本の APD 信号を処理しなくてはならず、あまりに精度の高いフィルタ係数は扱えない。そのため APMU での FIR フィルタは現在以下のようにになると想定して開発している。

	遅延器数	フィルタ係数
FastVETO	8 tap	-1, 0, +1
HitPattern	16 tap	-256 ~ +255 整数

表 4.1: HXI/SGD で用いる FIR フィルタの条件

FastVETO はデータ信号が入力されて 5 usec 以内に出力することが要求されている。実機での動作サンプリング周波数 1 MHz の場合、遅延器数が 8 なら最大でも 4 usec で出力されると想定されるため、出力タイムラグに関しては実機での要求性能を満たす。HitPattern は遅延器数が多いため複雑な周波数特性をとることが可能である。これによって FastVETO でのフィルタでは落としきれなかったノイズ信号を除去する。

本研究ではそもそもデジタルフィルタが微弱な信号に効果があり、低いスレッショルドでもパルスを検出できるのか、またこの条件での FIR フィルタで低いスレッショルドのヒストグラムを達成できるのかを調べる。しかし、これまで X 線衛星で本格的なデジタルフィルタが FPGA に搭載されたことがなく、どの周波数帯を通過域とすればいいのか、またはそれは各周波数帯に対してどの程度の透過度であればいいのかということもまだ調べられていない。フィルタ係数や遅延器数も FPGA での容量や VETO 信号の出力時間制限から決定しているが、これが FIR フィルタとしてどの程度の性能を発揮するのかを調べる必要がある。私はそれらをソフトウェアで組み立て、遅延器数の違い、フィルタ係数の精度、任意の周波数特性の実現を行ってそれぞれの検証を行う。次に FPGA 上で FIR フィルタ実装をし、実際の動作環境を模擬して BGO 結晶と APD 信号を用いたセットアップに線源を照射し実証実験を行う。

以下 4.3 節では VETO 信号の検出効率試験においてデジタルフィルタが低スレッショルドに効果があるかどうかを容易に製作できるデジタルフィルタを元に調べる。4.4 節では FPGA で波形を取得し、ソフトウェアでより複雑な周波数特性を取ることが出来る FIR フィルタを通してヒストグラムでフィルタの効果を試験する。4.6 節では実機で用いる ADC 素子 ADC128S102 と同じく実機同様のプリアンプを用いて FPGA 上で FIR フィルタを通し、ヒストグラムにおいて低スレッショルドが達成できるかを試験する。

4.2 ディジタルフィルタ

一般にフィルタとはある特定の成分を弱める、あるいは強める効果を持つ機能のことを指すが、本研究でフィルタをかける対象は時間で変動する電気信号なので便宜上、フィルタを「特定の周波数成分を弱める、あるいは強める効果を持つ機能のこと」と定義する。

フィルタはその通過帯域によって大きく四種類に分けることが出来る。

- 低域通過フィルタ (Low Pass Filter : LPF)
- 高域通過フィルタ (High Pass Filter : HPF)

- 帯域通過フィルタ (Band Pass Filter : BPF)
- 帯域除去フィルタ (Band reject Filter : BRF)

フィルタにも連続信号を扱うアナログフィルタと離散信号を扱うディジタルフィルタとある。アナログフィルタは最も簡単なフィルタを抵抗とコンデンサのみで実装できるなどかなり手軽に扱うことができ、簡単な処理の場合は低電力ですむ。また 0 Hz から無限の周波数まで全ての周波数帯域に影響を持つため、AD 変換で問題となるエイリアシング (Aliasing) もサポートすることができる。しかし抵抗とコンデンサ、あるいはオペアンプなどの能動素子はいったん実装すると変更が難しく、要求する処理が複雑になればディジタルフィルタよりもコストが大きくなり、回路の実装面積も広がってしまう。ディジタルフィルタの利点は計算可能であれば原理的にどのような周波数特性であっても実現できる。ディジタル信号を扱うのでデータ処理部分に関しては経年劣化をおこさず、安定したデータ処理が行える。初期コストは AD 変換器や LSI 費用などアナログフィルタに比べても割高だが、複雑な周波数特性になども消費電力や回路面積が大きく変動しないという利点がある。ただし上述したようにエイリアシングやナイキスト周波数以上の周波数帯域はディジタルフィルタで対処できないという欠点もある。

HXI/SGD では APD 素子の出力信号に簡易的なアナログフィルタとともに、限られた回路面積と回路パラメタの変更を考慮しつつ更にノイズを落とすために後段でディジタルフィルタを用いる。一般的なディジタルフィルタの話は付録:A に記載する。

4.2.1 移動平均フィルタと差分フィルタの表現

FIR フィルタは複雑な周波数特性を取ることが可能である一方、その扱いは複雑で簡易的にフィルタの評価を行うには向いていない。そこでまず FPGA 上で容易に LPF を実現できる移動平均フィルタと HPF である差分フィルタを用いてディジタルフィルタの効果を調べた。それらの一般的なブロック図とその係数導出式、そして周波数特性を以下に記載する。なお、これら移動平均フィルタと差分フィルタの詳細な説明は付録:A.2 に載せている。

移動平均フィルタ (LPF)

移動平均フィルタは変化が激しい場合はその変化がなまり、変化がなければ出力されるデータもフィルタを通す前と変わらない。よって低域通過フィルタである。入力離散データを $x[n]$ ($n = 0, 1, \dots$)、出力離散データを $y[n]$ とし、フィルタの処理方法を表す図であるブロック図を用いると図 4.2 のようになり、フィルタの処理方法式である差分方程式を用いると移動平均フィルタは式 4.1 となる。ここで z^{-1} は 1 clock データを遅らせる遅延器のことと M は遅延器の数を表す。

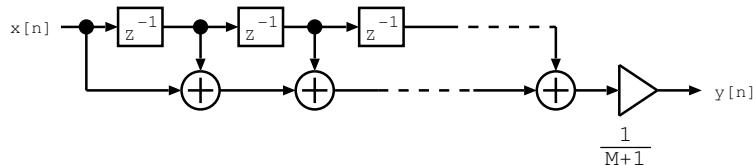


図 4.2: 移動平均フィルタのブロック図

$$y[n] = \frac{1}{M+1} \sum_{m=0}^M x[n-m] \quad (4.1)$$

この差分方程式から伝達関数 $H(\omega)$ を求めるとき周波数特性は式 4.2 となる。なお、このとき $\omega = 2\pi f$ 、 $T = 1/f_s$ 、 f_s :サンプリング周波数である。

$$|H(\omega)| = \frac{1}{M+1} \left| \frac{\sin\left(\frac{M+1}{2}\omega T\right)}{\sin\left(\frac{\omega T}{2}\right)} \right| \quad (4.2)$$

この周波数特性からサンプリング周波数を 1 MHz として遅延器数 M を変更した図が図 4.3 である。低い周波数帯域ほど透過しやすく、また遅延器数が増えるにつれて透過する帯域が狭まっていくことが分かるが、高帯域も完全には除去できない。

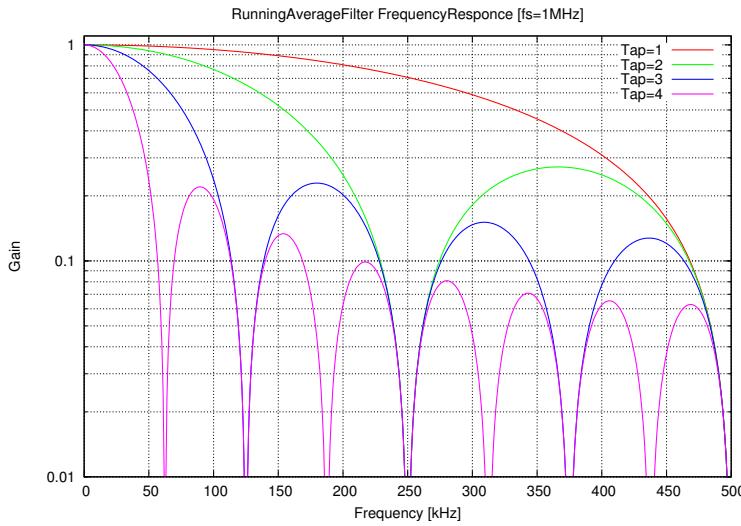


図 4.3: 一般的な移動平均フィルタの遅延器数を変えた場合の周波数特性 ($f_s = 1\text{MHz}$)

差分フィルタ (HPF)

差分フィルタはあるデータ点とその何点か前のデータ点との差分を出力する。そのため、変化が無い場合は出力が 0 になり、変化が大きい場合は出力も大きくなる。そのため差分フィルタは HPF となる。移動平均フィルタ同様にブロック図と差分方程式を以下の図 4.4、式 4.4 に記載する。

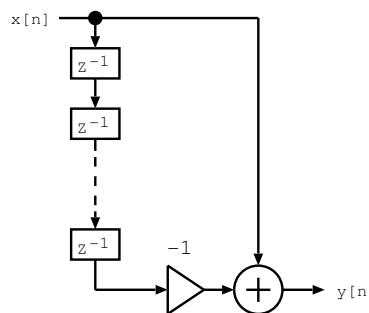


図 4.4: 差分フィルタのブロック図

$$y[n] = x[n] - x[n - M] \quad (4.3)$$

この差分方程式から伝達関数 $H(\omega)$ を求めると周波数特性は式 4.4 となる。こちらも移動平均同様に遅延器数 M にのみ依存する。周波数特性は図 4.5 となる。いずれの遅延器数であっても 0 Hz 周辺はほぼ通過しないが遅延器数が多くなってくると帯域フィルタに似てくる。

$$|H(\omega)| = 2 \left| \sin \frac{M\omega T}{2} \right| \quad (4.4)$$

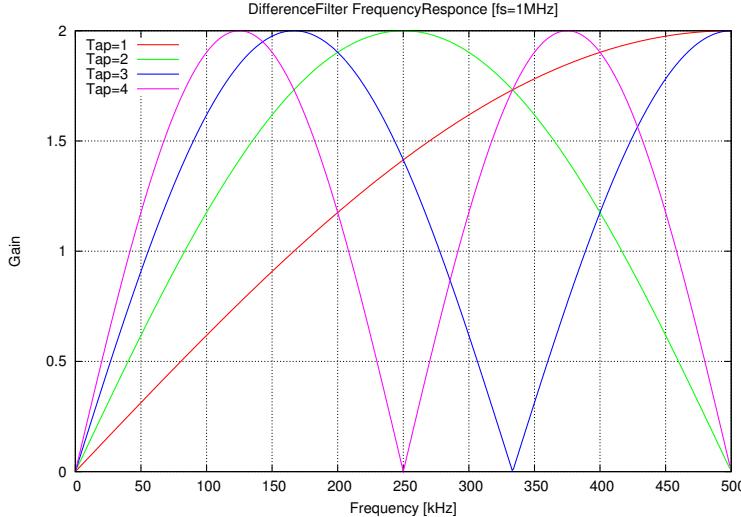


図 4.5: 差分フィルタの遅延器数を変えた周波数特性 ($f_s = 1\text{MHz}$)

4.3 ディジタルフィルタを用いた Trigger 試験

アクティブシールドにおける APMU の大きな役割として BGO 結晶を APD 素子で検出した信号の処理、VETO 信号出力出力とヒストグラム取得がある。VETO 信号は 4.1 節で述べたように FastVETO 信号と HitPattern 信号の二種類があり、FastVETO は主検出部がイベントを処理するまでに出力しなくてはならないので 5 usec 以内に出力しなくてはならない。HitPattern は 30 usec まで時間がかかるてもいいので FastVETO で拾えなかったノイズに埋もれるほどの微弱なバックグラウンド信号も検出するようとする。ここではテストパルス信号に移動平均フィルタを通し、低いスレッショルドでも VETO 信号に効果があるかを試験する。

VETO 信号の出力に関する試験はテストパルスが信号を出力した際、Trigger Module がその信号に同期して VETO 信号を出力できるかをスレッショルドを変えながら検証する。テストパルスには信号が出力されるときに同時にトリガ信号も出力するので、FADC Board が出力するトリガとテストパルスのトリガの AND を取ることで FADC Board がテストパルスの信号を処理できているか調べる。このときテストパルスの出力数に対する AND でのカウント数の比を検出効率とする。テストパルスの信号を全て拾えれば検出効率は 100 % となる。

VETO 信号の出力効率の測定はテストパルスを用いて試験を行う。ノイズレベルが 10 mV ほどであるため、テストパルスの波高値は 5 倍程の 50 mV になるように調整した。このときの波形を図 4.6 に載せる。

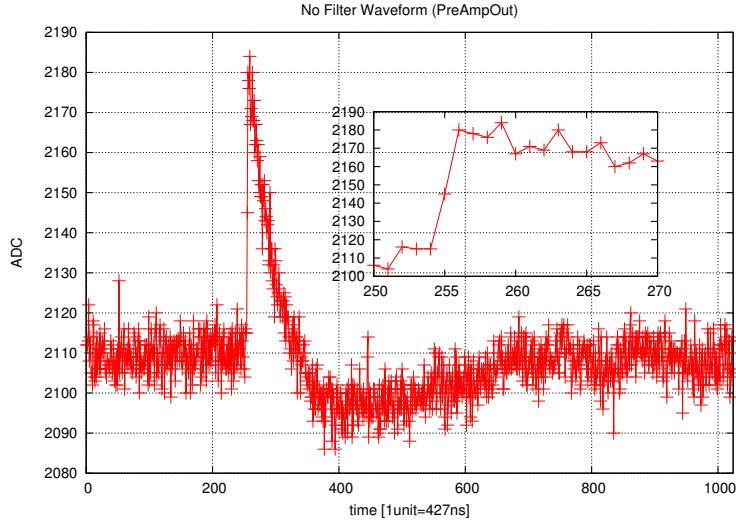


図 4.6: VETO 信号出力効率の測定で用いるテストパルス波形 ($T_{rise} = 0.5\text{usec}$, $T_{fall} = 50\text{usec}$ 、プリアンプ 580K 出力, $f_s = 2.35\text{MHz}$)

VETO 信号出力をうため、本研究ではアナログ入力があり、ディジタル入力、出力が可能な FADC Board を用いた。このアナログ入力にプリアンプ信号を入れ、VETO 信号をディジタル出力させる。信号の処理を行うためのクロックは BGO 結晶波形を処理するうえで十分な余裕がある 2 MHz サンプリング周波数で行った。FPGA で構築した VETO 信号出力のための Trigger Module を図 4.7 に簡単なブロック図として載せる。

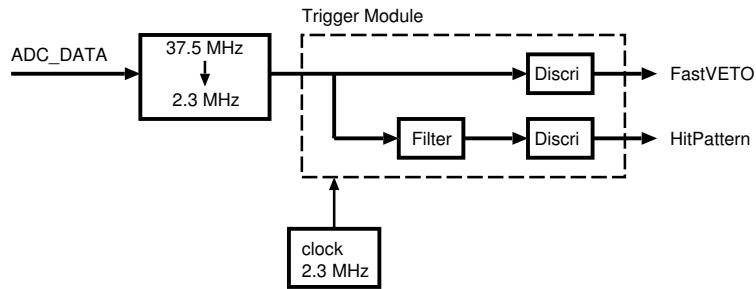


図 4.7: VETO 信号出力効率の試験における Trigger Module の全体図

このテストパルスはおよそ BGO シンチレーション光を APD 素子で受け取り、プリアンプ 580K で処理された際の波形に近いものに合わせている。立上り立ち下がりともに緩やかなため 37.5 MHz のサンプリング周波数を 2 MHz まで間引いている。この波形データを Trigger Module の Discri 部で遅延させて現在信号と遅延信号の差分を取る。この差分データでスレッショルド判定を行う。つまり図 4.8 のようなプロセスでスレッショルド判定する。差分を取ることでオフセットが除去でき、スレッショルド判定が行いやすくなっている。この判定プロセスは移動平均フィルタ有り無し

共に共通である。

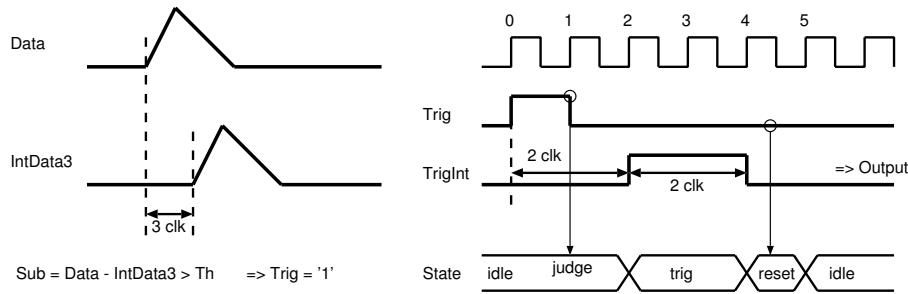


図 4.8: 図 4.7 の Discr. 部における Trigger Module スレッショルド判定プロセス

図 4.8 の左図がデータの差分を表す。差分データがスレッショルドを上回るとき Trig が'1'となる。この Trig が'1'になれば右図のプロセスが動き、一定幅のトリガを出力する。この差分は図 4.4 と同様の差分フィルタ になっており、このときの周波数特性は図 4.9 で波形は図 4.10 となる。差分の度合は図 4.6 の波形から 3 つ遅延させたデータであれば十分であるため遅延器数は 3 とした。

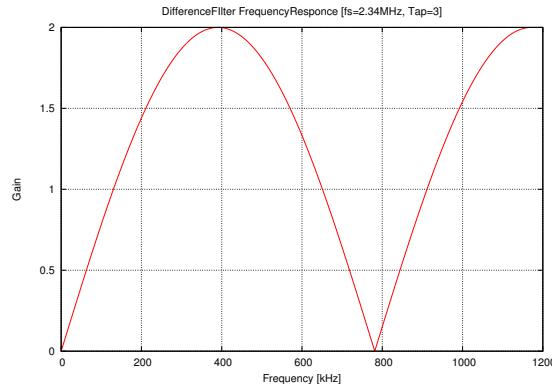


図 4.9: 差分フィルタ (Tap=3) 周波数特性

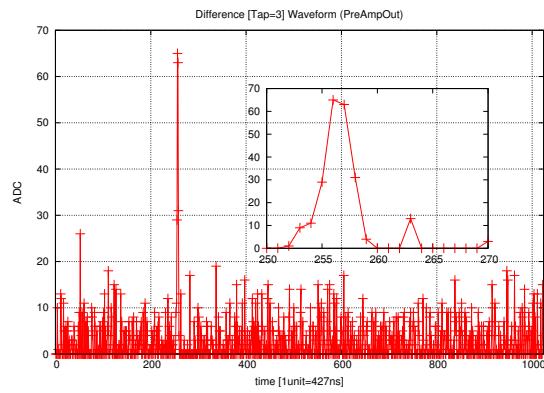


図 4.10: 差分フィルタを通した波形

一方、この差分フィルタの前段に移動平均フィルタを設置して移動平均フィルタ有無の効果を調べた。移動平均フィルタは構築が簡単で、高周波ノイズを抑えることができる。この移動平均フィルタと差分フィルタによる周波数特性は図 4.11 であり、そのときの波形は図 4.13 となる。フィルタ有りがこの波形でスレッショルド判定を行う。

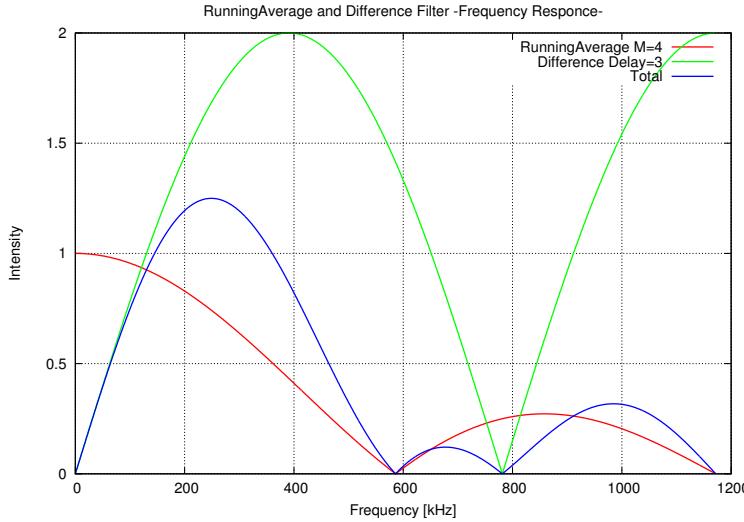


図 4.11: 差分+移動平均フィルタ周波数特性(赤:移動平均フィルタ、緑:差分フィルタ、青:移動平均と差分フィルタの合成)

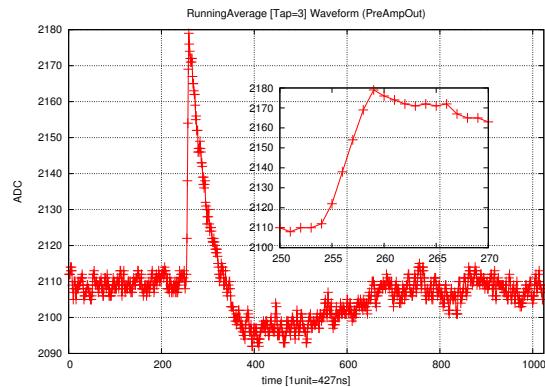


図 4.12: 図 4.11 の赤線フィルタを通した波形

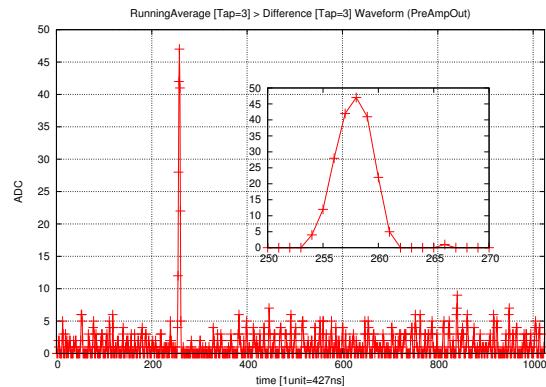


図 4.13: 図 4.11 の青線フィルタを通した波形

VETO 信号出力効率を調べるためにセットアップ図が図 4.14 である。

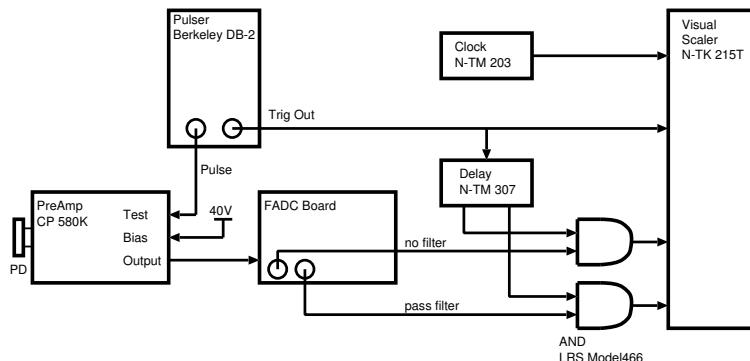


図 4.14: テストパルスを用いた VETO 信号の出力効率測定セットアップ

このFADC Boardにさきほどの図4.6のような波形を入力し、FADC BoardのTrigger ModuleのスレッショルドをSpaceCubeで変更して出力効率を測定した。移動平均フィルタがノイズに効果がある場合、フィルタ無しに比べて低いスレッショルドでも検出効率は高いはずである。用いるテストパルスはBerkeley Nucleonics社Model DB-2で、出力にランダムか一定間隔(リピート)の出力かを選べるためまずはリピート出力で調べた。バックグラウンドは～数kHz程度と想定されているため、十分ゆとりのある500Hzとバックグラウンド程度である10kHzの二つで調べた。

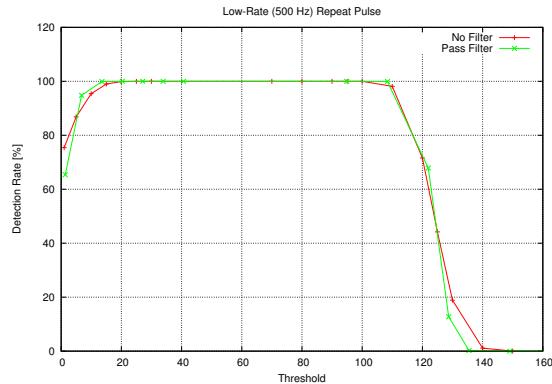


図4.15: 500 Hz リピートにおけるスレッショルドを変えた場合の出力効率

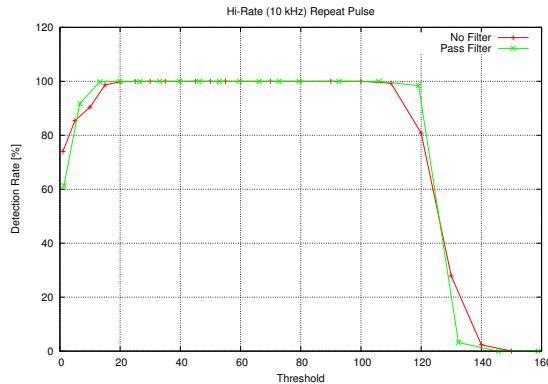


図4.16: 10 kHz リピートにおけるスレッショルドを変えた場合の出力効率

図4.15と図4.16までがリピートの周波数を500Hzと10kHzに変更したときのVETO信号出力効率で、いずれも緑が移動平均フィルタ有り、赤がフィルタ無しである。横軸がスレッショルドで縦軸がVETO信号出力効率を表している。リピートは信号間隔が一定なのでデッドタイムの間に信号が来ることがない。そのため、スレッショルドが適正であれば出力効率が100%を達成するはずである。図4.15と図4.16を見るといずれのパルス周波数でもスレッショルドが30から80程度であればフィルタの有無によらず出力効率は100%を達成できている。スレッショルドを高くしてテストパルスの波高値よりも高くなるとVETO信号が出なくなる。図を見ると移動平均フィルタ有りの方が低いスレッショルドでVETO信号が出力されなくなるが、これは移動平均フィルタだと波形がなまるためピークが低くなってしまっているからである。しかし低いスレッショルドではノイズがフィルタの効果によって減少するためフィルタ有りの方が出力効率が高い。ノイズに埋もれるような微弱な信号を取り出すことが目的のため、ディジタルフィルタは有益な効果がある。

続いてテストパルスを10kHzのランダム出力に変えて測定を行った。このときの出力効率が図4.17である。

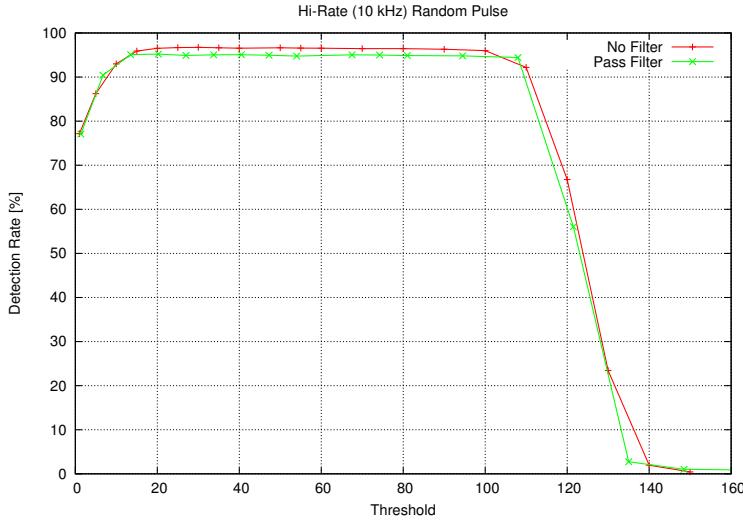


図 4.17: 移動平均 (Tap=3) フィルタ有無でのランダムパルス出力効率

ランダムパルサーでは移動平均フィルタの有無によって低スレッショルドでの効率に変化が見られなかった。これは移動平均の効果と、後述するフィルタによるデッドタイムの効果が相殺されたためと考えられる。そのため簡易フィルタである移動平均フィルタ以上の中程度なフィルタを用いることで低スレッショルドを達成できると考えられる。またランダムだとリピートでは達成できた出力効率 100 %にならないのはデッドタイム中に次の信号が入ってきたため信号を処理できなかつたからと考えられる。そこで意図的に連続信号を作り、それによって出力効率がどうなるかを検証する連続パルス試験を行い、デッドタイムがどのくらいかを調べた。このときのセットアップが図 4.18 となる。

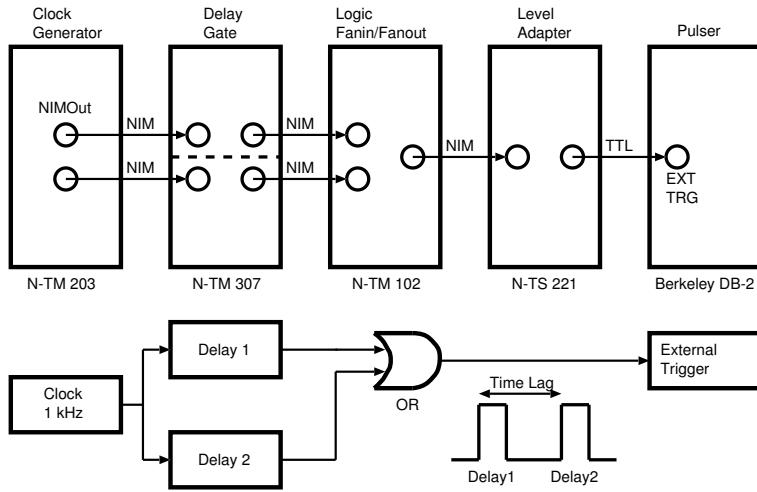


図 4.18: 連続パルス時の検出効率の測定セットアップ

テストパルスには外から入ってきたトリガ (EXT トリガ) を元にパルスを出力する機能があるため、EXT トリガに連続信号を入力すれば連続パルスが出力される。連続信号の作成には NIM モジュールのクロックジェネレータ N-TM203、ディレイゲート N-TM307 を用いる。クロック信号

を遅延の度合が異なる二つのディレイゲートに通し、その出力信号を OR することで連続信号を生成する。これによって生成した信号を EXT トリガとし、あとは図 4.14 同様に VETO 信号出力効率を求める。連続信号の間隔が狭くなればデッドタイム内に信号が来ることになり、本来は二つきているはずの信号が一つとしか認識されないため出力効率は 50 %となるはずである。この結果を図 4.19 に記載する。

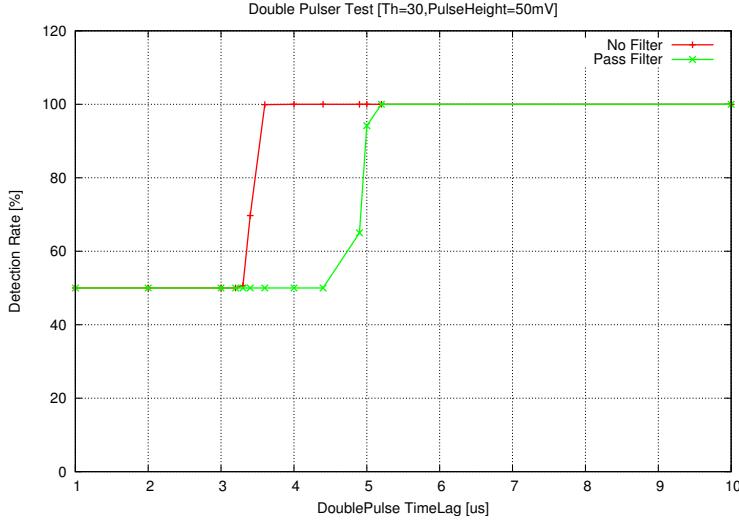


図 4.19: 移動平均 (Tap=3) フィルタ有無での連続パルス検出効率

この図からデッドタイムは移動平均フィルタ有りが 5.2 usec、フィルタ無しが 3.6 usec ほどである。この回路は PoGOLite の回路を流用しているためこんなに長くなるが、実際はサンプリング周波数を半分の 1 MHz に落としても 2.4 usec を達成している。以下の考察では 3.6 usec を元に説明する。移動平均フィルタ有りは図 4.13 の波形から分かるように立上りに時間がかかる。立上りに時間がかかることで差分データがスレッショルドを上回り続けるために図 4.8 の Trig が立ち続けることになり、結果として Trigger Module の reset ステートが長くなってしまう。これによりデッドタイムがフィルタ無しに比べて長くなる。フィルタ有りと無しではデッドタイムが 2 usec ほど違うが、これはフィルタ無しの波形 (図 4.10) とフィルタ有りの波形 (図 4.13) からスレッショルドを上回るデータ点の数から説明できる。

このデッドタイムを用いて図 4.17 のスレッショルドが 20 - 80 における出力効率が妥当かどうかを調べる。フィルタ有はスレッショルドが 20 - 80 で最高 95.2 %、フィルタ無しは 96.7 %の出力効率を達成し、また連続パルスはフィルタ有りが 5.2 usec、フィルタ無しが 3.6 usec を達成している。ここで待ち時間分布がポアソン分布に従っているとする。ポアソン式は以下の式 4.5 で表すことが出来る。

$$f(t) = N \lambda \exp(-\lambda t) \quad (4.5)$$

N はパルスのトリガ総数で λ はパルスの時間平均である。ある測定では N=134634、 $\lambda=10632$ c/sec であった。ここでフィルタ無しの連続パルス限界である 3.6 usec を用いる。式 4.5 から、あるパルスが出力され、次のパルスが 0 - 3.6 usec までに出力される個数は以下の式 4.6 となる。

$$\int_0^{3.6 \times 10^{-6}} f(t) dt = 5055 \quad (4.6)$$

これを VETO 信号出力効率に換算すると 3.75 %となり、フィルタ無しの出力効率 96.7 %と合わせておよそ 100 %であることから妥当であると思われる。同様に移動平均フィルタ有りでも求めると 0 - 5.2 usec までに次のパルスが来る個数は 7241 個で出力効率に換算すると 5.38 %であり、これも妥当である。

図 4.17 からディジタルフィルタを用いることでノイズを少なくし、低いスレッショルドを達成できる。ここで用いたディジタルフィルタは簡易フィルタであるため、様々な周波数特性を取ることが出来る HitPattern 信号はより低いスレッショルドを達成できると期待できる。また図 4.19 より多く検出するためにはデッドタイムが短ければよいので遅延器数の少ない FastVETO 信号は多くの信号を検出できると考えられる。

4.4 高度な FIR フィルタの適用

HXI/SGD は 4.1 で述べたように、FastVETO が 5 usec 以内、HitPattern が 30 usec 以内に出力しなくてはならない。現在 BGO シンチレーション光を APD 素子で受け取りプリアンプで出力される電圧信号は時定数が立上りに 1 usec、立下りに 5 - 50 usec ほどになるように設計されている。この信号を処理するうえでのサンプリング周波数は 1 MHz あれば十分な分解能、閾値を達成できる。ため、ADC のサンプリング周波数は後述する ADC のスペックも考慮して 0.94 MHz に決定した。

先ほどまではディジタルフィルタとして移動平均フィルタと差分フィルタを用いたが、これらフィルタは遅延器の数にのみ周波数特性が変化するため、複雑な周波数特性を作ることが出来ない。そのため上空での APD 経年変化などに応じて周波数特性を細かく変化させることが難しい。そこで HXI/SGD ではより複雑な周波数特性を持つことができる FIR フィルタを用いることにした。図 4.20 が FIR フィルタのブロック図である。

1 MHz サンプリングの場合 FastVETO 信号と HitPattern 信号の要求された時間以内に出力するため、FIR フィルタはそれぞれ 8 個、16 個の遅延器数となるように設計する。またフィルタ係数も現在のところ FPGA の容量を加味し、-1, 0, +1 か -256 ~ +255 の整数と予定されている。しかしこれらは一般的の FIR フィルタを用いるうえではひじょうに数が少ない（音の世界では遅延器数も 100 を越えるし、フィルタ係数も 2^{20} 以上の精度を持つ）。また、実装する周波数特性もよりノイズを落とすためにはただ LPF を用いればいいというわけではない。どの周波数特性を通せばよいのかを調べる必要がある。

そこでここでは遅延器数、フィルタ係数精度を変えた場合による FIR フィルタの影響と、アナログ波形整形アンプを目的とした任意の周波数特性の実装を行い、上の遅延器数と係数精度が十分であるか調べた。

4.4.1 FIR フィルタ

FIR フィルタの詳細な説明は付録 A.4 の FIR Filter に記している。

FIR フィルタは移動平均フィルタや差分フィルタでは難しい、任意の周波数特性を取ることが出来るフィルタである。FIR フィルタのブロック図を図 4.20 に載せる。 h_m は FIR フィルタにおけるフィルタ係数である。

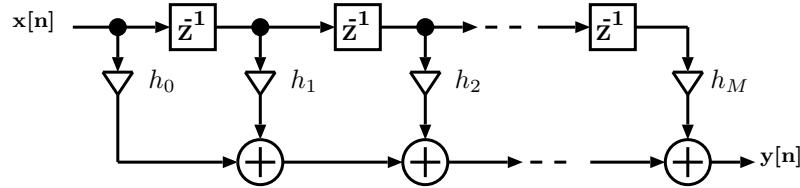


図 4.20: FIR フィルタのブロック図

このように移動平均フィルタと似ているが各遅延データにフィルタ係数という重みを加えて足し合させている。この係数を変更することで計算可能なものであればいかなる周波数帯域も実現できる。この差分方程式は式 4.7 となり、周波数特性は式 4.8 となる。

$$y[n] = \sum_{m=0}^M h_m x[n-m] \quad (4.7)$$

$$|H(\omega)| = \sqrt{\left(\sum_{m=0}^M h_m \cos(m\omega T) \right)^2 + \left(\sum_{m=0}^M h_m \sin(m\omega T) \right)^2} \quad (4.8)$$

ここで $T = 2\pi/\omega_s$ である。この式から分かるように、遅延器数とそれによるフィルタ係数を決めることで周波数特性が決定する。

4.4.2 BPF を FIR フィルタで実現

FIR フィルタはそのフィルタ係数を変更することにより、計算可能で、かつ遅延器数が充分であればいかなる周波数特性も再現できる。ここではフィルタの代表的な例である BPF を再現する。

理想的な BPF は以下のような周波数特性を持つ。なお、 $G(\omega)$ は最初に与えるフィルタの周波数特性と定義している。そのため伝達関数 $H(\omega)$ と同じ意味だが、 $H(\omega)$ はフィルタがあって、フィルタ係数、遅延器から求めた結果としての周波数特性として区別している。

$$G(\omega) = \begin{cases} 0 & |\omega| < \omega_{cL} \\ 1 & \omega_{cL} < |\omega| < \omega_{cH} \\ 0 & \omega_{cH} < |\omega| < \omega_s/2 \end{cases} \quad (4.9)$$

このときフィルタ係数は式 4.10 を満たす。 g_n は無限個の遅延器が存在する場合の理想のフィルタ係数とする。このときの周波数特性を図 4.21 に示す。理想フィルタは無限個の遅延器数が必要だが、ここでは幅 L 個で遅延器数を区切っている。そのため遅延器数は $2L+1$ である。遅延器数で区切った、実際に用いるフィルタ係数を h_n とする。

$$\begin{aligned}
g_n &= \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(\omega) \exp(in\omega T) d\omega \\
&= \frac{1}{n\pi} \left\{ \sin\left(\frac{2n\pi\omega_c H}{\omega_s}\right) - \sin\left(\frac{2n\pi\omega_c L}{\omega_s}\right) \right\} \\
n &= -\infty, \dots, -1, 0, 1, \dots, +\infty
\end{aligned} \tag{4.10}$$

$$h_n = g_n \quad (|n| < L) \tag{4.11}$$

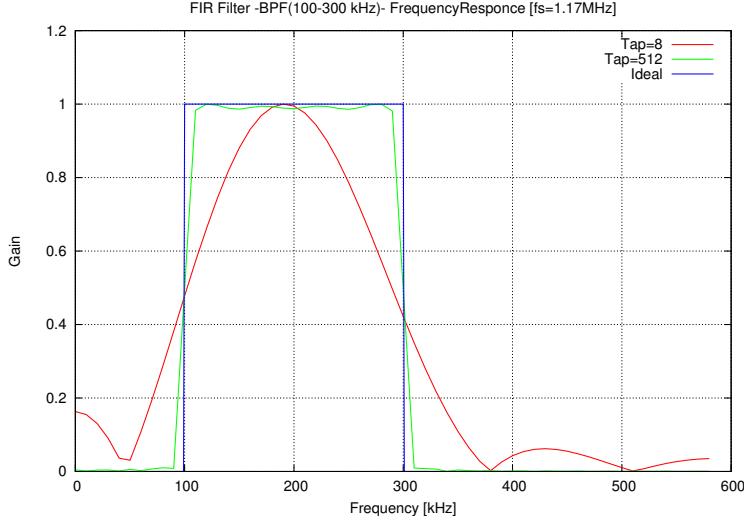


図 4.21: FIR BPF(透過帯域 100-300 kHz) 遅延器の数を変えたときの周波数特性 ($f_s = 1.17\text{MHz}$)

4.4.3 FIR フィルタ遅延器数・係数精度

さきほどの 4.4 では HXI/SGD での時間応答による理由から遅延器数を決定したが、遅延器数がこの数で十分かどうかを調べる必要がある。なぜなら FIR フィルタは遅延器数とフィルタ係数の精度によって周波数特性が決定するからである。遅延器数が少ないとそれだけフィルタ係数も少なくなり、自由度が無くなる。フィルタ係数もソフトウェアの double 型のように何桁も使えるのならば複雑な周波数特性が可能だが、 $\pm 1, 0$ しか使えないとかなり限定される。そこでどの程度の遅延器数、フィルタ係数精度だと期待する周波数特性を実現できるのかということを調べた。今回調べた FIR フィルタは式 4.12 で表される 100-300 kHz 透過の BPF である。BPF のフィルタ係数は式 4.10 によって計算が可能である。

$$G(\omega) = \begin{cases} 0 & 0 < 100 \\ 1 & 100 < |\omega| < 300 \\ 0 & 300 < |\omega| < f_s/2 \end{cases} \tag{4.12}$$

unit : kHz

遅延器数の違いによる周波数特性の変化

図 4.22 は FIR Band Pass Filter(BPF)[100-300kHz] の遅延器数の違いによる周波数特性プロットである。サンプリング周波数は 1.17 MHz でフィルタの係数は 2^{22} の値まで取ることが出来る。この図 4.22 を見る限りでは遅延器数 4 だとあまり期待される周波数特性を再現してくれないが遅延器数 8 以降は遅延器数数が変わっても大きく変化しないことが分かる。

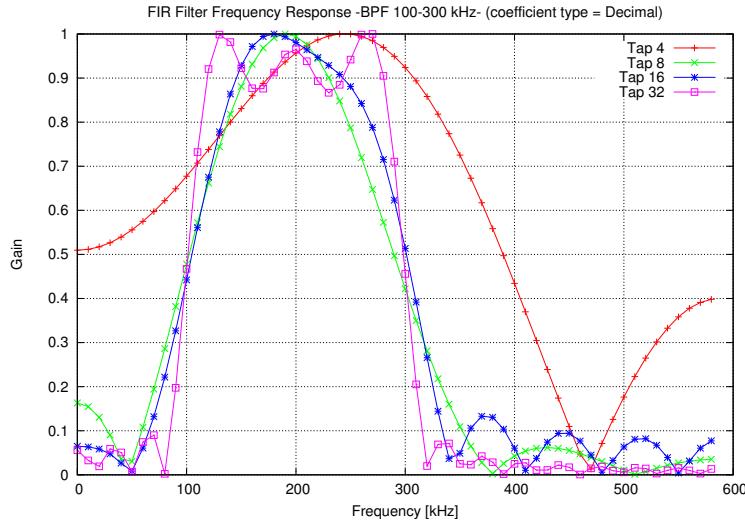


図 4.22: FIR BPF(通過帯域 100-300kHz) の遅延器数を変えた場合の周波数特性

このことをより詳しく検証するために最小二乗法を用いた。理想的な周波数特性は式 4.12 であり、各周波数での理想と計算値での値の差(残差)を二乗し足し合わせた数値を遅延器数毎にプロットする。理想に近ければよいので遅延器数数が増えれば残差の二乗和は低くなっていくはずである。これをプロットしたグラフが図 4.23 である。

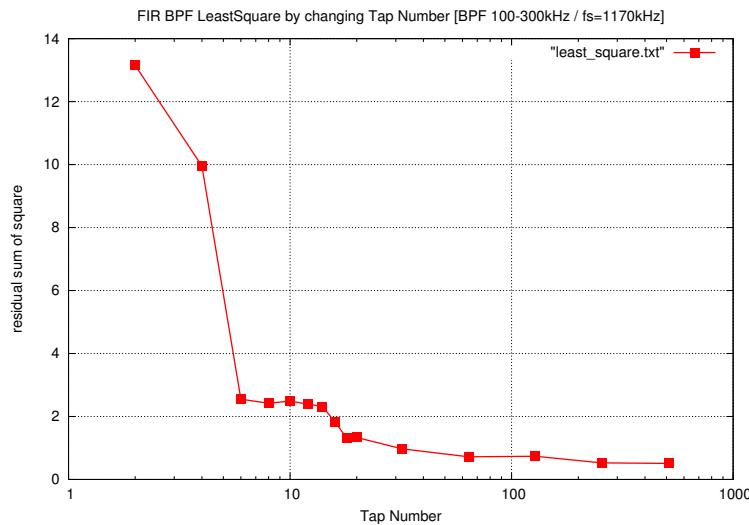


図 4.23: FIR BPF 遅延器数の違いによる残差二乗和

横軸が遅延器数で縦軸が残差の二乗和である。遅延器数が増えるにつれて残差二乗和が下がっていることがよく分かる。遅延器数が 6 から 14 まではほぼ横ばいで 16 から再び下がり始め、20 からは緩やかに減少している。図 4.22 を見る限り遅延器数 8 でも 100 - 300 kHz の帯域を通過させる特徴が見られるため、遅延器数を 6 まで下げても同じような周波数特性が見られると思われるが、HXI/SGD では余裕を少し持たせて 8、および更に複雑な周波数特性のために 16 とした。

係数精度による周波数特性の変化

今度は遅延器数を 16 に固定し、係数の精度を変化させたときの周波数特性の違いを調べた。係数の精度とはその数字が 2 の何乗で表すことができるかで定義している。 $\pm 1, 0$ しかとれないのであれば $-1, 0, +1$ の 3 つで表せるので $2^2 = 4$ 以内より精度は 2 であり、 $-8 \sim +7$ ならば 16 個の数字で表せるので $2^4 = 16$ から精度 4 となる。図 4.24 は係数の精度を変えたときの周波数特性である。

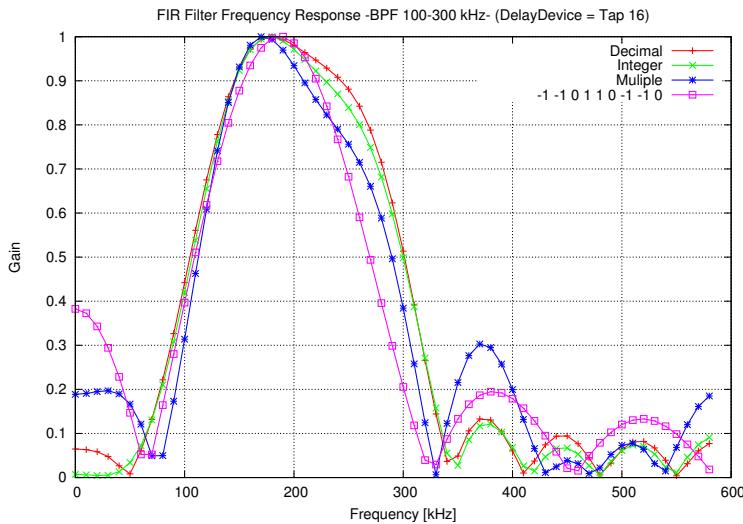


図 4.24: FIR BPF(通過帯域 100-300kHz) の係数精度を変えた場合の周波数特性

各係数は赤が小数で $2^{20} = \pm 1048576$ までの精度を持ち、緑が整数で $2^8 = \pm 128$ 、青が倍数で $\pm 2^n$ 、そしてマゼンタが $\pm 1, 0$ のみとなっている。この図を見る限りでは各係数でも大きな変化は見られないようと思われる。一見すると精度が落ちると、指定した 100 - 300 kHz 以外の帯域におけるゲインが 0 に落ちにくいように思えるが、この効果は遅延器数の変化でも見られたので係数だけの問題とは思えない。図 4.23 同様に係数の精度を横軸に残差の二乗和をとる。それが図 4.25 である。

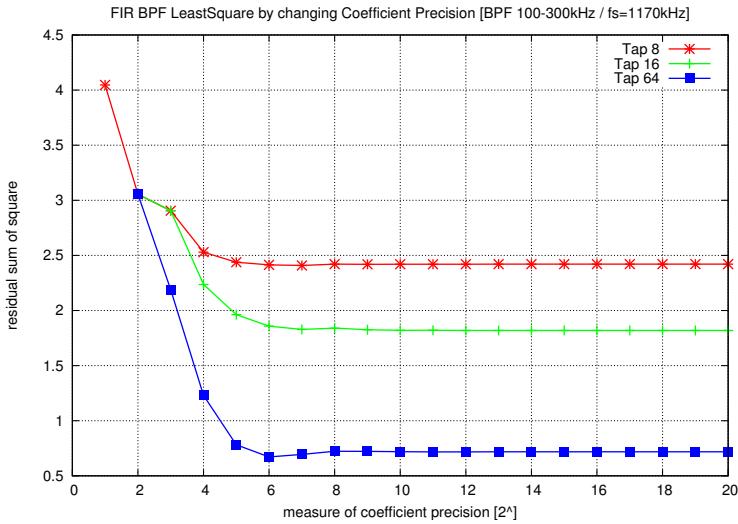


図 4.25: FIR BPF 係数精度の違いによる残差二乗和 (横軸: フィルタ係数精度 2^n の n、遅延器数別にプロットしている)

図の横軸はフィルタ係数の精度 2^n の n を横軸に取っている。また、遅延器数での違いを見るため、遅延器数が 8, 16, 64 の場合もプロットしている。この図を見るといずれの遅延器数でも係数精度が $2^6 = \pm 64$ からはほぼ横ばいであることが分かる。このことから係数精度はそう高い桁数を用いなくてもよいと思われる。 ± 1 に関してだが、図 4.24 を見る限りではユーザの期待する周波数特性を再現していると思われるため、FPGA の容量消費を抑える上では最も桁数を抑えられる係数ではある一方、 ± 1 だと自由度が少ないデメリットがある。選べる係数が少ないとめどれかの係数を変えるだけで周波数特性が大きく変動してしまい、数 kHz だけ周波数特性をずらすといったことが出来ない場合が多く、微調整がひじょうに難しい。

以上を踏まえて HXI/SGD では係数精度が $\pm 1,0$ か $-256 \sim +255$ とした。図 4.25 からでは $\pm 1,0$ と $-256 \sim +255$ では精度が大きく異なる可能性がある。そのためこの二つの違いを意識した検証を行う必要がある。

4.4.4 波形整形アンプを模擬した FIR フィルタの設計

BGO 結晶を APD 素子で検出した信号の周波数成分

フィルタをかけてスレッショルドを下げるためにはまずノイズの周波数成分と信号の周波数成分を調べ、どの周波数帯域を除去または通過するかを決める必要がある。そのために線源を照射しないノイズ信号と ^{241}Am 信号の波形を取得し、その波形を離散フーリエ変換 (Discrete Fourier Transform:DFT) にかけて周波数成分を取得した。DFT の計算式は式 4.13 で周波数成分は式 4.14 で導出できる。 $x[n]$ は離散時間系の時間信号であり、 $G[k]$ はそのフーリエ変換で周波数成分を表す。N は波形のデータ点数である。BGO+APD のノイズ成分と ^{241}Am 信号成分の周波数成分を図 4.26 に載せる。

$$G[k] = \sum_{n=0}^{N-1} x[n] \exp\left(\frac{-i2nk}{N}\right) \quad (4.13)$$

$k = 0, 1, \dots, N - 1$

$$|G[k]| = \sqrt{\operatorname{Re}\{G[k]\}^2 + \operatorname{Im}\{G[k]\}^2} \quad (4.14)$$

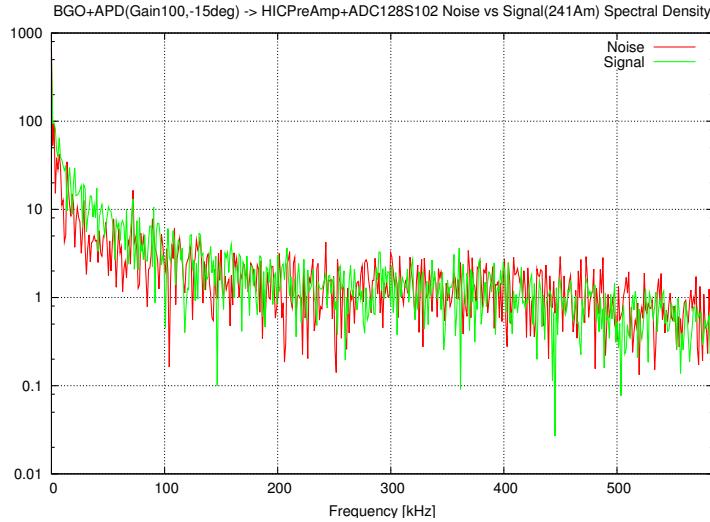


図 4.26: BGO+APD(Gain100,-15deg) HICPreAmp,ADC128S102 による ^{241}Am 信号 (緑) とノイズ (赤) の周波数成分

波形の取得においてプリアンプは衛星で用いる実機と同等の性能であるもの (HIC プリアンプ) を用いた。この図から除去すべき周波数を求める。信号波形にはノイズ成分も含まれるため、周波数成分で共通の部分もあるが、120 kHz 以下では信号成分がノイズ成分よりも強く表れている。そこでフィルタの透過すべき周波数を少し余裕を持たせて 20 - 140 kHz 辺りになるように作成することにする。

アナログフィルタ (Shaper)

検出器を用いる実験で現在良く使われるのがアナログフィルタの波形整形増幅器 (Pulse Shape Amplifier:Shaper) である。波形を整形する上で高周波成分のノイズをカットし、信号の PreAmp 出力に合わせた shaping time(内部 CR-RC 回路の時定数) を選択することで信号の周波数成分を強く引き出すことができる。さらに良い Shaper だと通過後の波形がガウシアンの形になるため、ADC しやすいなどのメリットがある。ClearPulse(CP) 4417 型と Ortec 571 型の Shaper は地上試験で実績があるため、まずはこの Shaper に近い周波数特性を持つディジタルフィルタを作ることにした。フィルタを作る上でまず Shaper の周波数成分を知ろうしたが、カタログには周波数特性が記載しておらず、また内部回路は $CR - (RC)^n$ 回路の $n=4$ であると思われるが周波数特性計算がよく分からないので今回はテストパルス信号を Shaper に通す前と通した後の周波数成分を

式 4.13 と式 4.14 の DFT で計算し、その比から周波数成分を求めた。これによる CP Shaper の shaping time 別周波数特性が図 4.27 であり、Ortec Shaper の周波数特性が図 4.28 である。

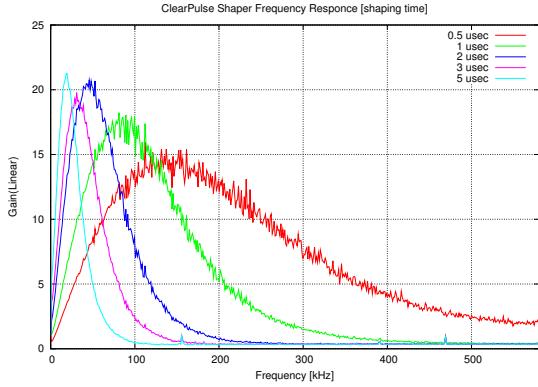


図 4.27: ClearPulse Shaper 4417 ShapingTime 別 周波数特性

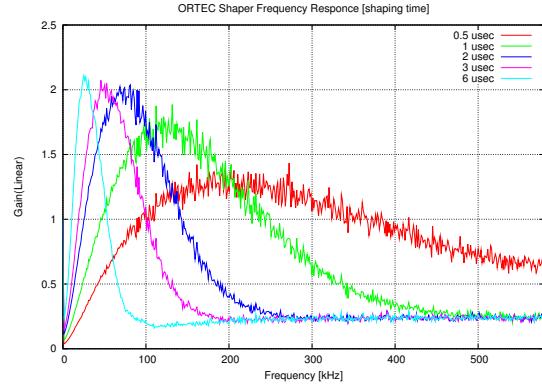


図 4.28: ORTEC Shaper 571 ShapingTime 別 周波数特性

各社で同じ shaping time であっても少し周波数特性がずれているのが分かる。これを比較した図が図 4.29 である。

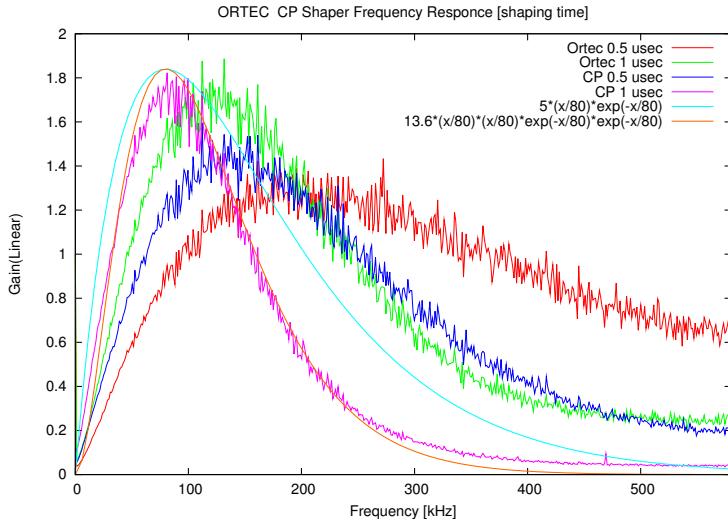


図 4.29: CP と ORTEC Shaper 周波数特性 比較

図 4.29 には仮りに ClearPulse の Shaper $\tau=1$ における周波数特性を関数で表した場合の線も引いている。これより Shaper の周波数特性を式で表すのであれば $|G(\omega)| = \left(-\frac{\omega}{\omega_{Peak}}\right) \left\{ \exp\left(-\frac{\omega}{\omega_{Peak}}\right) \right\}$ か $|G(\omega)| = \left(-\frac{\omega}{\omega_{Peak}}\right)^2 \left\{ \exp\left(-\frac{\omega}{\omega_{Peak}}\right) \right\}^2$ が近いのではないかと思われる。よってこの周波数特性を持つ係数を計算するようとする。例として以下では周波数特性を式 4.15 として計算する。

$$|G(\omega)| = \left(-\frac{\omega}{\omega_{Peak}}\right) \left\{ \exp\left(-\frac{\omega}{\omega_{Peak}}\right) \right\} \quad (4.15)$$

アナログフィルタの FIR フィルタ化

従来の FIR フィルタ係数の計算は全て透過帯域、遮断帯域がステップ型であった。そのため計算式が参考書に載るほど簡単で今までその計算結果を用いて係数を導出していたが、今日のアナログフィルタの再現では新しく自分で求めることになるため、定義式から始めることにする。

$$\begin{aligned}
 g_n &= \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(\omega) \exp(in\omega T) d\omega \\
 &= \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(\omega) \cos(n\omega T) d\omega + i \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(\omega) \sin(n\omega T) d\omega
 \end{aligned} \quad (4.16)$$

$n = -\infty, \dots, -1, 0, 1, \dots, +\infty$

今まで付録の式 A.40 のように参考書に習ってここで $G(\omega)$ を偶関数と [仮定] し、式 4.16 の \sin の項を消去したが、この計算結果では係数も偶対称になる。実は係数が偶対称の場合、波形が図 4.30 緑線のようにピーク前に大きな窪みができてしまうことが分かった。これは図 4.31 緑線のように係数が偶対称の場合、真ん中の遅延器で負正負と大きく変動するためピークに負が掛けられてしまい、図 4.30 のような窪みが出来てしまうのである。

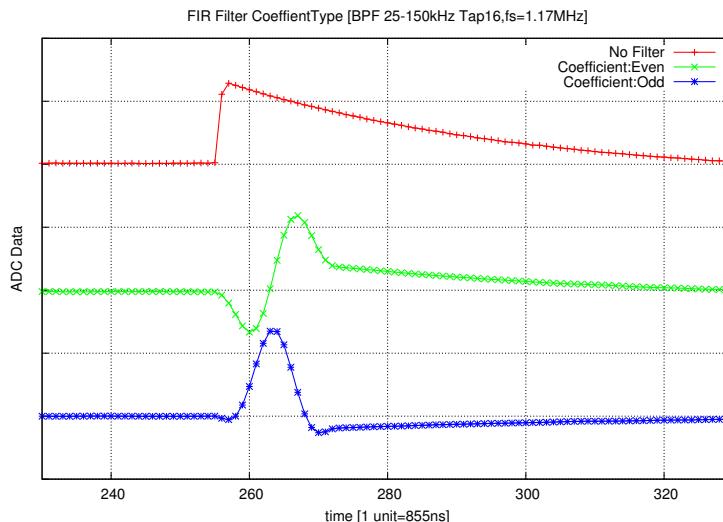


図 4.30: 係数の対称性の違いによる波形の変化(赤:元となるプリアンプ信号、緑:偶対称係数フィルタに通したときの波形、青:奇対称係数フィルタに通したときの波形)

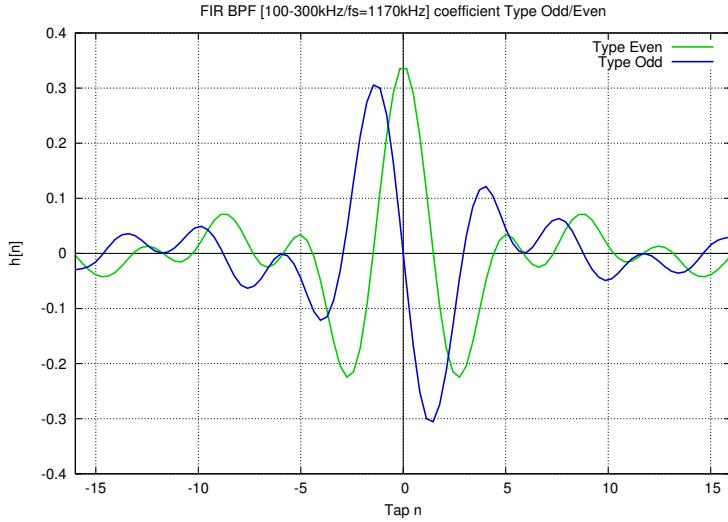


図 4.31: 係数が偶対称の場合(緑)と奇対称の場合(青)

これを回避するためには図 4.30 青線のように係数が奇対称であればよく、逆算して式 4.16 の \cos の項が消えて \sin の項が残ればよい。同じ BPF (通過域 100-300 kHz) を奇対称、偶対称の係数で表した場合の周波数特性を図 4.32 に表す。この図を見る限り、奇対称・偶対称で大きな違いがあるようには見えない。

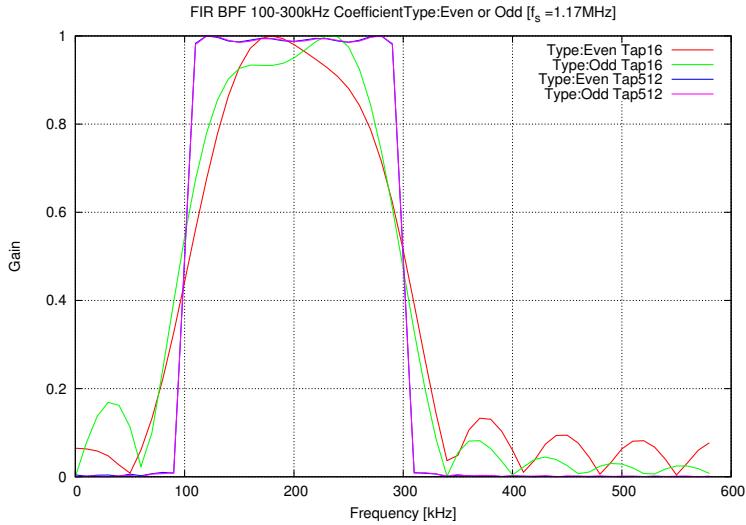


図 4.32: 係数が偶対称と奇対称の場合による周波数特性(BPF 100-300kHz)

周波数特性において奇対称・偶対称に大きな違いはないと思われるため、 $G(\omega)$ を奇関数と[仮定]して計算した。これにより、式 4.16 は虚数を省略して以下の式 4.17 のようになる。

$$\begin{aligned}
g_n &= \frac{2}{\omega_s} \int_0^{\omega_s/2} G(\omega) \sin(n\omega T) d\omega \\
&= \frac{2}{\omega_s} \int_0^{\omega_s/2} \left(\frac{\omega}{\omega_{peak}} \right) \exp \left(-\frac{\omega}{\omega_{peak}} \right) \sin(n\omega T) d\omega
\end{aligned} \tag{4.17}$$

$n = -\infty, \dots, -1, 0, 1, \dots, +\infty$

この式を計算することで奇対称の係数を算出することができる。 g_n は無限個のときのフィルタ係数より式 4.17 から任意の幅 (遅延器数) を決めて計算を行い、係数 h_n を導出する。その係数から再び周波数特性にプロットし直したものが図 4.33 である。サンプリング周波数は 1.17 MHz であり、遅延器数は 20 個で ω_{peak} 10 kHz から 300 kHz までプロットしている。また正しく周波数特性が再現できているかどうかを見るために元となった式 4.15 の Peak に 80 kHz を代入したものもプロットしている。図からは 300 kHz をピークとするものは形が乱れているが、それ以外は概ね目的とする形を保っていることがわかる。このようにアナログフィルタと同様の周波数特性の形でありながらピーク周波数を自由に変えることができるフィルタを作ることができた。

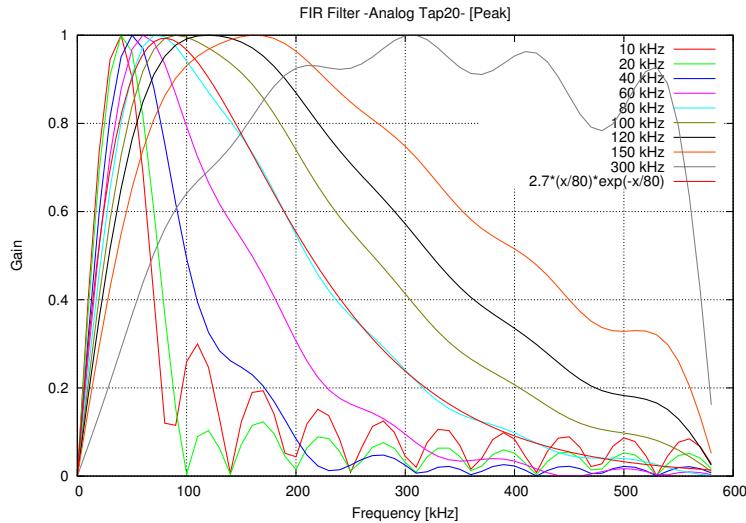


図 4.33: FIR フィルタで Shaper(式 4.15) を再現し、ピーク位置を変化させた場合の周波数特性

今回は周波数特性において、ピークが 50 - 60 kHz になるフィルタを作成する。これは shaping time で 1 - 2 usec 相当である。しかし、元々アナログフィルタは式 4.15 を元に作成したが、図 4.26 を良く見ると信号成分はノイズ成分同様に低い周波数成分を持つ。この周波数は信号波形においてピークからなだらかに下がる成分であると思われるため、このような低い周波数成分が信号成分の重要なファクタである可能性もある。そこで低周波数成分に若干のオフセットを加えれば良いが、フィルタ係数が少し複雑になるのですべての周波数帯にてオフセットを加え、このオフセットがノイズカットに効果があるかどうかを調べる。そこで係数に少し変化を加えて周波数特性にオフセットを上乗せする。そのため図 4.34 のように係数 $h_0 = 1$ というデルタ関数を加える。

オフセットは図 4.34 にあるように、最も高い係数値との比率を用いて $n=0$ のフィルタ係数値を決定した。図 4.34 は 150 % の場合を表している。

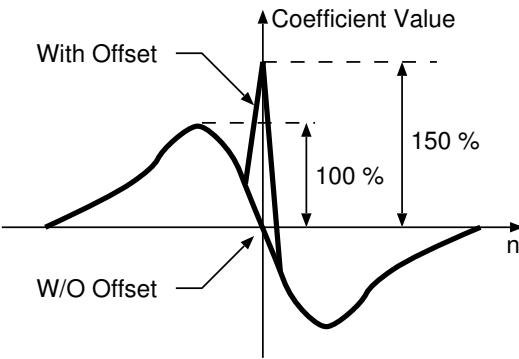


図 4.34: 周波数特性にオフセットを与えるための FIR フィルタの係数 g_n

実際に係数 h_n として図にしたものと、その係数から周波数特性にしたものが図 4.35 と図 4.36 である。図 4.35 は図 4.34 の状態から $n > 0$ となるようシフトしている。図 4.36 は 1 に正規化している。周波数特性を見て分かるようにピークの位置は変わっていないが、全体的に周波数特性が一定値だけ上乗せされているのが分かる。

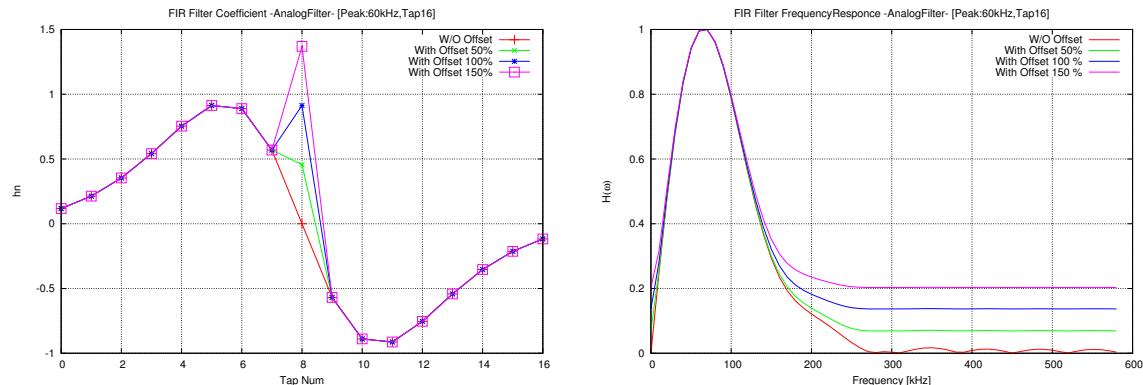


図 4.35: 周波数特性にオフセットをつけるための係数 h_n (色によってオフセットの度合が異なる)

図 4.36: 図 4.35 の各係数における周波数特性 (色は図 4.35 と対応している)

これらアナログ波形整形アンプを元にしたフィルタにオフセットを加えたものや典型的な BPFなどのフィルタを用いてヒストグラムを取得する。

4.4.5 ソフトウェアによる FIR の効果検証

FIR フィルタを用いてより低い閾値を達成するためには適切なフィルタを見つける必要がある。そのために最終目的である FPGA 上でいきなり FIR フィルタを用いたヒストグラム作成のために装置開発や論理合成に時間を費すよりもソフトウェアで開発を行ったほうがトータルで時間の節約になる。そこでまずはソフトウェアでの FIR フィルタを用いた検証を行った。

ソフトウェアで FIR を検証するためには波形データを用いて処理したほうがやりやすい。波形を保存できればその波形に FIR フィルタを通して再びヒストグラムにするときに同じ波形同士を比

較することもできるなど、ハードウェアでは難しいことも容易となる。そこでまずはFADC Boardで波形を取得した。このときの波形フォーマットは図 4.37 である。

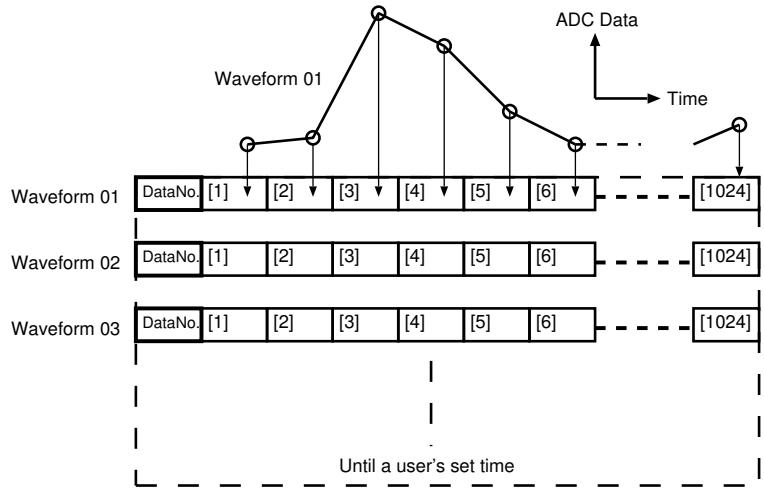


図 4.37: FPGA での波形取得時のデータフォーマット

サンプリング周波数は実機に最も近い 1.17 MHz である。1 波形で 1024 点取得し、トリガ前が 250 usec、トリガ後が 600 usec ほどである。これは 1000 usec. ほど取得すれば一つの APD 信号波形が完全に入り、また前後のノイズを含まるからである。このフォーマットを元に取得した波形を図 4.38 に示す。これは ^{241}Am を BGO 結晶 (1cm 角) に当て、APD 素子で検出したものである。APD は-15 度に冷やし、Gain が 100 になるようにしている。実験ではこのような波形を 10000 点ほど取得する。

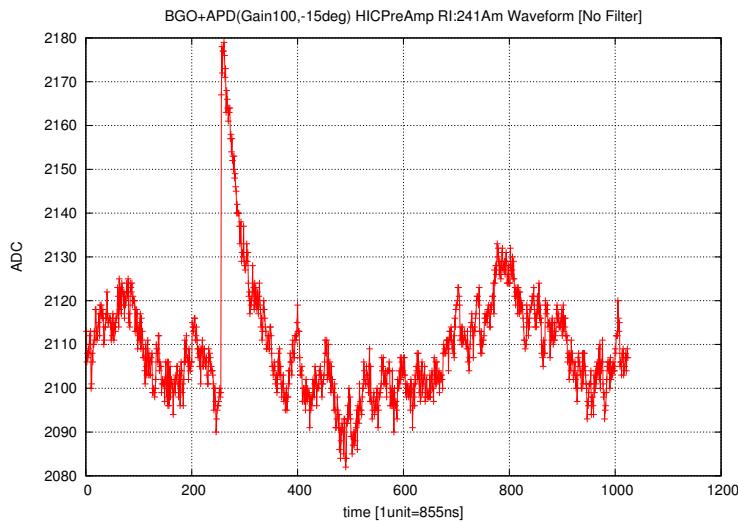


図 4.38: ^{241}Am を BGO+APD(Gain100,-15deg) に照射、HICPreAmp 出力信号を FADC Board で取得した波形 ($f_s = 1.17\text{MHz}$)

この波形を通す FIR フィルタは以下の四つである。これらのフィルタは $\pm 1,0$ 以外は遅延器数

が 16 であり、各フィルタ係数を周波数特性に表したもののが図 4.39 である。

- 1 1 1 1 0 -1 -1 0 0 (FIR2bit)
 $\pm 1,0$ に絞ったもの。波形整形アンプのピークが 80 kHz のものに最も近いものを選んでいる。
- FIR BPF 通過域 25-150kHz オフセット 150 %
- FIR Analog1 Peak:60kHz オフセット 150 %
 $G(\omega) = \left(-\frac{\omega}{\omega_{Peak}}\right) \left\{ \exp \left(-\frac{\omega}{\omega_{Peak}}\right) \right\}$ を元にして作成したフィルタ。
- FIR Analog2 Peak:60kHz オフセット 150 %
 $G(\omega) = \left(-\frac{\omega}{\omega_{Peak}}\right)^2 \left\{ \exp \left(-\frac{\omega}{\omega_{Peak}}\right) \right\}^2$ を元にして作成したフィルタ。

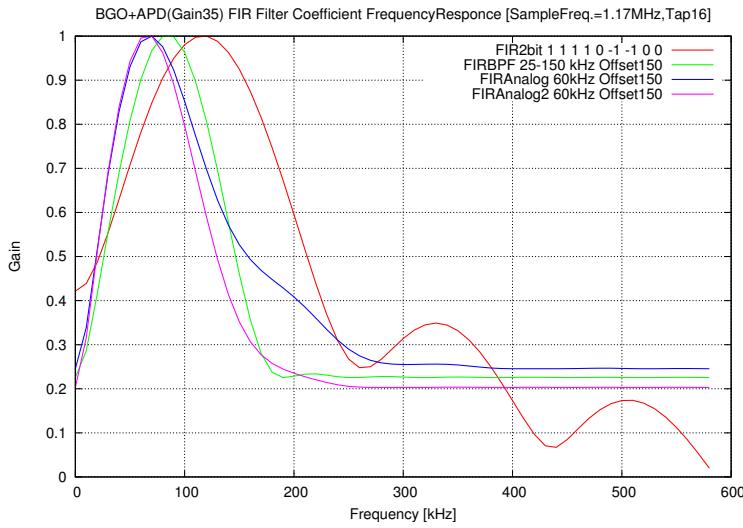


図 4.39: ソフトウェアでの FIR フィルタの検証に用いる各フィルタ係数における周波数特性

$\pm 1,0$ のフィルタは他のフィルタと違って遅延器数が 8 個としている。また、これは $\pm 1,0$ を用いた全ての組合せを全て考えるわけにはいかないので $G(\omega) = \left(-\frac{\omega}{\omega_{Peak}}\right) \left\{ \exp \left(-\frac{\omega}{\omega_{Peak}}\right) \right\}$ のピーク位置が 80 kHz となるものに最も近いものを用いている。これは波形整形アンプの shaping time=1.5 usec に大体対応している。FIR BPF は同じ理由で 80 kHz が中心になる 0 - 175 kHz, 25 - 150 kHz, 50 - 125 kHz の通過域となるフィルタを作成し、その内最も低いスレッショルドだった 25 - 150 kHz を選んでいる。その 25 - 150 kHz の BPF からオフセットを加えていく、その中で最も効率が良いものが 150 % だったのでここではそれを選んだ。FIR Analog1 または FIR Analog2 も同様に、まずはピーク位置を変えて最も低いスレッショルドだったものを選び、更にそこからオフセットを加えていくってその中で最も低いスレッショルドだったものを選んでいる。

これらのフィルタ係数を用いた FIR フィルタに波形 4.38 を通したものが図 4.40 から図 4.43 である。

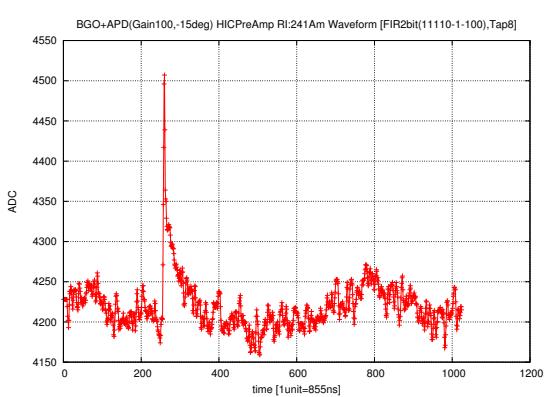


図 4.40: 図 4.38 に FIR2bit フィルタを通した波形

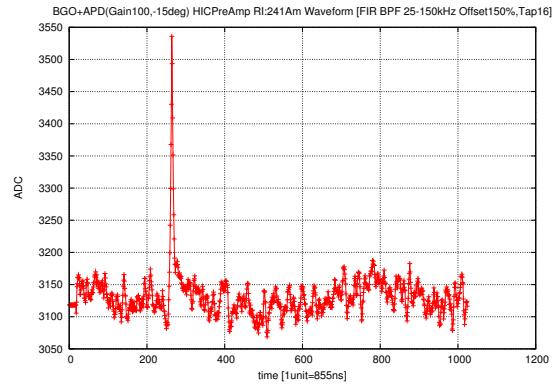


図 4.41: 図 4.38 に FIR BPF 25-150kHz を通した波形

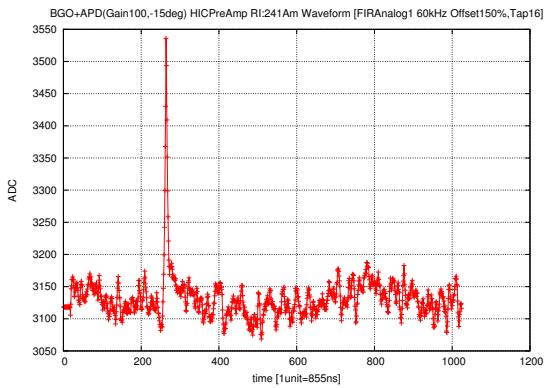


図 4.42: 図 4.38 に FIR Analog1 60kHz を通した波形

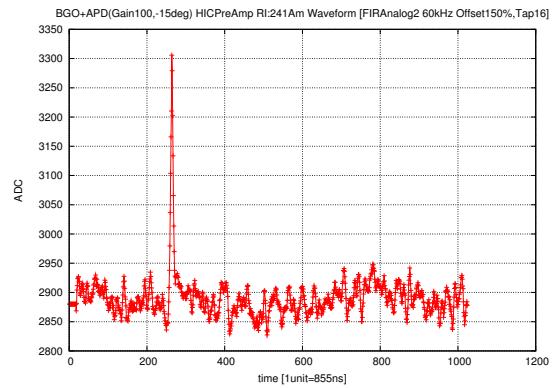


図 4.43: 図 4.38 に FIR Analog2 60kHz を通した波形

図 4.40 はフィルタ係数が全て整数なので重みが増し、他のフィルタに比べてゲインが高いのが分かる。また他のフィルタは元々の波形と比較してアンダーシュートがあまり長引かないことが見てとれる。この波形を用いてヒストグラムを描いたものが図 4.44 である。

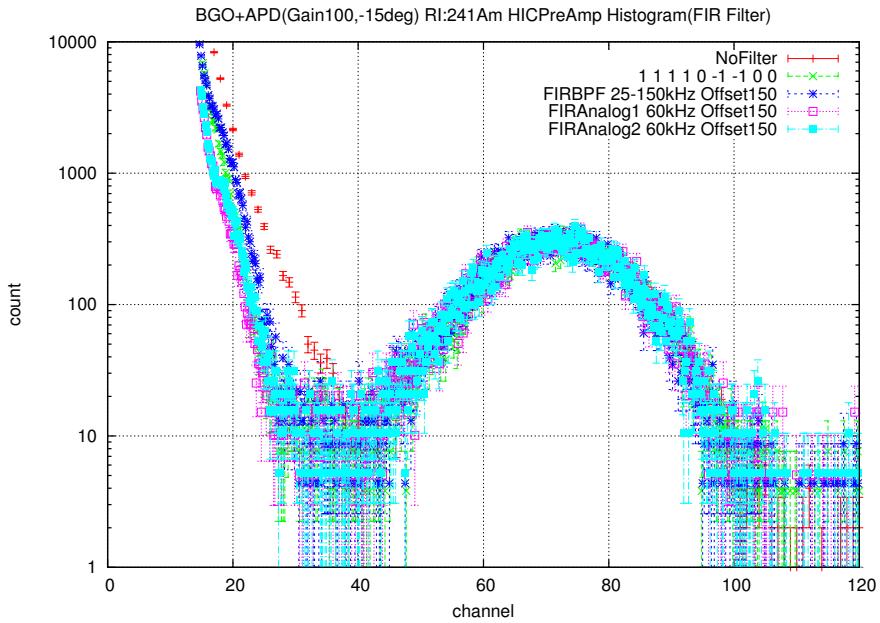


図 4.44: ソフトウェアによる FIR フィルタを用いたヒストグラム比較 (BGO+APD ^{241}Am -15deg)

^{241}Am の 59.5 keV ピークが 70 ch. ほどにあり、その分解能を見た目にはほぼ崩さず、低スレッショルドを達成できているのがよくわかる。この図からは FIR Analog1 か FIR Analog2 フィルタが最も効果があると思われる。

4.5 ADC 素子の選定

HXI/SGD におけるアクティブシールドの APD 信号をディジタル変換するための素子 (ADC 素子) は以下のような要求がかせられている。

1. 1 MSPS

BGO+Preamp 信号を処理するには 1MHz で十分である。

2. 12 bit 分解能

3. 低電力

ADC 素子 1 ch. あたり 50 mW。

4. 小面積

80 × 80 mm の基板上に APD 信号 14ch. 分の処理回路を設置する。

衛星という資源が限られた環境であるため、ADC 素子と FPGA を含む APMU 全体に与えられた電力は 1.4 W であり、APD 信号 1 ch. あたり 0.05 W 以内の電力で動作できるもの、そして APD 信号 14 ch. で 8 × 8cm の基板に ADC 素子、APMU-FPGA、各種アナログ回路を搭載しなくてはならないので小面積であることが求められている。候補に上がった ADC 素子が以下の表 4.2 である。

メーカ	型式名	入力数	MSPS	出力形式	消費電流 (@3 V)	放射線耐性
NS	ADC128S102QML	8	0.5-1	serial	0.9 mA	100 krad(Si)
TI	THS1206-EP	4	6	para	62 mA	無
ST	TSA1204	2	0.5-20	para	31 mA(10 MSPS)	無
ST	RHF1201	1	0.5-50	para	15 mA(0.5 MSPS)	300 krad(Si)

NS = National Semiconductor

TI = Texas Instruments

ST = ST Microelectronics

表 4.2: HXI/SGD ADC 素子の候補リスト

THS1206EP はアナログ入力ピンの数が 4 本と多く、サンプリング周波数も最大 1.5 MSPS (6 MSPS を 4 本で分ける) であるが、ディジタル出力ピンがパラメタ変更のためのディジタル入力ピンと混同である inout ピンのため設計が複雑になり、回路面積が限られていることを考えると実装しにくい。TSA1204 は入力ピンが 2 本でサンプリング周波数も要求に対して十分な性能を持つが、ST Microelectronics 社でまだ放射線耐性の試験を行っておらず、また APD 信号 1ch. に対する消費電力も 15 mA と少し高い。RFH1201 は放射線耐性の試験を行って衛星向けの製品ではあるが素子面積が $15.75 \times 9.65\text{mm}$ と少し広く、ピンの数も 1 石につき 48 本あるため信号線が多くなってしまう。それらを考え、HXI/SGD チーム内で ADC128S102QML を ADC 素子として選択した。

4.5.1 ADC128S102

National Semiconductor 社の ADC128S102QML は小さい面積ながら 0.05 - 1 Mbps が可能で、データ転送もシリアル通信のためパラレル通信に比べて配線も少なくてすむ。実験室で用いるには放射線耐性は必要ないため、ほぼ同様の性能の地上試験用 ADC128S102 をここでは用いる。以下の表 4.3 に ADC128S102QML と ADC128S102 の基本的な性能を載せる。

ADC128S102QML と ADC128S102 との違いはサンプリング周波数のレンジの広さ、放射線耐性の有無、素子の大きさ程度であり、AD 変換方式、ピン配置、クロックタイミングは全く同じである。ADC128S102 はアナログ入力ピンは 8ch. あるが、一度に AD 変換を行えるのは 1ch. だけなので 8ch. 全て使うとなるとその入力ピンの切り替えによってサンプリング周波数が実質 $1/8$ に落ちてしまう。APMU では APD 一つに対する AD 変換素子に許される電力が 50 mW 以下なので、APD 素子一つに対して ADC128S102QML を一つに割り振っても電力量は問題ない。これによって各 APD 素子で 1 MHz を達成できる。

4.5.2 ADC128S102 シリアル通信方式

ADC128S102 はディジタル入力が SCLK、CS、DIN の三つである。この三つでサンプリング周波数を決め、AD 変換を行う入力ピンを切り換え、AD 変換を行うタイミングを決定する。AD 変換された信号はディジタル出力ピンである DOUT から SCLK に従って出力される。各ピンの役割を表 4.4 に記載する。評価実験ではこれらを FADC Board で制御する。

項目	ADC128S102QML	ADC128S102
Sampling Freq.	0.05 - 1 Mbps	0.5 - 1 Mbps
分解能	12 bit (=4096ch)	12 bit
アナログ入力	8 ch.	8 ch.
電源電圧 (V_A)	$+2.7 < V_A < +5.25V$	$+2.7 < V_A < +5.25V$
電源電圧 (V_D)	$+2.7 < V_D < V_A$	$+2.7V < V_D < V_A$
リファレンス電圧 (V_{ref})	$=V_A$	$=V_A$
アナログ入力レンジ	0 to V_{ref}	0 to V_{ref}
消費電力	2.3 mW(3V) / 10.7 mW(5V)	2.3 mW(3V) / 10.7 mW(5V)
素子面積	$9.91 \times 6.35\text{mm}^2$	$5 \times 4.4\text{mm}^2$
素子高さ	1.78 mm	0.9 mm
外部クロック	0.8 MHz - 16 MHz	8 MHz - 16 MHz
デジタル入力	SCLK,CS,DIN	SCLK,CS,DIN
デジタル出力	DOUT	DOUT
AD 変換方法	逐次比較型	逐次比較型
放射線耐性	100 krad(Si)	無し

表 4.3: ADC128S102QML/ADC128S102 基本性能

SCLK	Serial Clock。サンプリングを行うためのクロックであり、16 MHz 入力で 1 MSPS。
CS	Chip Select。AD 変換の on/off。'L' で on。
DIN	Digital data Input。アナログ入力ピンを切り替えるための入力。b'000' で ch.0、b'111' で ch. 7。
DOUT	Digital data Output AD 変換データを出力。CS が'0' の間、SCLK に合わせて、MSB から順に 12 bit 出力。

表 4.4: ADC128S102 ディジタルピン

シリアル通信のため、アナログ信号を入力したら AD 変換後のデータが 1 clock で全て出力されるわけではなく、1 clock 毎に順に出力される。そして一度 ADC128S102 の AD 変換が on になると DOUT から出力されるディジタル信号が現在上位何ビット目なのかを知る方法がないので CS を含め、データ取得タイミングが重要になる。ADC128S102 のクロックタイミングを図 4.45 に載せる。

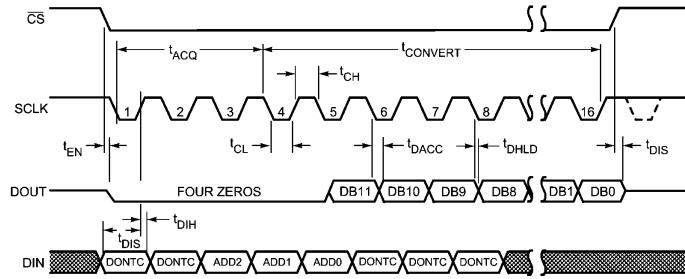


図 4.45: ADC128S102 Serial Timing Diagram[29]

まず始めに読み出すチャンネルを選択するよう ADC128S102 にチャンネルデータを入力する。入力が完了したらそのチャンネルに対応したピンに入力されたアナログ信号が AD 変換され MSB から順に 12 bit 分が DOUT から出力される。この仕組は Serial Peripheral Interface(SPI) と呼ばれる通信方法と良く似ている。SPI 通信は少ないピンでありながらデータ量に係わらず相手へと送ることができる。APMU による制御では ADC128S102 が 1 MSPS を実現するために後段の FPGA で 15 MHz クロックを供給している。図 4.46 に SPI のブロック図を記載する。図の MOSI というのは Master Out Slave In のことで、ADC128S102 では DIN にある。MISO はその逆で Master In Slave Out のことであり、DOUT に該当する。Slave Select の略である SS が ADC128S102 の CS と同等の役目を担う。

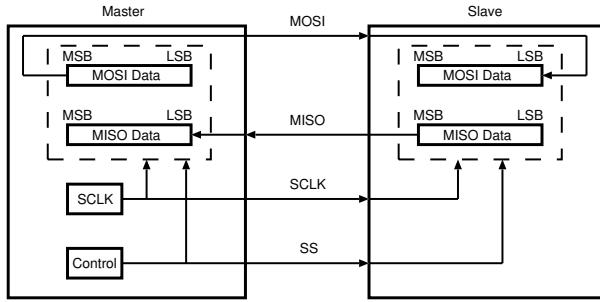


図 4.46: SPI 通信ブロック図

今回は Slave に該当する ADC128S102 からデータをもらうので SPI 通信での MISO モードとなり、その MISO モードの基本的なクロックタイミング(図 4.47)が図 4.45 ととても良く似ている。SPI 通信ではまず始めに Slave へ OP-Code と呼ばれるモードの選択データを送信する。Slave 側でこのデータを読み取り、MISO モードなのか MOSI モードなのかを判断し、もし MISO モードならばクロックに合わせてデータを Master へ送信する。これによって少ないピンでありながら Slave から Master へとデータの送信が可能である。ADC128S102 では OP-Code がアナログ入力チャンネルの選択となる。

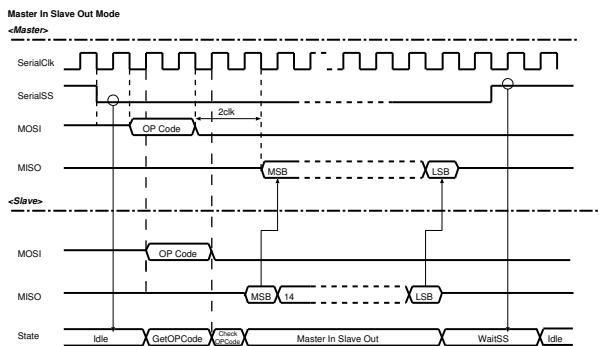


図 4.47: SPI Master In Slave Out Mode

4.6 ADC128S102 を用いた FIR フィルタ の性能評価

ADC128S102QML は宇宙用に開発されたものだが、今まで衛星で実際に使われた例がないためその性能を評価する必要がある。HXI/SGD のアナログ回路も含めたAPMU 評価基板BBM は2月に届き、届いた上でも優先させる試験項目が多いため、評価用基板を待っているのではプロジェクトのスケジュールとして間に合わない。そこで ADC128S102 評価用基板を自作し、それをFADC Board で制御することで ADC128S102 の評価を行った。また、 BGO と APD 素子、この ADC128S102 評価用基板と HXI/SGD で用いる PreAmp にほぼ近い HIC 評価回路 (HICPreAmp) を用いて実際の搭載回路に近いものを再現し、その状態での FIR フィルタを用いた波形取得、そして FPGA 上での FIR フィルタが低スレッショルドに効果があるのかということを調べた。

4.6.1 ADC128S102 評価用基板

これらの ADC128S102 基本性能、制御方法を元に評価用基板を作成した。基板製作における詳細は付録Bに載せている。評価用基板の全体回路図を図4.48に記載する。この評価用基板には入力にプリアンプ信号を想定しているが、プリアンプ信号は負の信号であり、ADC128S102 の入力レンジは正の信号でないといけないためアナログ入力から ADC128S102 までの間に反転回路を設置している。図4.48と図4.49が作成した基板の写真である。

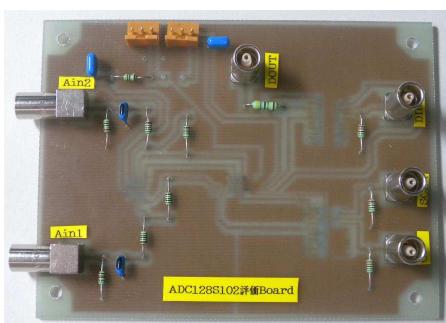


図 4.49: ADC128S102 評価基板表面 (コネクタ、受動素子実装側)

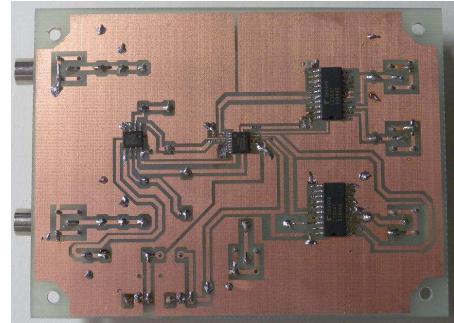
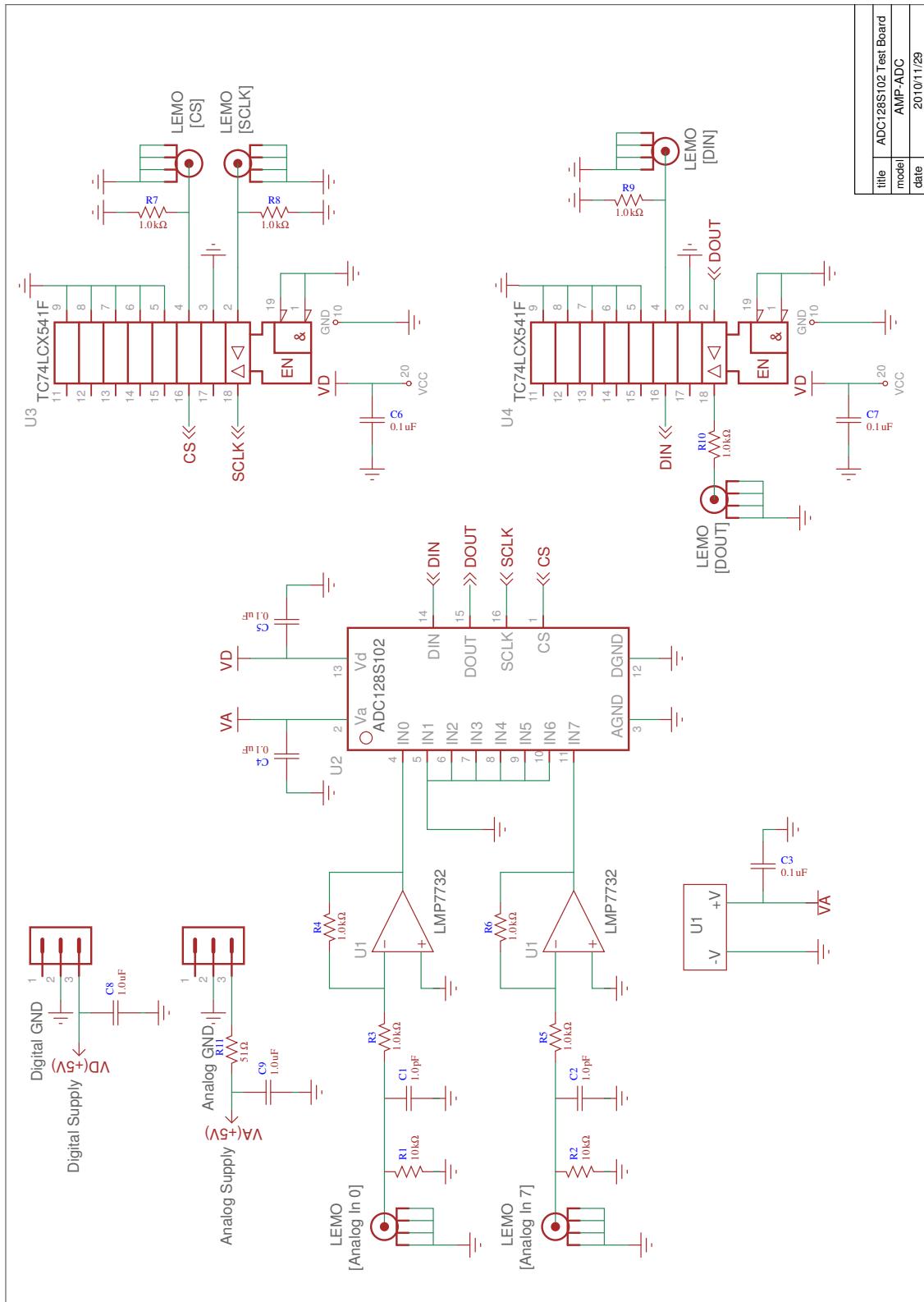


図 4.50: ADC128S102 評価基板裏面 (パターン、能動素子実装側)

人出力端子として記載する。

图 4.48: ADC128S102 芯片级基板回路图。ADC 套子の前段部分と 1 の接続部分、データ送り出し



ADC128S102 のクロックタイミングを元に FADC Board 上の FPGA で ADC128S102 コントロールモジュールと波形保存モジュールの開発をし、アナログ波形を取得した結果が図 4.51 である。これは Function Generator による 10 kHz の正弦波波形を入力している。このようにきれいな波形を取得できている。この制御・保存モジュールを用いて波形を取得し、フィルタの性能を評価する。

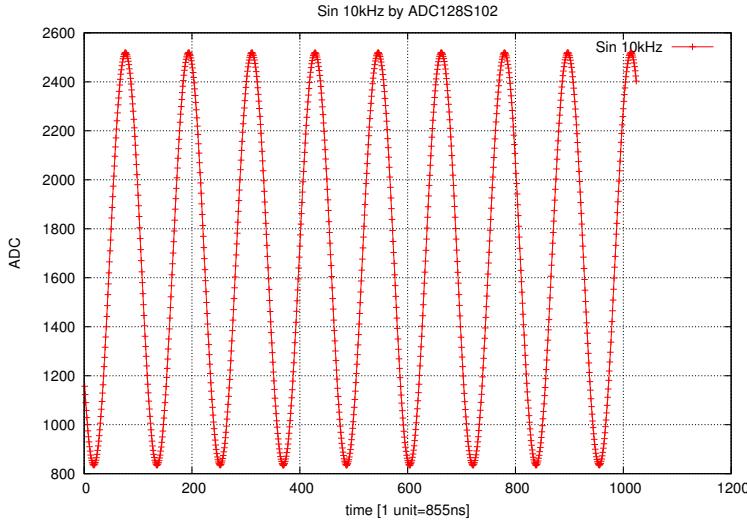


図 4.51: ADC128S102 によって取得した正弦波 (10 kHz) 信号波形

4.6.2 FPGA への FIR フィルタ組込み

FPGA 上での FIR フィルタ

4.4.5 節においてソフトウェアで FIR フィルタを実装して検証した。ここでは波形取得、FIR フィルタ、VETO 信号出力、パルスハイトヒストグラム取得の全てを FPGA 上に実装して実機同様の環境を作り、その上で FIR フィルタが低スレッショルドを達成するのかどうかの試験を行う。

FPGA 上での FIR フィルタは移動平均フィルタや差分フィルタと違って任意の係数での乗算、またそれらの加算・減算が複雑となる。また FPGA での乗算器にも限りがあるので乗算も工夫しなければならない。そこで FPGA 上での乗算にはビットシフトを利用した乗算を行った。26(二進数で 11010) を元にビットシフトの簡単な説明を図 4.52 に示す。ビットシフトとはその名前のように数字を左、または右にずらすことである。図 4.52 のように、左のシフトは二進数の桁が上がる所以数字が倍され、右のシフトは桁落ちする所以数字が半分になる。つまり、 n ビット左にシフトすれば 26×2^n となり、右にシフトすれば 26×2^{-n} となる。

$$26 = b''11010''$$

1 bit Left shift :	$0 \boxed{11010}0$	$= 52 = 26 \times 2^1$
2 bit Left shift :	$\boxed{11010}00$	$= 104 = 26 \times 2^2$
1 bit Right shift :	$000 \boxed{1101}$	$= 13 = 26 \times 2^{-1}$

図 4.52: $26=b'11010'$ のビットシフト

このビットシフトを用いた乗算を、例として 30×53 を用いて説明したものが図 4.53 である。まず 30 を $\times 32, \times 16, \times 8, \times 4, \times 2$ の場合で図 4.52 のようにビットシフトする。ここで乗算における係数を 2 の乗数の和に分解する。この場合には $53 = 32 + 16 + 4 + 1$ なので $30 \times 32, 30 \times 16, 30 \times 4, 30$ だけを取り出し、その和を取る。こうすることで $30 \times 53 = 1590$ を実現する。このビットシフト乗算は仕組が容易ながら整数のいかなる係数にも対応できる。FIR フィルタで負の係数を用いるときはこの計算を行った後で減算をすればよいので問題ない。

$$30 \times 53 = 30 \times (32 + 16 + 4 + 1)$$

$$30 = b''11110''$$

$30 \times 32 =$	$\boxed{11110}00000$	$\times 1 = 960$
$30 \times 16 =$	$0 \boxed{11110}0000$	$\times 1 = 480$
$30 \times 8 =$	$00 \boxed{11110}000$	$\times 0 = 0$
$30 \times 4 =$	$000 \boxed{11110}00$	$\times 1 = 120$
$30 \times 2 =$	$0000 \boxed{11110}0$	$\times 0 = 0$
+)	$30 \times 1 =$	$00000 \boxed{11110}$
		$= 1590$

図 4.53: ビットシフトを利用した乗算方法

このビットシフト乗算を用いて FPGA 上で FIR フィルタを実装し、三角波の波形を取得した。使用した FIR フィルタは係数が $[1, 0, -1]$ である。現在のデータと二つ前のデータの差分を取るというもので、式 4.3 で $M=2$ としたものと等しい。三角波なので電圧が増加している間は傾きが一定なので出力も一定となる。この図 4.54 でもそれが見られるため正しく動作していると思われる。

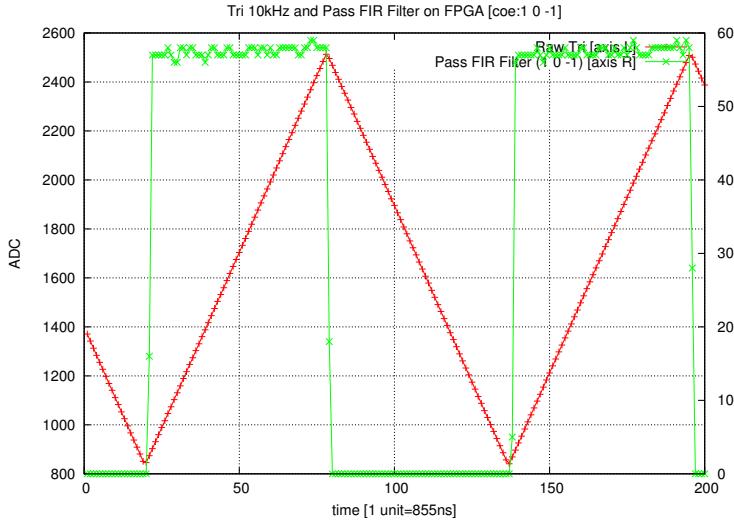


図 4.54: 三角波と FPGA 上で FIR フィルタ [係数 1,0,-1] を通した後の波形

FPGA でこの FIR フィルタを用いて波形の取得、及びヒストグラムの作成を行う。使用する FIR フィルタ係数は表 4.5 に載せている。これらはソフトウェアでのヒストグラムにおいて、係数が $\pm 1, 0$ のものである [1 1 1 1 0 -1 -1 0 0] と任意の数字で最も効果があった FIRAnalog2 Peak:60kHz オフセット 150% を選んでいる。ただし FPGA で小数のコーディングに対応していないので FIR8bit は FIR Analog2 Peak:60kHz オフセット 150% を元にして ± 255 に収まる整数に変換している。図 4.55 には元々の 2^{20} の精度を持つ小数における周波数特性と、今回用いる ± 255 整数における周波数特性の比較を載せる。図 4.25 にあったように、整数に変換しても係数精度が 2^8 ならば周波数特性に変化が無いため、このような ± 255 整数でも小数の場合と同じ効果があると期待される。

名称	係数条件	本研究で用いる係数値
FIR2bit	$\pm 1, 0$	1, 1, 1, 1, 0 , -1, -1, 0, 0,
FIR8bit	± 255 の整数	10, 19, 31, 48, 67, 82, 80, 51, 123, -51, -80, -82, -67, -48, -31, -19, -10

表 4.5: FPGA での FIR フィルタで用いるフィルタ係数

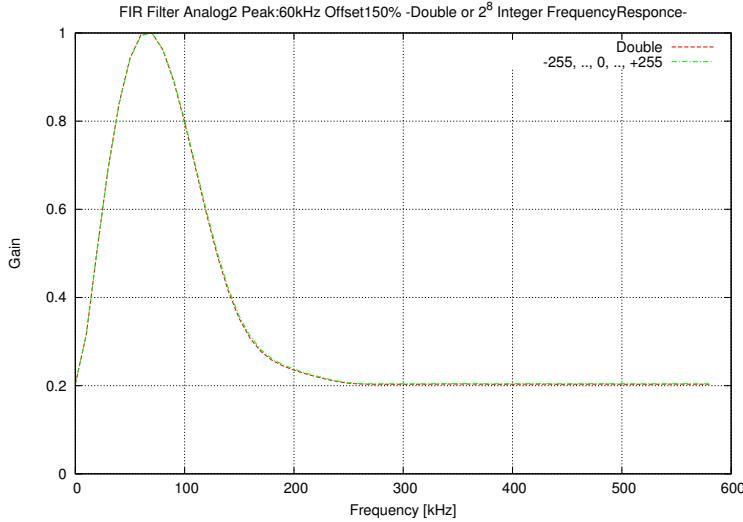


図 4.55: FIRAnalogue2 Peak:60kHz Offset150% Tap16 で係数が 2^{20} 小数 (赤) と 2^8 整数 (緑) での周波数特性

しかし、この ± 255 整数を用いることで FIR フィルタのゲインが約 100 前後と高くなる。図 4.56 は ^{241}Am 波形に対し表 4.5 の FIR8bit フィルタをソフトウェアで通す前後の波形である。この波形からソフトウェアでヒストグラムにし、 ^{241}Am の 59.5 keV ピークを取り出すと図 4.57 となる。このときフィルタ無しが大体 120 ch. ほどに、FIR8bit フィルタが大体 55000 ch. ほどにピークが来ているのでおよそ 460 倍のフィルタゲインであると分かる。これはフィルタ係数が ± 255 の整数なので FIR フィルタの入力と出力の関係式 4.7 より出力信号がおよそフィルタ係数の桁ほどに倍されるからである。

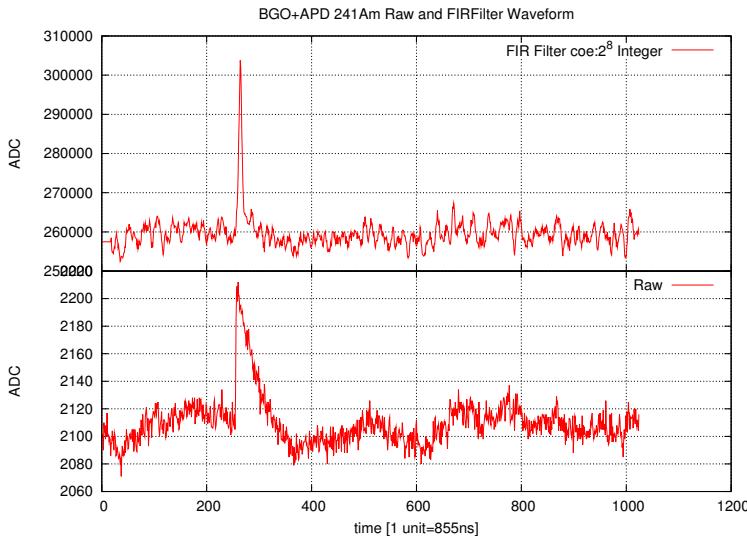


図 4.56: BGO+APD PreAmpOut ^{241}Am 波形 (下) とソフトウェアで FIR8bit フィルタを通した波形 (上)

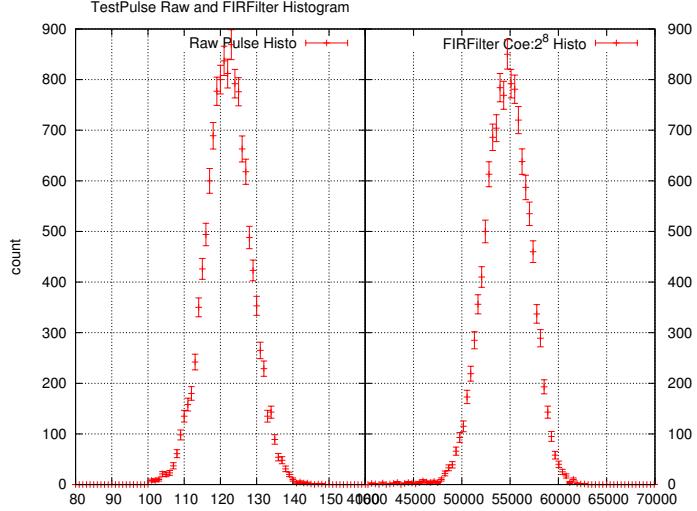


図 4.57: 図 4.56 の波形から得られた ^{241}Am 59.5 keV ヒストグラム (左: フィルタ前、右:FIR8bit フィルタ後)

入力と出力信号の大きさを同じぐらいにした方が処理の都合上良いので、フィルタゲインで出力信号を割る必要がある。フィルタゲインは図 4.57 から 460 となっている。ただし任意のフィルタ係数についてフィルタゲインを定義したいのでここではフィルタゲインを「フィルタ係数の正の数の和」と定義する。表 4.5 の FIR8bit フィルタだとフィルタゲインは 511 となり、ピークから求めた約 460 と 2 倍以上違わないので問題ない。FPGA では FIR フィルタの出力データに対し、このフィルタゲインで割る必要がある。ただし除算器は FPGA での容量を消費しそぎるため、除算はビットシフトで行った。まとめると FPGA での FIR フィルタは以下の式 4.18、式 4.19 となる。

$$FilterGain \equiv \sum_{m=0}^M \begin{cases} h_m & h_m \geq 0 \\ 0 & h_m < 0 \end{cases} \quad (4.18)$$

$$y[n] = \frac{1}{2^k} \sum_{m=0}^M h_m x[n] \quad (4.19)$$

$$2^{k-1} < FilterGain < 2^k$$

FIR フィルタを用いた FPGA でのデータ取得

HXI/SGD でのデータ取得環境を実験室で出来るだけ近付けたものが図 4.58 にあるセットアップ図である。ASTRO-H HIC 評価回路装置 (HIC PreAmp) に 1cm APD 素子をつけ、シンチレータに 1cm 角 BGO 結晶をセットした。APD 素子と BGO はアルミニウム箱の中に入れている。これらを恒温槽の中に入れ、実機で稼働する際の温度である -15 度にセットし、APD 素子には 400 V の電圧を与えた。これによって APD ゲインは 100 ほどにしている [6]。HIC 評価回路装置からのプリアンプ出力信号は ADC128S102 評価基板に入力する。このとき ADC128S102 評価基板は 3.3 V で動作している。この一連を FADC Board によって ADC128S102 制御、線源データの取得・処理を行った。

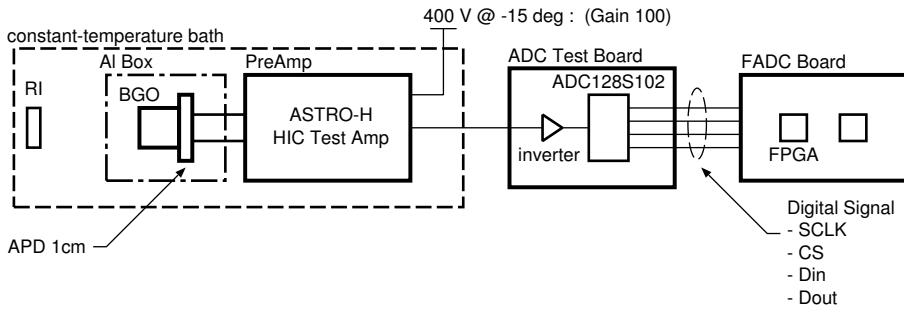


図 4.58: ADC128S102 評価基板を用いた線源データ取得環境セットアップ [BGO+APD(Gain 100, -15 度)]

これらセットアップにより、 ^{241}Am 60 keV の HICPreAmp 信号を取得した波形を図 4.59 に示す。これが正しく取得できているかどうかのため、比較ととして Tektronix 社製オシロスコープ TDS3032B にて取得した同じ 60 keV の HICPreAmp 出力信号を図 4.60 に示す。これらは同時に取得できないため、図 4.60 は同時刻の波形ではない。比較のため ADC128S102 信号は上下反転させており、また両信号のオフセットとパルスハイドが揃うように調整している。

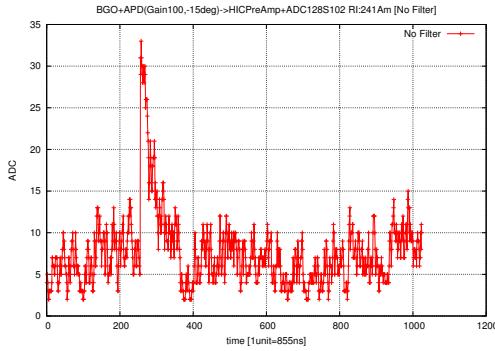


図 4.59: BGO+APD(Gain100,-15 度)

ADC128S102 を用いた ^{241}Am 波形 [フィルタ無し]

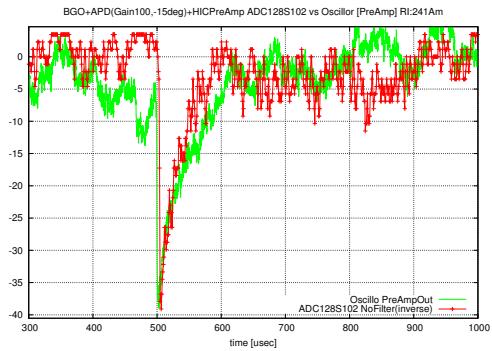


図 4.60: HICPreAmp 信号を ADC128S102(赤) と オシロスコープ(緑) で取得した波形

図 4.60 から波形の取得は問題なく行われているといえる。また表 4.5 による FIR フィルタ係数を用いて FPGA 上でフィルタを通した波形を図 4.61、図 4.62 に示す。これらはソフトウェアのときと違って各フィルタ毎に取得しているため波形は同じではない。

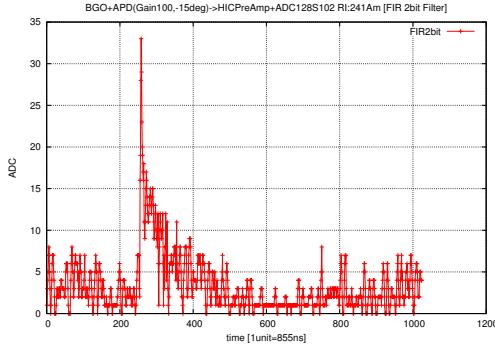


図 4.61: BGO+APD(Gain100,-15 度) ADC128S102 を用いた ^{241}Am 波形 [FIR 2bit フィルタ]

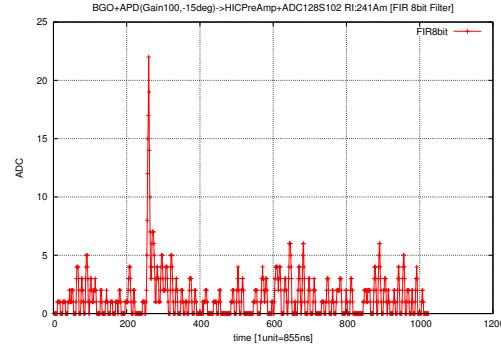


図 4.62: BGO+APD(Gain100,-15 度) ADC128S102 を用いた ^{241}Am 波形 [FIR 8bit フィルタ]

この波形がモデルとなったアナログシェイパと同様の波形になっているかを調べるために、図 4.63 に FIR8bit フィルタと Ortec Shaper($\tau=2\text{usec}$)の周波数特性を、図 4.64 にこのフィルタによって HICPreAmp 信号を整形した波形を載せる。

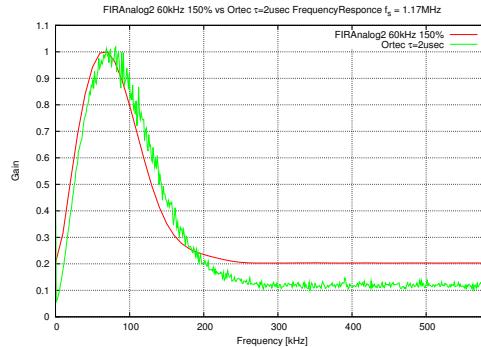


図 4.63: FIR8bit フィルタ(緑)と Ortec Shaper($\tau=2\text{usec}$)(赤)の周波数特性

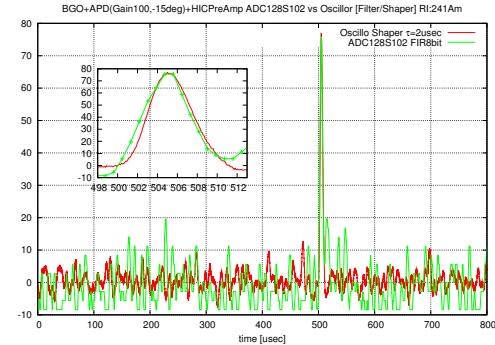


図 4.64: FIR8bit フィルタ(緑)と Ortec Shaper($\tau=2\text{usec}$)(赤)による HICPreAmp 波形の整形

この周波数特性・波形を見るに FIR8bit は波形整形の面では Shaper と同程度であると言える。しかしノイズの揺らぎは Shaper が圧倒的に少ない。この FIR フィルタを元に 3.2.2 節 PoGOLite での図 3.11・図 3.13 によるヒストグラム生成モジュールを基本とした FIR フィルタ用のヒストグラム生成モジュールを作成し、それによって取得したヒストグラムデータが図 4.65 と図 4.66 である。それぞれ ^{241}Am と ^{137}Cs のヒストグラムである。参考として、Ortec の Shaper と PocketMCA によるヒストグラムも載せてている。

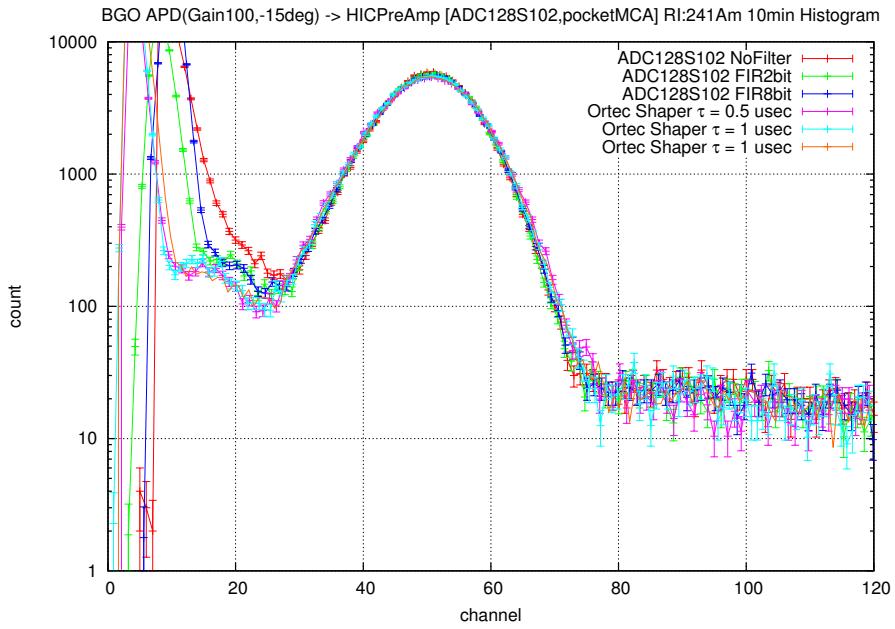


図 4.65: BGO+APD(Gain100,-15 度) ADC128S102 を用いて FPGA 上で取得した ^{241}Am ヒストグラム

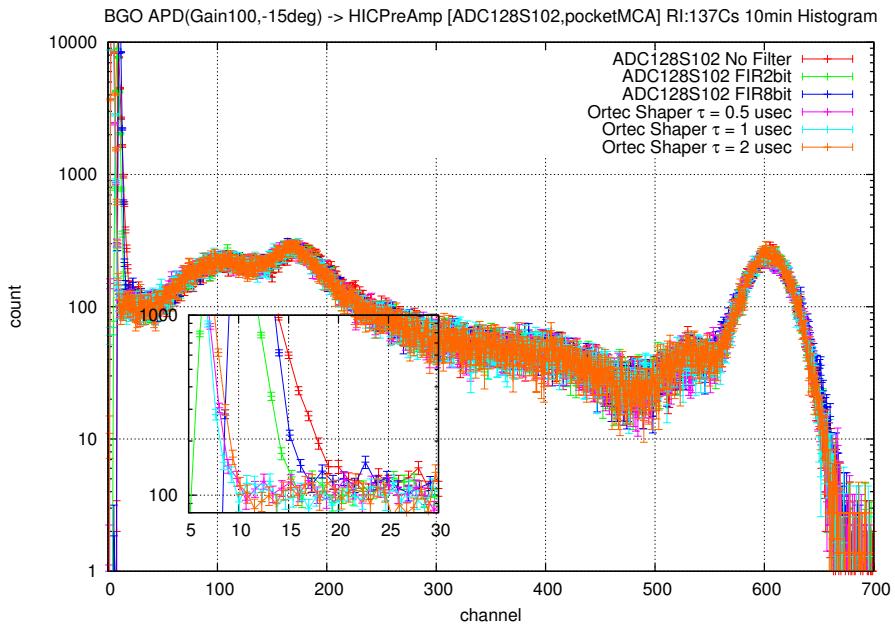


図 4.66: BGO+APD(Gain100,-15 度) ADC128S102 を用いて FPGA 上で取得した ^{137}Cs ヒストグラム

波形を処理するソフトウェア同様に FPGA というハードウェアでも FIR フィルタによって ^{241}Am と ^{137}Cs ともにスレッショルドが 20 から 15 へより低いスレッショルドを達成できている。よって 8 と 16 遅延器数のどちらのフィルタも効果があり、ノイズを約 3/4 に軽減できていることがわかった。

た。デジタルフィルタの周波数特性は Shaper のものに一致するように設定したが、粗い FIR2bit フィルタの方が精巧に合わせた FIR8bit フィルタよりも低いスレッショルドを得ている。今後はこの周波数特性を雛型として、より良い感度を達成できる係数を探していく。

Shaper を通した PocketMCA のデータと比較すると、60 keV と 662 keV のエネルギー分解能は、デジタルフィルタでもがアナログフィルタと同程度の性能を達成できている。一方でスレッショルドの絶対値は見かけ上アナログフィルタの方が低い値になっているが、これは PocketMCA とデジタルフィルタで波高値の求め方が異なるため、単純な比較はできない。PocketMCA ではその整形過程でオフセットを 0 V にし、0 V からの絶対値を波高値としてしている。ADC128S102 の場合には、オフセットをデータ処理のたびに求め直し、相対値として波高値を計算するため、ノイズの波高値が最大で PocketMCA の 2 倍となる。よってスレッショルドの値を両者で比較するためには、テストパルスを使い低エネルギー側のリニアリティを測定する必要がある。

第5章 まとめと今後

アクティブシールドにおける BGO 結晶の信号処理を行う上で PoGOLite では回路全体のシステム構築、ASTRO-H では HXI/SGD におけるディジタルフィルタを用いた信号処理を行った。以下にそれぞれについて簡単なまとめを述べる。

- PoGOLite

PoGOLite では先行研究によって主検出部である PDC 部とアクティブシールド部である SAS 部で独立に開発されていた回路を一つの共用回路にすることができた。これにより回路デバッグが 1 回路分になるため信頼性も香城氏、PDC 部では従来の波形保存機能に加え、ヒストグラム取得機能も追加されることになった。この共用回路を PoGOLite 実機に焼き込み、正常に波形保存、全チャンネルのヒストグラム取得が確認された。

また世界初の中性子シンチレータを搭載するためのシステム構築を行った。中性子シンチレータは他のユニットと異なり、偏光観測に関与していないため独立に動かすようにし、またシンチレーション光の減衰時定数の違いからサンプリング周波数を他のモジュールの 1/6 にダウンサンプリングすることができた。この中性子シンチレータシステムを用いて中性子照射実験を行い、熱中性子を効率よく検出することを確認した。

- ASTRO-H

ASTRO-H の HXI/SGD-APMUにおいては回路の実装面積の制限からアクティブシールド (BGO 結晶) の APD 素子による信号は低スレッショルドになるように FPGA でディジタルフィルタを用いた処理が行われる。本研究では衛星という FPGA 容量の限られた条件でも FIR フィルタはユーザの期待する周波数特性をとることを確認し、BGO 結晶と APD 素子、実機と同程度の性能を持つプリアンプ (HICPreAmp)、ADC128S102 を用いてアナログシェイバを再現したディジタルフィルタ (FIR フィルタ) を 1 から作成し、その性能を確認した。FPGA での低いスレッショルドを達成するためにまずアナログシェイバの周波数特性を調べてそのフィルタと最も近いと思われる式を導出、そして FIR フィルタの定義式から係数を算出した。FPGA で波形を取得してソフトウェアでパラメタを変えた FIR フィルタを作成し、それらの中で最もよいものを実機に組むことで低スレッショルドが達成できることを確認した。

今後は PoGOLite はこの夏に放球されるまで回路のデバッグを行い、万全の体制で望むようにする。ASTRO-H は正しいノイズの周波数特性を導き出し、どの周波数成分を通せばノイズと信号をより精度良く分離できるのかを詳細に吟味していく必要がある。そしてその結果をこの 2 月に新しく作成される APMU 評価基板を用いて実証し、どの程度低スレッショルドを達成できるのかを調べる。

付録A ディジタルフィルタ (Digital Filter)

一般にフィルタとはある特定の成分を弱める、あるいは強める効果を持つ機能のことを指す。画像処理において輝度変化が激しい箇所を抑えるようにしたり、カメラ等で撮影する際に波長や偏光などを制限したりと、一口にフィルタと言ってもその用途は多彩である。この研究でフィルタをかける対象は時間で変動する電気信号なので便宜上、フィルタを「特定の周波数成分を弱める、あるいは強める効果を持つ機能のこと」と定義する。

フィルタはその通過帯域によって大きく四種類に分けることが出来る。

- 低域通過フィルタ (Low Pass Filter : LPF)
- 高域通過フィルタ (High Pass Filter : HPF)
- 帯域通過フィルタ (Band Pass Filter : BPF)
- 帯域除去フィルタ (Band reject Filter : BRF)

これらは通過帯域 (pass band)、除去帯域 (stop band) の違いにより分類されているものである。例えば低域通過フィルタというものはその名前のように高周波成分は除去し、低周波成分のみを通過させるものである。激しく変動するノイズの中にデータ信号があった場合、この LPF に通せば高周波ノイズを除去してデータ信号をうまく引き出せることが期待できる。オフセットなどの信号を除去するときはオフセットが 0Hz の信号であることから高域通過フィルタを用いることもある。このようなフィルタを単体、あるいは複数組み合わせて用いることでノイズとデータ信号を切り離すようなフィルタを構築する。

上記の四種類が振幅比が 1 の通過域、振幅比が 0 の除去域のみに分けられる理想フィルタの場合、周波数ごとの振幅比は以下の図 A.1 のように表せる。図 A.1 における f_c 、 f_{cL} 、 f_{cH} は遮断周波数 (cutoff frequency) というもので、通過域と除去域の境界周波数である。

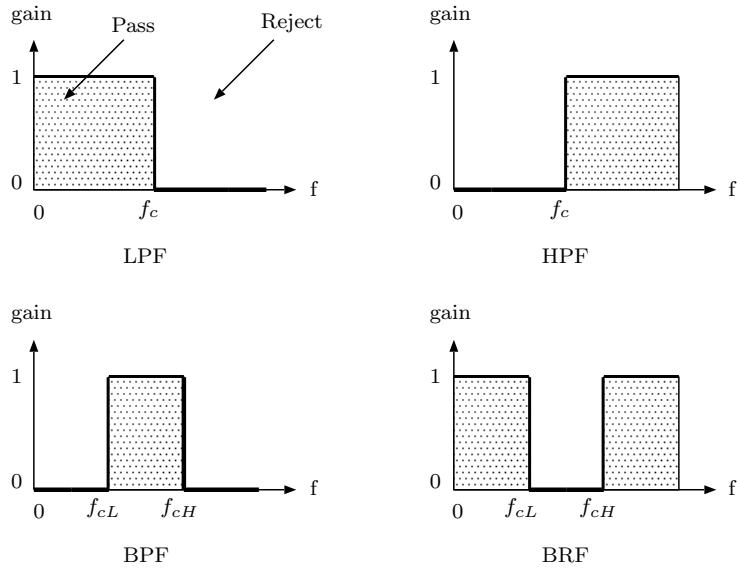


図 A.1: 各フィルタの通過帯域、除去帯域のようす

A.1 アナログフィルタとデジタルフィルタ

従来、信号の周波数成分を取り除くのに用いられていたのは抵抗やコンデンサ、状況によっては OP-Amp などの素子を用いたアナログフィルタで実現してきた。以下の図 A.2 が抵抗とコンデンサのみのアナログ Low Pass Filter である。

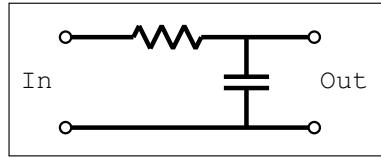


図 A.2: アナログ Low Pass Filter

以下の解析では抵抗 (R) とコンデンサ (C) の積で与えられる定数時定数 (time constant) を式 A.1 のように τ で表す。

$$\tau = RC \quad (\text{A.1})$$

図 A.2 の回路方程式から入力電圧 E_{in} と出力電圧 E_{out} の関係は式 A.2 のようになる。

$$\frac{dE_{out}}{dt} + \frac{1}{\tau} E_{out} = \frac{1}{\tau} E_{in} \quad (\text{A.2})$$

この式を線形微分方程式として解くと、 $E_{in} = E \sin(2\pi ft)$ という正弦波形を入力させた場合、回路出力波形との比は

$$\frac{E_{out}}{E_{in}} = |A| \sin(2\pi ft + \theta) \quad (\text{A.3})$$

$$|A| = \frac{1}{\sqrt{1 + (f/f_2)^2}} \quad \theta = -\tan^{-1}(f/f_2) \quad f_2 = \frac{1}{2\pi\tau}$$

となる。この式 A.3 より、 $f \gg f_2$ なら $|A| \simeq 0$ となり、 $f \ll f_2$ なら $|A| \simeq 1$ という LPF であることが分かる。例えば $C=500 \text{ pF}$ 、 $R=500 \Omega$ とした場合、 $f_2 = 637 \text{ kHz}$ の周波数で振幅比が $1/\sqrt{2} = 0.71$ 倍になる。 $C=500 \text{ pF}$ 、 $R=500 \Omega$ にした場合の計算による、周波数対振幅比応答を図 A.3 に載せる。

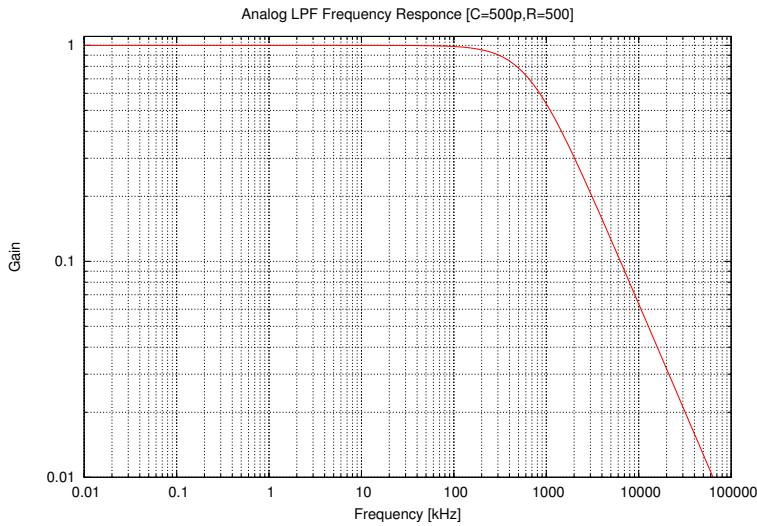


図 A.3: 図 A.2 において $C=500 \text{ pF}$ 、 $R=500 \Omega$ の場合の周波数特性

上図のような周波数に対する振幅比を表したやうすを周波数特性 (Frequency Responce) と呼ぶ。このようにアナログフィルタは抵抗、コンデンサ、あるいはインダクタなどの受動素子を組み合わせるだけでも簡単に実現できる。このような受動素子によるアナログフィルタは電力を必要とせず、シンプルな LPF であれば広い場所も必要としない。そのため、日常の様々な電気回路において安定供給を実現するために電源回路上や、信号の突発的な変化によって素子破壊が起きることを防止するために信号線上に LPF を設置してきた。

アナログフィルタはシンプル・省電力でありながらその効果は期待できるものではあるが、衛星上の回路においてはそれでは足りないところがある。アナログフィルタでは周波数帯域を詳細に決定できない、また自由に変更できない、また複雑な信号処理を行う場合、コストがかかる、実装回路が広面積になるというデメリットがあるからである。衛星の電源回路などは安定供給のために高周波数成分を削るだけでもよいが、検出器からの信号には高周波、低周波が入り交じった様々なノイズが含まれており、より高感度な観測を行うためにはそれらノイズは取り除きつつ必要な信号成分は残すようなフィルタにしなくてはならない。アナログフィルタでは我々の期待する周波数成分を満たすフィルタを作成しにくく、また検出器やその他のアナログ素子が経年劣化した場合、ノイズ周波数成分が変動する可能性があるため作成したアナログフィルタではそのノイズを落としにくくなることも考えられる。そのため、周波数特性に自由度があり、後々に透過帯域が更新できるフィルタが必要となる。そこで我々はフィルタにデジタルフィルタを用いることでそれを実現することにした。

デジタルフィルタは簡単な処理の場合、アナログフィルタと比べるとハイコストになってしま

うが、複雑な要求になればなるほどアナログフィルタよりもコスト増が抑えられる。また信号をデジタル信号として扱うため AD 変換後に関して言えば温度・湿度などによる経年変化が無く、常に安定した処理結果を期待でき、さらに続いて行われるデジタル処理にもデータの損失無く伝送することができる。LSI 化することで回路面積もその複雑さに応じて大きく変動しにくく、設計の上でも利点が大きい。そしてアナログフィルタでは実現が困難な周波数特性であってもデジタルフィルタでは計算が可能であれば原理的にいかなるものも作成可能であり、さらにパラメタ化することによりフィルタ仕様変更も容易である。一方、デジタルフィルタは素子を用いるため多少の電力を消費し、さらにその特性からナイキスト周波数以下の周波数帯域でしか効果がないというデメリットもある。また信号が入力され、その次の信号が入力されるまでに結果を出力しなければならないので GHz を超えるようなあまりに高周波な信号は扱えない。そのためデジタルフィルタの必要な箇所をきちんと認識し、またナイキスト周波数以上の周波数帯域を除去する場合はアナログフィルタと組み合わせるなどといったことも考慮する必要がある。

図 A.4 に最も簡単なデジタルフィルタを載せる。これは移動平均フィルタというもので、Low Pass フィルタの代表例である。 $x[n]$ は入力信号であり、 $y[n]$ は出力信号である。 n は時間を表し、信号は $x[n], x[n+1], x[n+2] \dots$ と続いている。 n の間隔が標本化したときのサンプリング周期に相当する。 Z^{-1} は遅延器というものでここを信号が通過するときは 1 clock 遅れて出力されることを意味する。

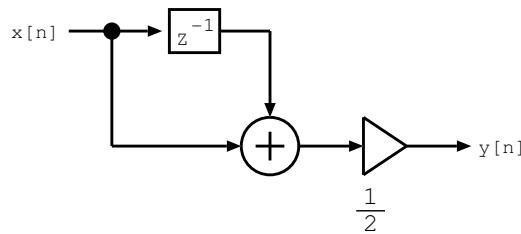


図 A.4: デジタル フィルタの例: 移動平均フィルタ

図 A.4 を入力信号 $x[n]$ と出力信号 $y[n]$ を用いてデジタル信号処理における処理方法を表した式である差分方程式 (differential equation) に書き表すと以下の式 A.4 になる。

$$y[n] = \frac{1}{2}(x[n] + x[n - 1]) \quad (\text{A.4})$$

つまりこの図 A.4 は二点の平均を取るもので、変化が激しい信号のときに効果を発揮し、ほとんど前後間で変化がなければ出力結果もそう変わらない。よって Low Pass フィルタというわけである。このフィルタを実例に応用したものが図 A.5 と図 A.6 である。これは広島県東広島市の 2010 年 1 月 1 日から 12 月 31 日における、気温 [32] の変化を表したものである。左図が気温を一時間毎に打点したものだが、一日内で大きく変動するため気温の大まかな変化が読み取りにくい。24 点の平均をとってプロットしたものが右図である。この平均化により一日以内の激しい変化がなくなり、全体の変化がわかりやすくなっている。厳密に言えば毎 24 点で平均を取ったのでは無く、一日の平均なので移動平均とは少し違うが、これもデジタルフィルタと言える。

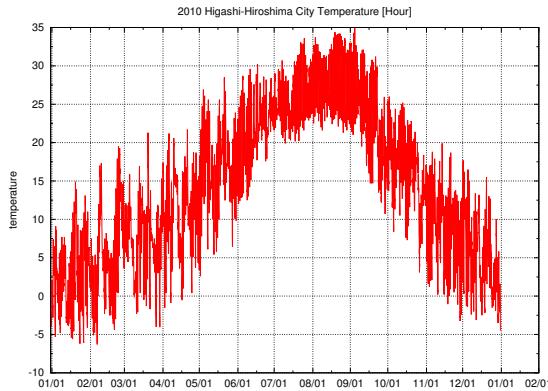


図 A.5: 2010 年東広島市気温 (1 点/1 時間) [32]

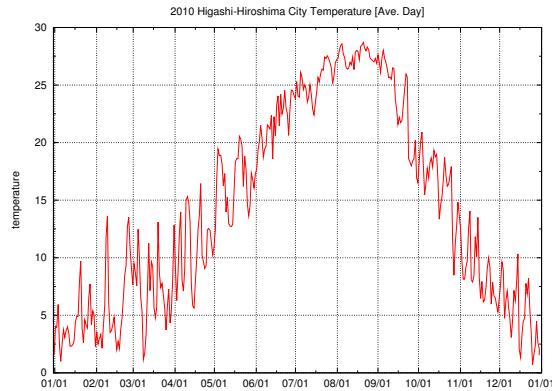


図 A.6: 2010 年東広島市の気温 (1 日平均)

A.2 ディジタル信号とディジタルフィルタ

A.2.1 ディジタル信号とは

ディジタルフィルタとあるように、信号はアナログ信号からディジタル信号にしなくてはならない。アナログ信号はある周期で標本化 (sampling) されて離散信号 (discrete signal) になり、素子などによって量子化 (quantization) される。この変換過程を AD 変換 (Analog-Digital Converter) と呼ぶ。これにより、標本化周期 T で取得した場合、連続的であった信号 $x(t)$ は離散信号 $x(nT)$ (n :整数) となる。 T は一定であり、表記しなくとも差し支えがないため省略する。また、連続か離散かを明確に区別するため、ここでは離散信号を $x[n]$ と書くこととする。AD 変換を行う簡単な様子を図 A.7 に示す。

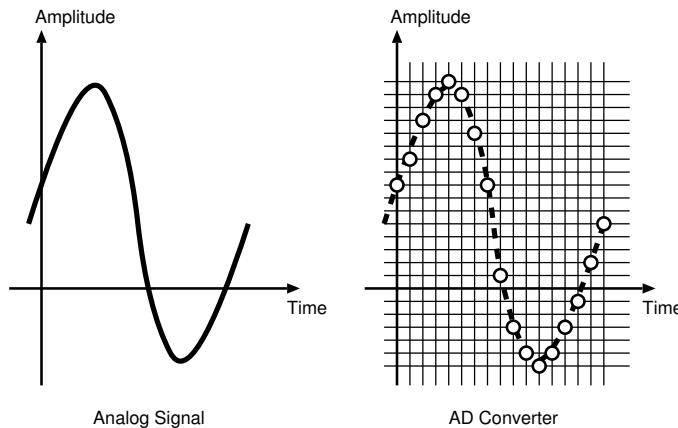


図 A.7: アナログ信号の AD 変換の様子

以下では標本化する際の周波数をサンプリング周波数 (sampling frequency) と呼ぶこととする。また、ディジタルフィルタの際に重要なのは AD 変換後のビット長の大きさではなくサンプリング周波数なのでここでは量子化は十分なビット長で行われたものとし、連続量であるとする。以下の話ではこのような信号をディジタル信号と仮定して進める。

A.2.2 フィルタブロック図、差分方程式から伝達関数と周波数特性まで

デジタルフィルタブロック図

デジタルフィルタを作成する上で重要なことはそのフィルタがどのような形で、どのような周波数特性を持つのかということである。フィルタの形は概ね図 A.4 にあったようなブロック図ではなくて、とんど全て表すことができる。図 A.8 にブロック図で用いられる基本要素を載せる。

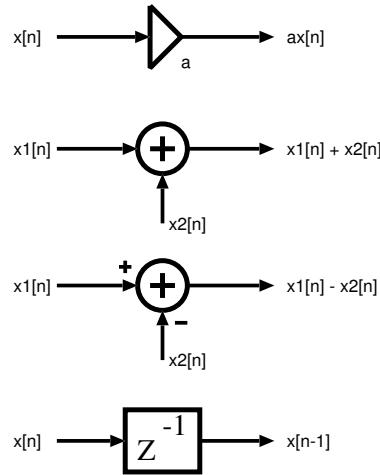


図 A.8: デジタルフィルタブロック図基本コンポーネント

差分方程式

ここからは簡単な基本的なフィルタを元に周波数特性を導出するまでを説明する。図 A.8 を元に以下の図 A.9 にある移動平均フィルタを作ったとする。

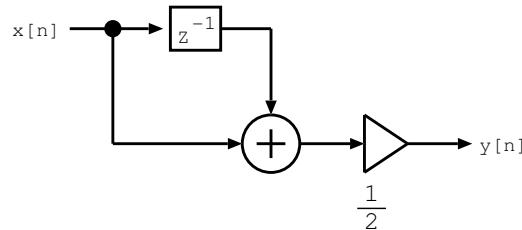


図 A.9: 移動平均フィルタのブロック図 (図 A.4)

この図は下のような差分方程式 (A.5) で示されるブロック図である。

$$y[n] = \frac{1}{2}(x[n] + x[n-1]) \quad (\text{A.5})$$

伝達関数と周波数特性

この式のままだとこのフィルタがどのような周波数特性を持つのかは、直感として LPF であると分かるが、定量的には分からぬ。Z 変換 (z-transform)¹ というものを用いる。Z 変換とは離散群上のラプラス変換 (Laplace transform)、または離散フーリエ変換 (Discrete Fourier transform:DFT) の拡張と説明されている。離散信号を $x[n]$ 、それを Z 変換したものを $X(z)$ とし、さらに $n < 0$ において $x[n] = 0$ の場合、二つは以下の式 A.6 を満たす。

$$X(z) = \sum_{n=0}^{\infty} x[n]z^{-n} \quad (\text{A.6})$$

この式を $X(z) = \mathcal{Z}\{x[n]\}$ で表すこともある。 $X(z)$ の z は $z = \exp(i\omega T)$ とおけ、更に $\omega = 2\pi f$ 、 $T = 1/f_s$ とおける。 f_s はサンプリング周波数なので $X(z)$ は周波数 f の関数であると分かる。ここで $x[n-1]$ の信号を Z 変換してみる。 $x[n-1]$ の信号を Z 変換したものを $X'(z)$ とすると以下のようになる。

$$\begin{aligned} X'(z) &= \sum_{n=0}^{\infty} x[n-1]z^{-n} \\ &= x[-1] + x[0]z^{-1} + x[1]z^{-2} + x[2]z^{-3} \dots \\ &= x[-1] + \sum_{n=0}^{\infty} x[n]z^{-n-1} \\ &= \left(\sum_{n=0}^{\infty} x[n]z^{-n} \right) z^{-1} \\ &= X(z)z^{-1} \end{aligned} \quad (\text{A.7})$$

$n < 0$ では $x[n]=0$ であることに注意する。これによって $x[n-1]$ の Z 変換は $X(z)z^{-1}$ とおくことができる。これは一つ遅れた信号 $x[n-1]$ の場合だが、一般化した $x[n-k]$ においても同様の変換が可能である。 $x[n-k]$ の Z 変換式を以下の式 A.8 に載せる。

$$\begin{aligned} X_k(z) &= \sum_{n=0}^{\infty} x[n-k]z^{-n} \\ &= X(z)z^{-k} \end{aligned} \quad (\text{A.8})$$

この Z 変換の性質を利用すると式 A.5 を Z 変換するときにひじょうに役に立つ。式 A.5 を Z 変換したものを式 A.9 に示す。

$$\begin{aligned} Y(z) &= \frac{1}{2} (X(z) + X(z)z^{-1}) \\ &= \frac{1}{2}(1 + z^{-1})X(z) \end{aligned} \quad (\text{A.9})$$

¹本研究では因果的信号、すなわち $n < 0$ に対して $x[n] = 0$ となる信号を扱うため、Z 変換は全て片側 Z 変換である。これに対して両側 Z 変換というものもあり、こちらは以下の式で表される。

$$X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n}$$

$X(z)$ は入力信号の周波数成分を表しており、 $Y(z)$ は出力成分のそれを表している。つまり、その両者の比を取れば作成したデジタルフィルタの周波数特性を知ることができる。そこで $H(z) = Y(z)/X(z)$ とし、この $H(z)$ を式 (A.9) に当てはめると式 (A.10) となる。

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{2}(1 + z^{-1}) \quad (\text{A.10})$$

この $H(z)$ を伝達関数 (transfer function) と呼ぶ。Z 変換の定義より $z = \exp(i\omega T)$ とおくと、伝達関数は式 (A.12) となる。

$$H(\omega) = \frac{1}{2}(1 + \exp(-i\omega T)) \quad (\text{A.11})$$

この伝達関数の値は複素数である。伝達関数には入出力の振幅比に対する周波数特性(振幅特性)と、入出力の位相差に対する周波数特性(位相特性)の二つの情報を持つ。振幅特性は伝達関数の絶対値に対応し、位相特性は偏角に対応する。よって伝達関数は次のようにも書ける。

$$H(\omega) = |H(\omega)| \exp(i\theta(\omega)) \quad (\text{A.12})$$

ここでは特に振幅特性が重要なので、振幅特性を周波数特性として扱う。この移動平均フィルタの周波数特性はオイラーの公式や倍角の公式などを用いて以下のようなになる。

$$\begin{aligned} |H(\omega)| &= \frac{1}{2}|(1 + \exp(-i\omega T))| \\ &= \frac{1}{2}|1 + \cos(\omega T) - i \sin(\omega T)| \\ &= \frac{1}{2}\sqrt{1 + \cos(\omega T)^2 + \sin(\omega T)^2} \\ &= \frac{\sqrt{2}}{2}\sqrt{1 + \cos(\omega T)} \\ &= \frac{1}{2}\frac{\sin(\omega T)}{\sin(\frac{\omega T}{2})} \end{aligned} \quad (\text{A.13})$$

上で述べたように $\omega T = 2\pi f/f_s$ なのでサンプリング周波数 f_s を決めれば伝達関数の絶対値式 A.13 から周波数特性が分かる。 f_s を 1MHz として周波数特性を求めてプロットしたものが図 A.10 である。この図から移動平均フィルタは LPF ということが定量的に分かる。また二点の平均を取る、というこのフィルタは直感ではサンプリング周波数と同じ周波数の箇所でしか影響がないようにも思えるが、図 A.10 を見ると全ての帯域でなんらかの影響があることが分かる。

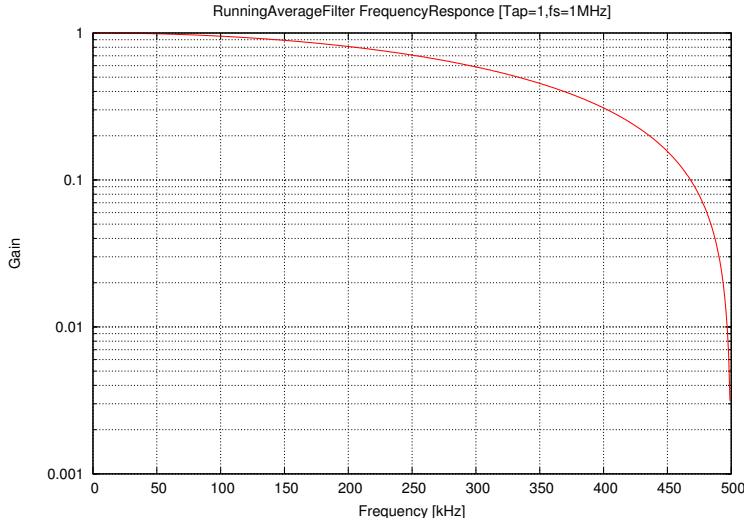


図 A.10: 移動平均フィルタ周波数特性 (Tap=1, $f_s = 1\text{MHz}$)

以上が移動平均フィルタにおけるブロック図から周波数特性までを導出する道筋である。このやり方はブロック図さえ分かっていれば知る限り全てのフィルタにも適応できる。HPF である差分フィルタ、共振器などの単純なものから FIR(Finite Impulse Response) フィルタといった複雑なフィルタまで周波数特性を求める基礎と言える。

A.3 ディジタルフィルタの例

ここでは遅延器と簡単な係数のみで構成される基本的なフィルタの紹介と伝達関数、周波数特性を求める。

A.3.1 移動平均フィルタ (LPF)

移動平均フィルタは先ほど求めたように LPF である。図 A.9 にあるように遅延器の数が 1 個の場合だとナイキスト周波数近くまでの帯域は通過するようなものだが、ここではより一般化したものを取り扱う。

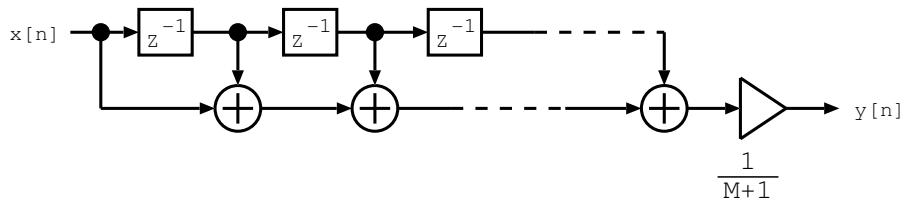


図 A.11: 移動平均フィルタの一般的なブロック図

このブロック図から任意の点数分を平均するフィルタであると分かる。この遅延器の数を M とすると差分方程式は以下の式 A.14 になる。

$$y[n] = \frac{1}{M+1} \sum_{m=0}^M x[n-m] \quad (\text{A.14})$$

この式を Z 変換して伝達関数を求めたものと周波数特性を算出したものが次のような式 A.15、A.16 になる。

$$H(z) = \frac{1}{M+1} \sum_{m=0}^M z^{-m} \quad (\text{A.15})$$

$$|H(\omega)| = \frac{1}{M+1} \left| \frac{\sin\left(\frac{M+1}{2}\omega T\right)}{\sin\left(\frac{\omega T}{2}\right)} \right| \quad (\text{A.16})$$

この周波数特性式からさきほどと同じように周波数特性を遅延器数別にプロットしたものが図 A.12 である。これも $f_s = 1\text{MHz}$ としてプロットしている。

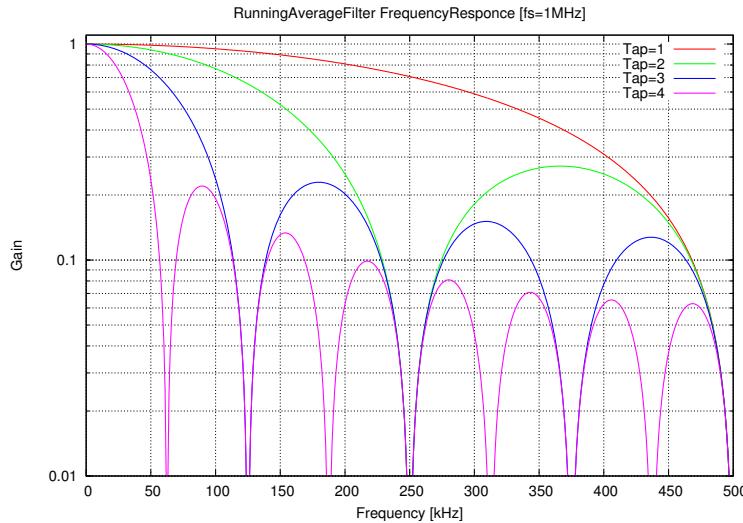


図 A.12: 一般的な移動平均フィルタ周波数特性 ($f_s = 1\text{MHz}$)

遅延器数が増えるにつれて通過する帯域に制限がかかっているのが分かる。しかし、単純な作りで実現できるというメリットがある一方、遅延器の数でのみしか周波数特性に変化を与えられないので自由に帯域を変更する、ということは難しい。例えば図 A.12 からは、300kHz を遮断周波数とすることは難しいと分かる。更に遮断する高周波領域にもリップルのようなものが見られるため、それがシステム上許容なのかどうかを知った上で扱う必要がある。

A.3.2 差分フィルタ (HPF)

差分フィルタも移動平均フィルタと同様、単純な仕組ながら効果が高いフィルタである。差分フィルタのブロック図を以下の図 A.13 に載せる。

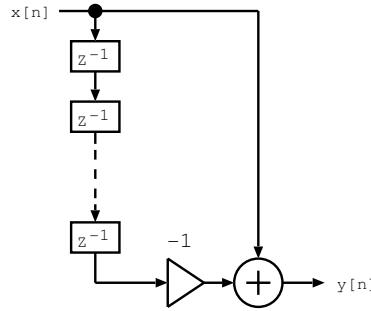


図 A.13: 差分フィルタの一般的なブロック図

この図から分かるように、現在の信号と、過去の信号の差分を取るようなものとなっている。そのため変化があまりない信号の場合は差分結果が 0 に近付き、逆に変化が大きい信号の場合は差分結果が大きい値となる。そのため差分フィルタはオフセットや低周波信号をカットし、高周波信号を透過させる High Pass Filter(HPF) となる。差分フィルタのブロック図から差分方程式を求めるところ以下の式 A.17 になる。

$$y[n] = x[n] - x[n - M] \quad (\text{A.17})$$

M は移動平均フィルタ時同様、遅延器の数である。この差分方程式を Z 変換し、伝達関数と周波数特性を求めるところ次のようない式 A.18、A.19 になる。

$$H(z) = 1 - z^{-M} \quad (\text{A.18})$$

$$|H(\omega)| = 2 \left| \sin \frac{M\omega T}{2} \right| \quad (\text{A.19})$$

この周波数特性式をもとに遅延器数別に図に表したもののが以下の図 A.14 である。サンプリング周波数を $f_s = 1\text{MHz}$ とし、遅延器数を 4 つまで用いた場合をプロットしている。

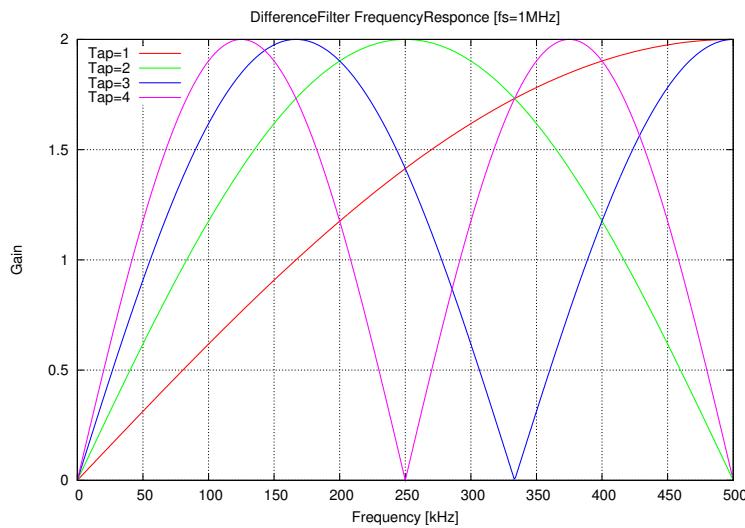


図 A.14: 一般的な差分フィルタ周波数特性 ($f_s = 1\text{MHz}$)

グラフの見やすさから対数ではなく、線形グラフを用いている。遅延器数によらずオフセット(0 Hz)を含む低周波帯域をカットしていることから差分フィルタはHPFであると分かる。しかし、このフィルタも移動平均フィルタ同様、単純なフィルタなので実装しやすいメリットがあるが、周波数特性は遅延器数にのみ依存しているので自由に周波数特性を変えるというわけにはいかない。また、差分の特性上、元の波形を増分することもあるのでその点にも注意する必要がある。図 A.14 にあるように、遅延器の数によっては単純な HPF ではなく、山状の透過域が出来てしまってしまうこともある。移動平均フィルタ同様、使い方に注意する必要がある。

A.3.3 共振器

遅延器のと乗算器の組合せにより、特定周波数を強める共振器を作成することも可能である。図 A.15 に共振器のブロック図を記載する。

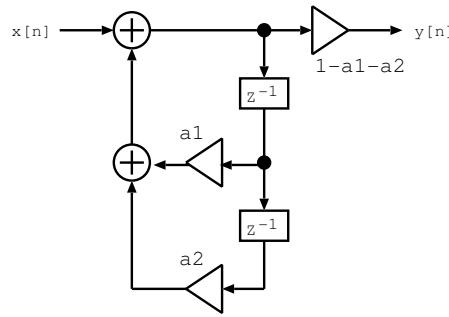


図 A.15: 共振器ブロック図

移動平均フィルタや差分フィルタと違ってフィードバックが存在する。これにより特定の周波数を繰り返し上乗せすることで共振を実現する。以下の式 A.20 に差分方程式を記載する。 a_1 と a_2 はブロック図における乗算の係数だが、式 A.21 にあるように共振器の共振周波数 F_0 、共振の帯域幅 B_0 から計算することが可能である。

$$y[n] = a_1 y[n-1] + a_2 y[n-2] + (1 - a_1 - a_2)x[n] \quad (\text{A.20})$$

$$\begin{aligned} a_1 &= 2 \exp(-\pi B_0 T) \cos(2\pi F_0 T) \\ a_2 &= -\exp(-2\pi B_0 T) \end{aligned} \quad (\text{A.21})$$

伝達関数と周波数特性を式 A.22 と式 A.23 に記載する。またこれによって周波数特性をプロットしたものを図 A.16 に記載する。これは共振周波数 F_0 をサンプリング周波数の $1/6$ (この場合 167 kHz) にし、帯域幅 B_0 を変えてプロットしている。

$$H(z) = \frac{1 - a_1 - a_2}{1 - a_1 z^{-1} - a_2 z^{-2}} \quad (\text{A.22})$$

$$|H(\omega)| = \frac{|1 - a_1 - a_2|}{\sqrt{1 + a_1^2 + a_2^2 + 2a_1(a_2 - 1)\cos(\omega T) - 2a_2\cos(2\omega T)}} \quad (\text{A.23})$$

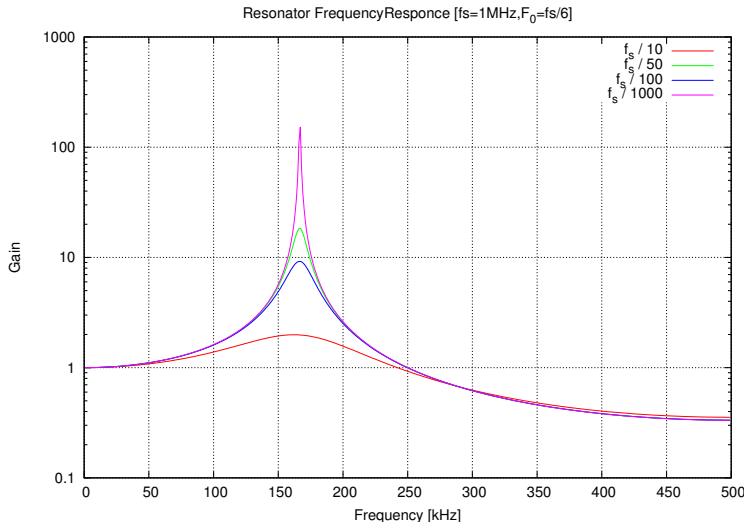


図 A.16: 共振器の周波数特性 ($f_s = 1\text{MHz}$, $F_0 = f_s/6$)

A.3.4 ノッチ・フィルタ

遅延器のと乗算器の組合せにより、共振器とは逆に特定周波数を弱めるフィルタ（ノッチフィルタ）を作成することも可能である。図 A.17 にブロック図を記載する。

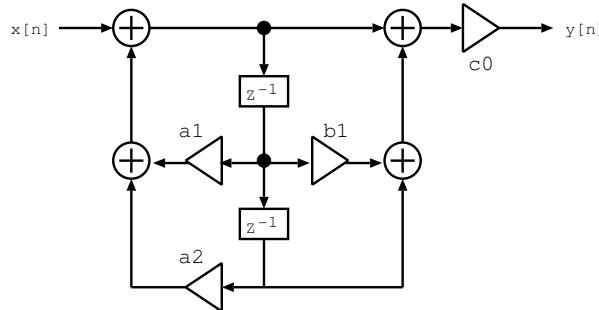


図 A.17: ノッチフィルタブロック図

以下の式 A.24 に差分方程式を記載する。 a_1 と a_2 、 b_1 、 c_0 はブロック図における乗算の係数だが、共振器同様に式 A.25 からノッチフィルタの除去周波数 F_0 、ノッチフィルタの帯域幅 B_0 から計算することが可能である。

$$y[n] = a_1 y[n-1] + a_2 y[n-2] + c_0(x[n] + b_1 x[n-1] + x[n-2]) \quad (\text{A.24})$$

$$\begin{aligned}
a_1 &= 2 \exp(-\pi B_0 T) \cos(2\pi F_0 T) \\
a_2 &= -\exp(-2\pi B_0 T) \\
b_1 &= -2 \cos(2\pi F_0 T) \\
c_0 &= \frac{1 - a_1 - a_2}{2 + b_1}
\end{aligned} \tag{A.25}$$

伝達関数と周波数特性を式 A.26 と式 A.27 に記載する。またこれによって周波数特性をプロットしたものを図 A.18 に記載する。これはノッチフィルタの除去周波数 F_0 をサンプリング周波数の $1/6$ (この場合 167 kHz) にし、帯域幅 B_0 を変えてプロットしている。

$$H(z) = \frac{c_0(1 + b_1 z^{-1} + z^{-2})}{1 - a_1 z^{-1} - a_2 z^{-2}} \tag{A.26}$$

$$|H(\omega)| = \frac{|c_0|^2 \{2 + b_1^2 + 4b_1 \cos(\omega T) + 2 \cos(2\omega T)\}}{\sqrt{1 + a_1^2 + a_2^2 + 2a_1(a_2 - 1) \cos(\omega T) - 2a_2 \cos(2\omega T)}} \tag{A.27}$$

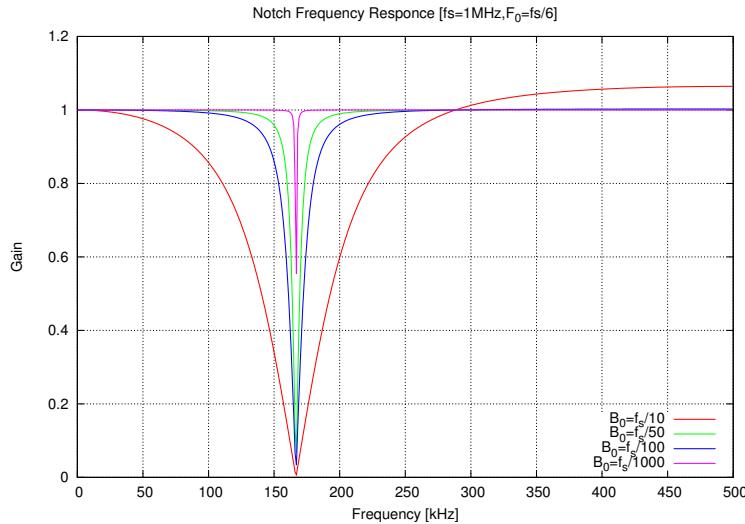


図 A.18: ノッチフィルタの周波数特性 ($f_s = 1\text{MHz}$, $F_0 = f_s/6$)

A.4 FIR フィルタ

移動平均フィルタや差分フィルタは遅延器と加算器の簡単な組合せだったためソフトウェア、ハードウェア問わず作成が容易であり、評価も楽に行えたが、自由に周波数特性を決められない、微調整がほとんど行えないところが難点であった。複数のフィルタを組み合わせることで周波数特性に自由度を与えることも可能であるが、FPGA のレジスタ値を少し変更するだけでそれが反映されるものを作りたい。そこで我々は任意の周波数特性を取ることが出来る FIR フィルタを用いることにした。

A.4.1 FIR フィルタ

フィルタの特性を調べるために、フィルタにインパルス信号 $\delta[n]$ を入力してその応答を見る場合がある。それをフィルタのインパルス応答 (impulse response) という。インパルス信号は以下の式 (A.28) で表される。

$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \quad (\text{A.28})$$

つまり、ある瞬間に 1 という信号を入力してその応答を見るというものである。このインパルス応答による出力信号を $h[n]$ とする。この $h[n]$ は 1 というインパルス信号を入力したときの出力であるが、次に入力信号を $x[n]$ にしたときにどうなるか考える。時間 n のときの入力信号は $x[n]$ 、このときの出力信号は現在までの入力信号 $x[n]|_{n=0,1,\dots,n}$ のインパルス応答の和となる。すなわち、式 (A.29) となる。

$$y[n] = \sum_{m=0}^n h[n-m] \cdot x[m] = h[n] * x[n] \quad (\text{A.29})$$

これは $m = 0$ のときの入力信号 $x[0]$ によるインパルス応答が位置 n に与える影響、 $m = 1$ のときの入力信号 $x[1]$ による影響、…と入力信号の各位置が与える影響を足し合わせたものが位置 n での出力信号 $y[n]$ になっているという解釈である。 $*$ はこの \sum 式を演算で表したもので、畳み込み演算 (convolution) という。この $y[n]$ を Z 変換すると、Z 変換の性質から、式 (A.30) のようになる。

$$\mathcal{Z}\{y[n]\} = \mathcal{Z}\{h[n] * x[n]\} = \mathcal{Z}\{h[n]\} \cdot \mathcal{Z}\{x[n]\} \quad (\text{A.30})$$

伝達関数の $H[z]$ を用いると、式 A.31 のようになる。

$$H(z) = \frac{\mathcal{Z}\{y[n]\}}{\mathcal{Z}\{x[n]\}} = \mathcal{Z}\{h[n]\} \quad (\text{A.31})$$

つまり、インパルス応答 $h[n]$ の z 変換が伝達関数 $H(z)$ (または $H(z)$ の逆 z 変換が $h[n]$ となる) このインパルス応答 $h[n]$ が有限時間内に 0 になるか、そうでないかで二種類のフィルタに分けることが出来る。それぞれのフィルタを FIR(finite impulse response) フィルタ、IIR(infinite impulse response) フィルタと呼ぶ。それぞれの特徴を伝達関数で比較すると、FIR フィルタは z^{-1} の多項式、IIR フィルタは z^{-1} に関する有理関数となっている。そのため、IIR フィルタは $|H(z)|$ が無限になる可能性があり、不安定になる場合がある。両フィルタの簡単な比較を表 A.1 に示す。

Filter Name	FIR フィルタ	IIR フィルタ
インパルス応答継続時間	有限	無限
差分方程式	$y[n] = \sum_{m=0}^M h_m x[n-m]$	$y[n] = \sum_{m=1}^M a_m y[n-m] + \sum_{k=0}^K b_k x[n-k]$
伝達関数	$H(z) = \sum_{m=0}^M h_m z^{-m}$	$H(z) = \frac{\sum_{k=0}^K b_k z^{-k}}{1 - \sum_{m=1}^M a_m z^{-m}}$

表 A.1: FIR フィルタと IIR フィルタの比較

ディジタルフィルタを実現する上で、通過域から除去域への遷移(あるいはその逆)が理想フィルタのように急峻にならないことがある。これは理想フィルタが無限個の遅延器を持つことに由来しているからであるが、IIR フィルタは比較的少ない遅延器で十分急峻な周波数特性を持つことが可能である。しかし、IIR フィルタは過去の信号をフィードバックするため演算誤差が大きく表れる可能性があり、またうまく扱わないと出力信号が発振してしまうことがある。我々は現在、急峻な遷移域は必要としていないのでここでは扱いやすい FIR フィルタについて述べる。

基本的な FIR の差分方程式、および伝達関数の一般形は式 A.32、A.33 である。

$$y[n] = \sum_{m=0}^M h_m x[n-m] \quad (\text{A.32})$$

$$H(z) = \sum_{m=0}^M h_m z^{-m} \quad (\text{A.33})$$

この式を崩さず、ハード的に遅延器、增幅器の配置を変えることによって、直接形(direct form)、転置形(transposed form)、継続系(cascade form)、格子形(lattice form)などの構成を取ることが出来るが、構成が最も簡単である直接形を用いることにする。FIR フィルタ直接形のブロック図を以下の図 A.19 に示す。

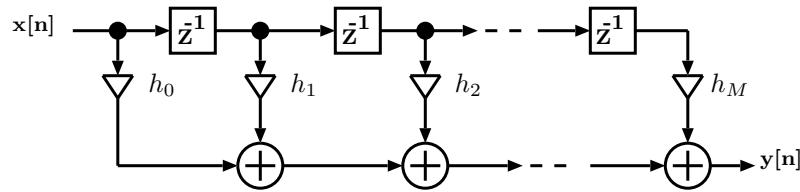


図 A.19: FIR フィルタブロック図(直接形)

このように FIR フィルタは多段遅延器と、各遅延後の信号に重みをかけ、それらを合成したものである。その重み $h_m (m = 0, 1, \dots, M)$ をフィルタ係数と呼び、FIR フィルタでは周波数特性を決定する重要な因子である。

$$h_m = \begin{cases} 1/2 & m = 0, 1 \\ 0 & m \neq 0, 1 \end{cases} \quad (\text{A.34})$$

と係数を決定すれば図 A.9 にある遅延器数が 1 の移動平均フィルタブロック図を再現でき、また

$$h_m = \begin{cases} 1 & m = 0 \\ -1 & m = M' \\ 0 & m \neq 0, M' \end{cases} \quad (\text{A.35})$$

と係数を決めれば図 A.13 にある差分フィルタを再現できる。つまり移動平均フィルタも差分フィルタも FIR フィルタの一種である。移動平均フィルタと差分フィルタは遅延器の数しか変更できなかったが、FIR フィルタはそれに加えてフィルタ係数が変更可能なため、かなり自由な周波数特性をとることが期待できる。例えば FIR フィルタを用いれば図 A.20 のような透過域を持つ BPF も作成可能である。これは 100 - 300 kHz を透過するフィルタで、遅延器数は 1024 である。

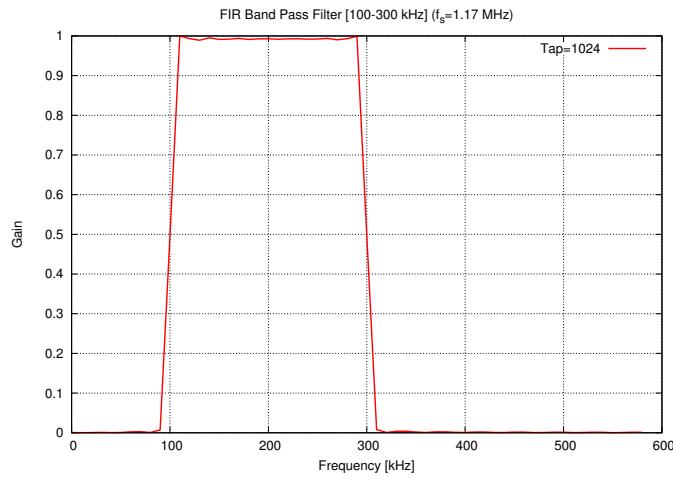


図 A.20: FIR BandPassFilter (透過域 100 - 300 kHz, $f_s = 1.17\text{MHz}$)

A.4.2 FIR フィルタの係数導出

FIR フィルタは自由な周波数特性を取ることが出来ると期待されるが、移動平均フィルタや差分フィルタと違ってフィルタ係数を決める必要がある。そのため遅延器数を一つずつ変えて各々の周波数特性を調べる、というだけではなく、各フィルタ係数を変えたときの周波数特性も調べる必要がある。FIR フィルタの伝達関数から周波数特性を求める式 A.36 のようになる。

$$\begin{aligned} |H(\omega)| &= \left| \sum_{m=0}^M h_m z^{-m} \right| \\ &= \sqrt{\left(\sum_{m=0}^M h_m \cos(m\omega T) \right)^2 + \left(\sum_{m=0}^M h_m \sin(m\omega T) \right)^2} \end{aligned} \quad (\text{A.36})$$

この式から分かるように、遅延器の数 (M) だけでなく、各フィルタ係数 (h_m) にも自由度がある。遅延器数、フィルタ係数の組合せは無限に近いほど多くなり、とてもではないが一つずつしらみ潰しに調べるというわけにはいかない。フィルタ係数があって周波数特性が決定するのではなく、任意の周波数特性があってそれに対するフィルタ係数を決定するようにする。ここでは LPF を FIR フィルタで作成するまでの方法を例として説明する。

理想 LPF は図 A.21 となる。遮断周波数 ω_c 以下の周波数では振幅比が 1 の通過域となり、 ω_c 以上、ナイキスト周波数 ($\omega_s/2$) までは振幅比 0 の除去域となる。 $G(\exp(i\omega T))$ は始めに与える理想的周波数特性として定義している。伝達関数 $H(\omega)$ と同じ意味ではあるが、 $H(\omega)$ はフィルタがあって、フィルタ係数、遅延器から求めた結果としての周波数特性として区別している。

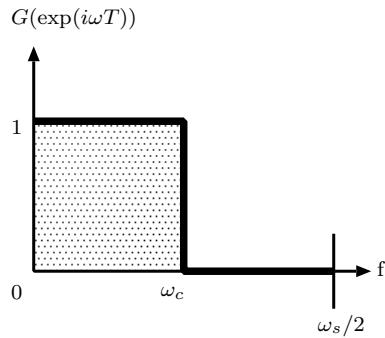


図 A.21: 理想的な Low Pass Filter

この周波数特性を式にすると以下の式 A.37 になる。

$$G(e^{i\omega T}) = \begin{cases} 1 & |\omega| < \omega_c \\ 0 & \omega_c < |\omega| < \omega_s/2 \end{cases} \quad (\text{A.37})$$

ここで FIR フィルタの周波数特性とフィルタ係数は式 A.33 を満たす。この式は離散フーリエ変換の拡張である Z 変換 A.8 の式と良く似ているので、式 A.33 も Z 変換同様、フーリエ変換式とみなすことができる。理想フィルタは無限個の遅延器を持ち、さらに過去から未来までの全ての信号に対してフィルタ係数が決まっているとする ($n = -\infty, \dots, -2, -1, 0, 1, 2, \dots, +\infty$) とこの理想 LPF と FIR フィルタ係数の関係は式となる。

$$G(e^{i\omega T}) = \sum_{n=-\infty}^{\infty} g_n z^{-n} \quad (\text{A.38})$$

g_n は $G(e^{i\omega T})$ を再現するための、無限個の遅延器が存在する場合のフィルタ係数とする。この式で $z = \exp(i\omega T)$ とし、フーリエ変換することで周波数特性 $G(e^{i\omega T})$ からフィルタ係数を求めることが出来る。フィルタ係数が連続量であるとするとそのフーリエ変換結果は式 A.39 になる。

$$g_n = \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(e^{i\omega T}) \exp(in\omega T) d\omega \quad (\text{A.39})$$

$$n = -\infty, \dots, -1, 0, 1, \dots, +\infty$$

ここまででは全ての FIR フィルタで共通の式である。この式 A.39 に LPF の理想となる周波数特性式 A.37 を適応させる。式 A.37 は偶関数と仮定している。

$$\begin{aligned}
 g_n &= \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(e^{i\omega T}) \exp(in\omega T) d\omega \\
 &= \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(e^{i\omega T}) (\cos(n\omega T) - i \sin(n\omega T)) d\omega \\
 &= \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(e^{i\omega T}) \cos(n\omega T) d\omega - \frac{i}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(e^{i\omega T}) \sin(n\omega T) d\omega \\
 &= \frac{2}{\omega_s} \int_0^{\omega_s/2} G(e^{i\omega T}) \cos(n\omega T) d\omega - 0 \\
 &= \frac{1}{n\pi} \sin\left(\frac{2n\pi\omega_c}{\omega_s}\right)
 \end{aligned} \tag{A.40}$$

$$n = -\infty, \dots, -1, 0, 1, \dots, +\infty$$

この式 A.40 の結果から分かる通り、図 A.21 を完全に再現するには無限個の遅延器が必要になるため現実的には作成できない。しかし、式 A.40 から $|n|$ が増えていくにつれ、そのフィルタ係数 g_n は 0 に近付き、全体で最も大きい係数値と比較すると無視できる値になる。そこである領域幅 ($|n| < L$) をもってフィルタの係数を暫定的に決めるところにする。この領域で区切られたフィルタ係数を \tilde{g}_n とする。領域幅で区切ることで現実的なソフトウェア、ハードウェアに実装することができるようになるが、その切断部分の不連続さからフィルタの振幅特性にリップルが生じてしまう。そこで、そのリップルを少しでも減少するようするため、窓関数 (window function) というものを用いる。これはその不連続部分をなるべく滑らかにするように考えられたもので、リップルの減少に大きく貢献する。以上の話をまとめると図 A.22 のようになる。

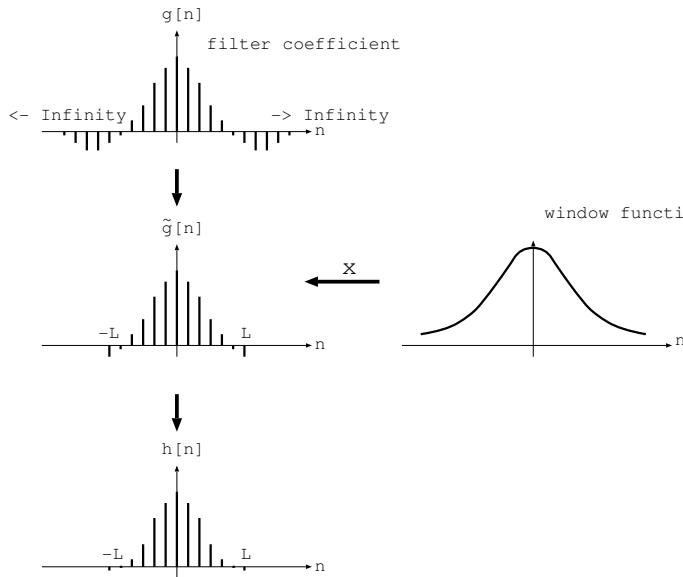


図 A.22: FIR フィルタ係数と窓関数との関係

窓関数はより理想的なフィルタを作成する上では重要であるが、窓関数そのものの多様性・複雑

さ、また APD 信号のデータ処理においては理想的なフィルタの持つ急峻な遷移域がいまのところ必要ないことを考慮して、今回は使用しないことにしている。しかし必要になったとしてもフィルタ係数が窓関数を考慮した分変化するだけなので切替えは容易である。よって今後は $h_n = \tilde{g}_n$ としてフィルタ係数を決定する。

$$h_n = \frac{1}{n\pi} \sin\left(\frac{2n\pi\omega_c}{\omega_s}\right) \quad (A.41)$$

$$n = -L, \dots, -1, 0, 1, \dots, +L$$

FIR フィルタの係数 h_n は図 A.19 のように、 $n = 0, \dots, M$ と決めている。しかし、ここで求めたフィルタ係数 h_n は n の範囲が $n = -L, \dots, 0, \dots, +L$ と $n < 0$ において係数値を持つことになる。これは出力信号が入力信号よりも先に出てきてしまうことになり、因果律を満たさない。このままではよくないのでフィルタ係数を 0 以上にシフトする必要がある。最終的にフィルタ係数は $h_n(n = 0, \dots, +2L)$ となる。

ここで遅延器数が 8 個のときと 512 個のときで LPF の周波数特性にどのような違いがあるのかを図にしてみる。フィルタ係数値はいずれも 2^{20} の精度を持つ小数のため、係数値による周波数特性の変化はないと考える。フィルタ係数を各々図にしたもののが図 A.23 である。

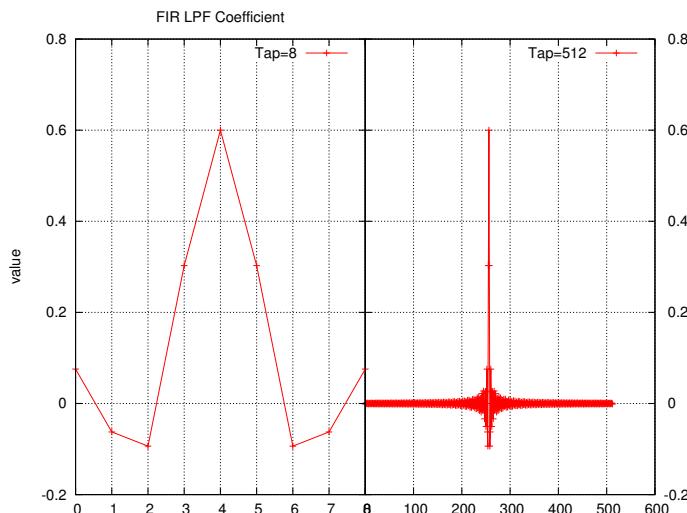


図 A.23: FIR LPF 遅延器の数を変えたときの周波数特性

遅延器数が 8 個のものは形が歪で 512 個のものは詳細なグラフになっていることがわかる。このフィルタ係数と、式 A.36 を元に周波数特性を求めたものが図 A.24 である。サンプリング周波数を 1MHz、遮断周波数を 300kHz として作成した。

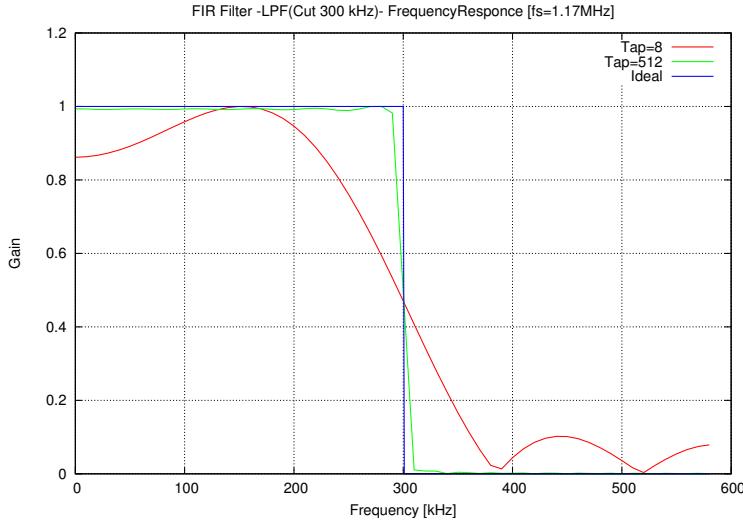


図 A.24: FIR LPF 遅延器の数を変えたときの周波数特性 ($f_s = 1.17MHz$)

赤が遅延器数 8 で緑が遅延器数 512 の場合である。また青色に理想の周波数特性ラインも載せている。どちらも LPF としては形になっているが、遅延器数が少ないものは通過域、除去域に大きなリップルが見られ、また遷移域も広い。それに対し、512 個と十分な遅延器を容易すれば理想フィルタとほぼ同じ形になっていることがわかる。

A.5 代表的なフィルタを FIR フィルタで実現

FIR フィルタはそのフィルタ係数を変更することにより、計算可能で、かつ遅延器数が充分であればいかなる周波数特性も再現できる。先ほどはその代表として LPF を再現したが、ここではフィルタの代表的な例である HPF、BPF を再現する。

A.5.1 HPF

理想的な HPF は以下のようない周波数特性を持つ。

$$G(e^{i\omega T}) = \begin{cases} 0 & |\omega| < \omega_c \\ 1 & \omega_c < |\omega| < \omega_s/2 \end{cases} \quad (\text{A.42})$$

このときフィルタ係数は式 A.43 を満たす。このときの周波数特性を図 A.25 に載せる。

$$h_n = \begin{cases} -\frac{1}{n\pi} \sin\left(\frac{2n\pi\omega_c}{\omega_s}\right) & n \neq 0 \\ 1 - \frac{2\omega_c}{\omega_s} & n = 0 \end{cases} \quad (\text{A.43})$$

$$n = -L, \dots, -1, 0, 1, \dots, +L$$

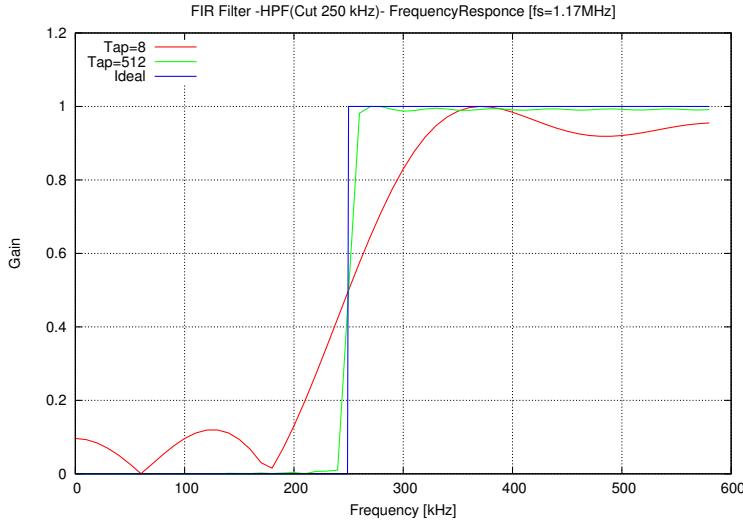


図 A.25: FIR HPF 遅延器の数を変えたときの周波数特性 ($f_s = 1.17MHz$)

A.5.2 BPF

理想的な BPF は以下のような周波数特性を持つ。

$$G(e^{i\omega T}) = \begin{cases} 0 & |\omega| < \omega_{cL} \\ 1 & \omega_{cL} < |\omega| < \omega_{cH} \\ 0 & \omega_{cH} < |\omega| < \omega_s/2 \end{cases} \quad (\text{A.44})$$

このときフィルタ係数は式 A.45 を満たす。このときの周波数特性を図 A.26 に示す。

$$\begin{aligned} h_n &= \frac{1}{\omega_s} \int_{-\omega_s/2}^{\omega_s/2} G(e^{i\omega T}) \exp(in\omega T) d\omega \\ &= \frac{1}{n\pi} \left\{ \sin\left(\frac{2n\pi\omega_{cH}}{\omega_s}\right) - \sin\left(\frac{2n\pi\omega_{cL}}{\omega_s}\right) \right\} \\ n &= -L, \dots, -1, 0, 1, \dots, +L \end{aligned} \quad (\text{A.45})$$

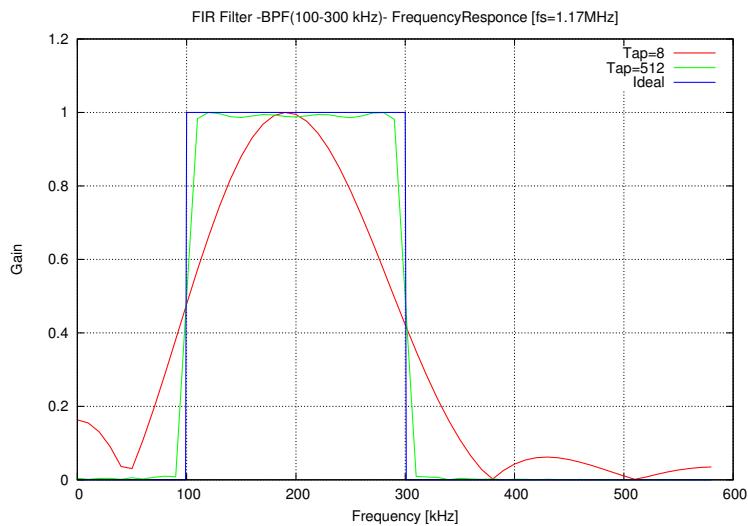


図 A.26: FIR BPF 遅延器の数を変えたときの周波数特性 ($f_s = 1.17MHz$)

付 錄B ADC128S102 評価用基板設計・製作

B.1 ADC128S102 評価用基板回路アナログ部

B.1.1 アナログ部回路設計

ADC128S102 評価用基板は基本的には図 B.1 にある素子製造元の National Semiconductor(NS) 社による典型的な回路を参考に製作している。

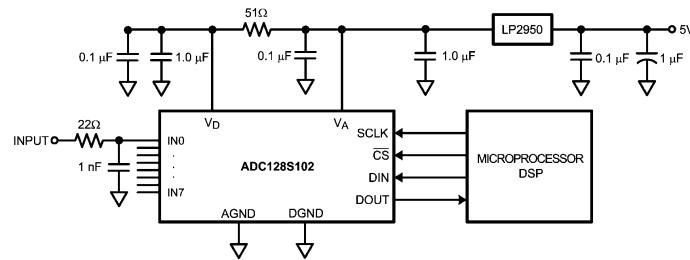


図 B.1: NS 社による ADC128S102 典型回路 [29]

製作する上でこの図 B.1 に加え、インピーダンス不整合による反射などを考慮し、さらに負の信号を出力する Pre-Amp 信号を直接入力できるように反転アンプも追加した。これによって外部で反転器を用いる必要がなくなる。これをまとめたアナログ部の回路が図 B.2 である。

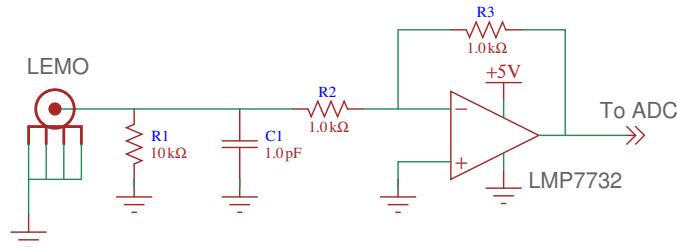


図 B.2: ADC 評価用基板 アナログ部回路

これによるアナログ回路の波形の変化を LTSpice でシミュレートする。

B.1.2 LTspice

これまで研究で回路を扱ってきたが、そのほとんどがデジタル回路だったため設計などは比較的容易だがアナログ回路は複雑である。それはアナログ回路はデジタルのような 1 と 0 に限定される

わけではなく、抵抗とコンデンサによるフィルタ、オペアンプ回路、さらにインピーダンスなどの挙動は多岐に渡るからである。この ADC Board を作る過程でそれらを全て理解できるようになったわけではないが、この LTspice を用いれば複雑な回路を設計する上で我々の手助けになるので紹介する。

アナログ回路で難しいのは回路を組んだときにどのような出力になるのか部品を組み立ててみないと分からないところである。部品の組み立ても VHDL とは違い、半田などを使って接着するので、なかなか自由に変更するというわけにはいかない。そこでソフトウェアでその挙動を調べてみようというユーザーの希望に沿ったソフトが SPICE(Simulation Program with Integrated Circuit Emphasis) である。1973 年にカリフォルニア大学バークレー校で開発された。抵抗、コンデンサといった受動素子やダイオードやトランジスタといった能動素子、配線、電源などを組み合わせた回路に対し、波形変化、周波数特性などの変化をシミュレートすることができる。

SPICE は派生されたものが多く、ユーザは色々選択することができるが、そのほとんどが有償である。今回は無償ながら高性能機能を有する Linear Technology 社開発の LTspice を使用することにした。LTspice は LT 社の製品はほぼ全て網羅し、他社の製品であっても SPICE マクロモデルがあればそれを追加することでシミュレートが可能である。図 B.3 が LTSpice によるシミュレーション画面である。下部に描画しているアナログ回路を作成した場合の周波数応答が左上、波形の変化が右上である。緑が入力電圧で赤が出力電圧となっている。下部回路は LPF とオフセット回路を組み合わせたもので、周波数特性や波形の変化を見るとわかるように回路が期待されたように動いていることが分かる。

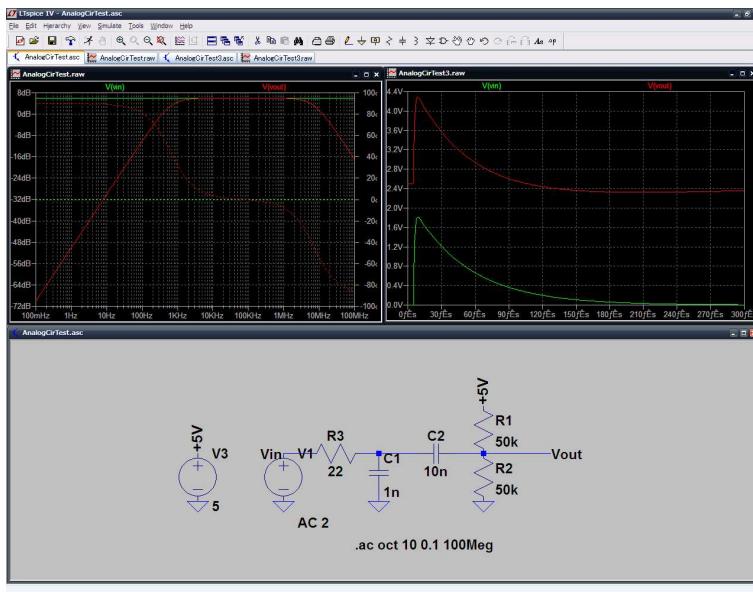


図 B.3: LTspice による ADC Board アナログ回路シミュレーション

LTspice はあくまでシミュレータなので現実はこのようになるとは限らない。しかし、シミュレーションが最高の結果であると考えれば、この段階で出力結果がおかしい回路は実機でも悪くなるという目安にはなると思われる。このソフトにより、今まででは計算でしか求めることが出来なかったアナログ回路の挙動が知りやすくなったといえる。

B.1.3 LTspiceによるアナログ回路シミュレート

アナログ回路の波形変化を LTspice でシミュレートすることによって図 B.4 のようになり、期待されているように信号が反転されていることが分かる。

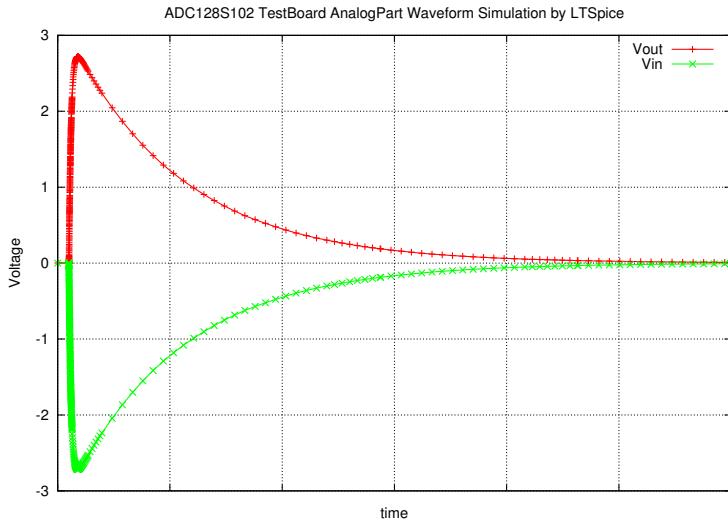


図 B.4: ADC Board アナログ部波形シミュレート (緑:Vin、赤:Vout)

反転アンプの抵抗(図 B.2 の R2 と R3)はその値によって過大入力時の出力波形が変わるものも確認している。抵抗値毎の出力波形を図B.5に記載している。これを見る限りでは $1\text{ k}\Omega$ では期待されているように OP アンプの電源電圧で出力信号が頭打ちされ、入力電圧が低くなると出力電圧もそれに従って低くなっていくが、 $100\text{ k}\Omega$ では細かな発振が発生し、 $10\text{ M}\Omega$ にもなると周期的なピークが発生する。また低い抵抗値の場合だと形が全く期待に沿わないものになってしまふことが分かる。

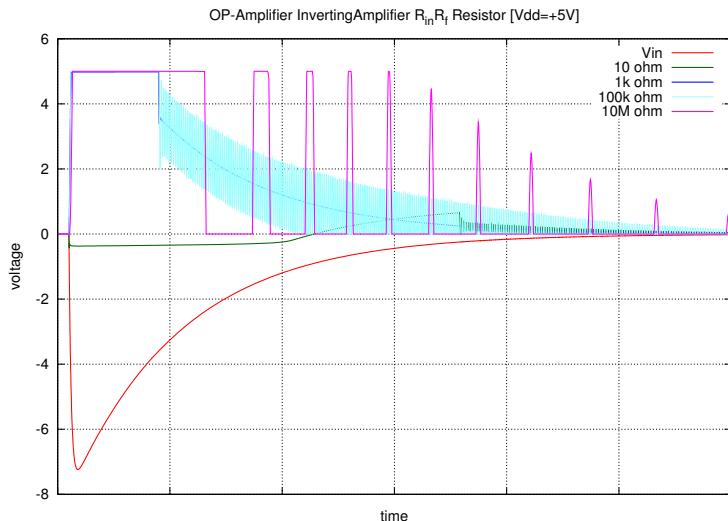


図 B.5: 過大入力時における反転アンプの抵抗別波形

このシミュレーションにより、反転アンプの抵抗を $1\text{ k}\Omega$ に決定した。またこの前段にあるコンデンサで LPF になっているのでその周波数特性もシミュレートしてみた。その結果が図 B.6 である。図からは大体 LPF になっていると思われる。1 MHz サンプリングで駆動させるつもりなので 1 MHz 以内において周波数成分がカットされていると困るがこの図を見る限りでは問題がないといえる。

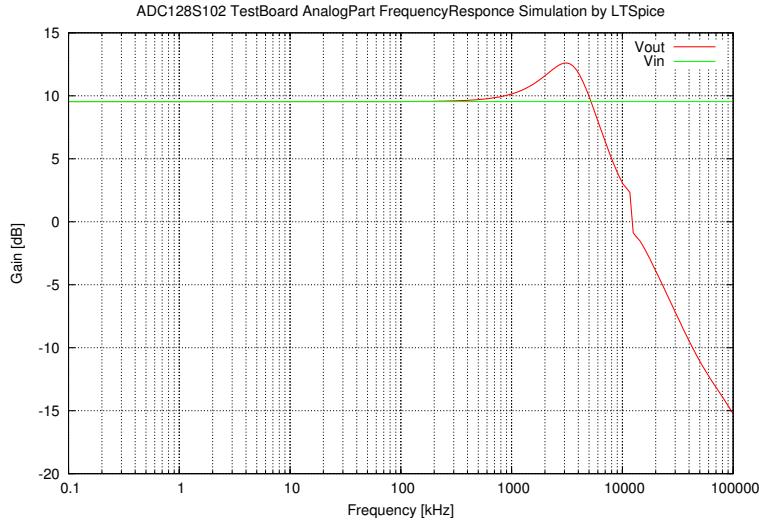


図 B.6: ADC Board アナログ部周波数特性シミュレート

B.2 ADC128S102評価用基板回路 デジタル部

この評価用基板を製作する前に一度別の評価用基板を作ったことがあるが、それは回路のデジタル部を全く意識していなかったため、反射が大きく影響してデジタル処理がほとんど行えなず、信号処理が全く行えなかった。そこで今回はインピーダンス整合や信号の緩衝を含め、Buffer を設置することにした。Buffer は東芝社の TC74LCX541F で、FADC Board や DIO Board の CMOS コネクタ部に採用されているものと同型器である。そのためデジタル部はほぼ全て FlashADC Board の CMOS 入出力回路を参考にした。図 B.7 がシマフジ電機による FADC Board の CMOS 入出力回路図である。Buffer は入力インピーダンスが高く、出力インピーダンスが低いため、それに整合するよう入力には高い抵抗を並列に、出力には低い抵抗を直列に接続している。

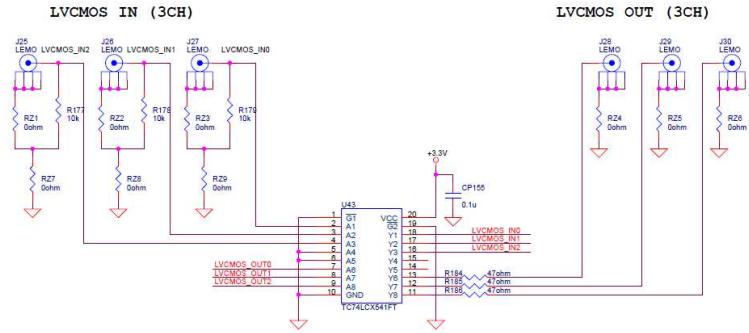


図 B.7: ADC Board デジタル部参考 FADC CMOS(In/Out) [28]

ADC128S102 評価基板回路図

それらを全てまとめた回路図が第四章の図 4.48 である。さらにこの回路を元に Eagle で配置配線をシミュレートし、さらに一点アースや GND の取り方などに注意して完成させた PCBE 図が図 B.8 である。

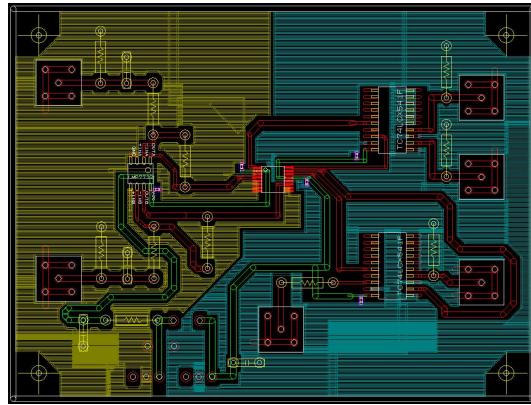


図 B.8: ADC128S102 評価基板 PCBE

左の黄土色がアナログ GND で右の藍色はディジタル GND である。両者は切り離されていて各電源 GND と繋がっている。電源端子にはなるべく近い位置にパスコンを配置し、素子の下に線が通ってしまってはいるが、信号線、電源線が切れないように注意して配置している。Buffer の入力で使用していないピンはディジタル GND に、出力で使用していないピンは open にし、さらに ADC チップの入力で使用していないピンはアナログ GND に落として他のピンへ影響がでないようになっている。この PCBE 図から版下印刷を行い、エッチングを行って評価基板を完成させた。

謝辞

本論文を執筆するにあたり、多くの方に大変お世話になりました。指導教官の深沢先生には学部4年から研究生活においてとてもお世話になり、修士2年の自転車による日本一周のため休学を申請した際も突然だったのにも係わらず心良く承諾していただき、心おきなく旅に行くことができました。ありがとうございます。飲み会に参加されたときは誰よりも話を聞き、誰よりも人の話に笑っている印象です。一緒に飲んでて楽しかったのでは非より多く参加してもらいたいと思っております。大杉先生には修士の頃はなかなか話をする機会もありませんでしたがコロキウムのときや学部4年の半導体セミナーでは半導体の基礎を教えていただきありがとうございます。ひろたかさんには公私ともにとてもお世話になりました。実験では結果が芳しくないときも夜遅くまで一緒に原因を究明してくださったり、実験の手法なども教えてくださったりと本当に自分のためにもなる研究でした。プライベートでは、富士山では私が救急車搬送騒ぎになったときも冷静に対処していただきました。折角の砂走りが私をおぶる形となってしまい、申し訳ありません。屋久島では運転できない私の代わりに最初から最後まで運転してくださりありがとうございます(車中、私は気が付いたら寝ていたりして本当に申し訳ありません)。また白谷雲水峠や海中温泉に行きたいですね。香川にうどんを食べに行ったときも運転をしてくださりありがとうございます。よもや父親にひろたかさんを紹介することになるとは思いもしませんでした。また香川に来るときは是非連絡をお願いいたします。PoGOLite が放球するときにはスウェーデンには行けませんが無事成功することを祈っております。水野先生には実験の説明や宇宙における物理的な解釈について懇切丁寧に教えていただきました。お互いが納得するまで理由を説明してくださり、自分で考えるよりも理解が深まりました。ありがとうございます。毎16時頃に小部屋にやきそばの匂いをまき散らして申し訳ありません。作るのが楽なのです。

今は卒業されました田中(琢)さんにも感謝を表したいと思います。田中さんには SpaceWire の基礎を始め、DAQ の基礎、VHDL/C++ の基礎を教えていただきました。今でも田中さんの作成したコードを参考してプログラムを書いています。また田中さんの御卒業後も一緒に秋葉原に行ったり、富士山へ流星群を観に行ったり、またメッセンジャーでチャットをしたりと大変楽しく大学院生活を送れました。今後も東京に行くときはまたよろしくお願ひいたします。あとディズニーランドに行ってきてください。行くなら案内します。宇宙研の大野さんには大野さんの広大在学中である私の学部時代よりも大学院時代に大変お世話になりました。SpaceWire やディジタルフィルタにおいて一緒に研究するのがとても楽しかったです。牛神で焼肉や、小倉でふぐを食べさせてもらったりしたのは感激でした。しかし小倉で私がひろたかさんに絡まれて救難信号を出していたのに助けてくれなかったのが残念です。あれはきっと面白がっていたに違いと思います。永江さん、違和感がありすぎるので以下おさむさん、にはスキーや飲み会、また塾などプライベートなどお世話になりました。2010年末に聖光学院の教師として訪問された際、生徒の鏡としてのおさむさんはカッコよかったです。高橋(拓)さんにはパソコンの知識に関してとてもお世話になりました。パソコンのトラブルは高橋さんに、という図式が出来上がっていた気がしないでもありません。趣味でも気が合い、お茶部屋に居たときは話がはずんで楽しかったです。安田さんには音楽で気が合いました。菅野よう子作の Tank! を教えてくれたときはとても嬉しかったです。Tank! がきっかけで Cow-Boy BeBop を全部見ました。山中さんにはお茶部屋・小部屋とずっと同じ部屋

だったのにも係わらず実験室に籠ってばかりの私にいつも明るく話題を振りまいてくださり、ありがとうございます。学会や夏の学校への参加などのアクティビティ溢れるその姿勢は見習いたいと思います。上原さんはいつも小部屋のムードメーカーで、楽しく小部屋生活を送ることができました。研究もとても真面目になさっているのにイライラすることは無いんだろうかと疑問に思うあまりに、実は研究は真面目ではないのでは?と軽く考えるほどです。またスキーに連れて行ってください。卒業後も GRB(Burst と Bank) 関係での指導などお願ひいたします。西野さんは半導体関係でお世話になりました。夜に大部屋でとりとめもない話をするのが楽しかったです。4年(次のM1)である数馬君や後藤君をいじめないで(いじって)くださいね。画伯復活してください。宮本さんは研究では本当に全くお世話になっていませんが、プライベートでの付き合いは大変お世話になりました。私が出水市にいると言えば鹿屋市から自動車で 100 km ほど駆けつけてくれたあげく豚カツ定食まで奢ってください、また鹿児島市では黒豚しゃぶしゃぶを奢ってくださいました。東京に移った後もコミケに一緒に行ったり翌日出勤なのに家に泊めてくださったりとお世話になりました。またよろしくお願ひします。吉田(広)さんには学部4年での最初の Linux の使い方から実験の指導などお世話になりました。御卒業されてからなかなか会えないのが残念です。東京にいる方はまた東京で会えることを期待しています。

同期の池尻君には学部4年時には一緒に実験を行い、お互い励まし合いました。M2 では私が休学したために同期5人中、卒業したのは池尻君ただ一人となって申し訳ないです。東京に行ってからはあまり会うことがないため残念ですが、また飲み会でもしましょう。誰にでも明るく、ノリがいいその姿勢は今後も見習いたいと思います。扇君は学部4年での付き合いだけですが、池尻君と並んで16のムードメーカーでした。自転車旅行において名古屋で泊まらせてもらったときは家族のみなさんにもてなしていただき、ありがとうございます。先本君は修士からの付き合いですが、なかなか会話する機会がなかったのが残念です。後輩に対して会話を心がけるその優しさをこれからも続けてください。花畠君は HXI/SGD 実験とともに APD 処理部の研究を行い、お互いの良い刺激になりました。BGO+APD 信号のセットアップも花畠君の力あってのことだと思っています。また意外とノリが良いのがびっくりでした。もっとそれを前面に出すべきです。林君は優しすぎて何を書けばいいのか分かりません。研究が趣味みたいなものだと思われるため苦では無いと思うのですが、たまにはゆっくり遊んだりしてください。マグネットありがとうございます。林君もヨーロッパに行くときは買い集めてみるのをお薦めします。単純な疑問から複雑なものまで心良く答えてくれて感謝しています。平木君は SGD チームとしてお互い拙差琢磨してきました。去年に提出したはずの修論が今年の1月に完遂したと言ったときはどゆこと?と一瞬考えてしまいました。今後の博士研究を楽しみにしています。

後輩の伊藤君はシェルスクリプトを教えてもらったり、もはや後輩とは思えない程知識も豊富で羨ましいです。博士に行っても頑張ってください。小松君は癪しキャラで研究室を盛り上げていましたが来年から朝起きられるのかどうかが心配です。佐田君はいつも楽しく飲み会を盛り上げてくれてありがとうございます。明るく小部屋を盛り上げてくれていたと思います。道津君は面白目すぎるので気が疲れないのか心配です。いつも笑顔な道津君のどこにそんなパワーがあるのか不思議です。本田君も佐田君と一緒に小部屋や研究室を盛り上げてくれてありがとうございます。東京に行くときは是非会いましょう。後藤君は次期 SpaceWire ユーザとして、とてもとっても期待しています。基本はすでに習ったのであとはコードを書いて書いて書き続けてください。深沢先生やひろたかさんの指示に従えば大丈夫です。私の卒論時よりも高度な内容をした卒論なので2年後の修士論文はそ

れはもう、素晴らしいものが出来ていると思います。今後に期待しています。

思い出話となってしましましたが、ここには私の持つ4年間の感謝の言葉を書くにはスペースが足りません。しかしその一部でも感じていただけたら幸いです。研究生活4年において様々な人と出会い、様々なことを教えていただきました。これらは次の社会人生活に直接影響を及ぼすことはないやもしれません。ですがみなさんとの出会い、ここでの研究へのスタンスを自分なりに昇華し、次のステップへと進むための糧としたいと思います。本当にみなさまありがとうございますございました。今後共よろしくお願ひいたします。

参考文献

- [1] 高橋忠幸 笠羽康正 高島健 吉光徹雄 山田隆弘 能町正治 「科学衛星データ処理系の将来展望」
- [2] 田中琢也「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」, 広島大学, 卒業論文, 2006
- [3] 田中 琢也「衛星搭載機器統一通信規格 SpaceWire を用いた宇宙 X 線・ガンマ線観測用データ収集システムの開発」, 広島大学大学院, 修士論文, 2008
- [4] 吉田 広明「気球搭載宇宙硬 X 線偏光検出器 PoGOLite の主検出部の地上試験による性能実証」, 広島大学大学院, 修士論文, 2009
- [5] 上田剛「次期宇宙 X 線衛星 ASTRO-H に搭載する硬 X 線・軟ガンマ線検出器の機構開発」, 東京大学大学院, 修士論文, 2009
- [6] 花畠 義隆「X 線天文衛星 ASTRO-H 搭載アクティブシールドの基礎特性評価とデジタル信号処理の検討」, 広島大学大学院, 修士論文, 2010
- [7] 平木 一至「モンテカルロシミュレーションによる衛星搭載軟 γ 線検出器の軌道上放射化バッカグラウンドの評価」, 広島大学大学院, 修士論文, 2010
- [8] 米谷 光生「宇宙 X 線観測における中性子バックグラウンドモニターのためのシンチレータの基礎特性実験」, 広島大学, 卒業論文, 2010
- [9] Hirokazu Odaka Takayuki Yuasa 「SpaceWire のつなげかた TRON 版 SpaceCube 編」
- [10] ECSS, Space engineering ECSS-E-50-12A, 2003
- [11] Xilinx 社, Spartan-3 ジェネレーション FPGA ユーザーガイド (UG331 v1.2), 2007
- [12] Personal Media Corporation, Teacube/VR5701 評価キット取扱説明書 1.A0.01, 2004
- [13] 'SpW-10X SpaceWire Router User Manual' Issue 3.3 ,30 April 2008
- [14] 長谷川裕恭「VHDL によるハードウェア設計入門」 CQ 出版
- [15] 三上直樹「はじめて学ぶ ディジタル・フィルタと高速フーリエ変換」 CQ 出版
- [16] T. Mizuno et al. 'High Sensitivity Balloon-Borne Hard X-Ray/Soft Gamma-ray Polarimeter PoGOLite' 2007 IEEE Nuclear Science Symposium Conference Record
- [17] H. Takahashi et al. 'Data Acquisition system of the PoGOLite Balloon Experiment' 2008, International SpaceWire Conference 2008

- [18] T. Kamae et al., 'PoGOLite - A Hight Sensitivity Balloon-Borne Soft Gamma-ray Polarimeter', Astroparticle Physics 30 (2008) 72 arXiv:0709.1278v2
- [19] M.Pearce , 'PoGOLite - a high sensitivity balloon-borne soft gamma-ray polarimeter', 2008 IEEE Nuclear Science Symposium
- [20] H. Takahashi et al., 'A Thermal-Neutron Detector with a Phoswich System of LiCaAlF6 and BGO Crystal Scintillators Onboard PoGOLite', 2010 IEEE Nuclear Science Symposium Conference record
- [21] 次期 X 線天文衛星計画ワーキンググループ 「NeXT 計画提案書」,2005
- [22] M.Kokubun et al., 'In-Orbit Performance of the Hard X-ray Detector on board Suzaku',2006 PASJ:Publ. Astron. Soc. Japan
- [23] M.Kokubun et al., 'Hard X-ray Imager (HXI) for the ASTRO-H mission',2010 SPIE2010
- [24] H.Tajima et al.,'Soft Gamma-ray Detector for the ASTRO-H Mission',2010 arXiv:1010.4997v1
- [25] ASTRO-H プロジェクトサイト,'<http://astro-h.isas.jaxa.jp/>'
- [26] シマフジ電機株式会社, SPW Digital I/O ボード (双向拡張バス、uart 機能付き) 取り扱い説明書 ver 0.3, 2007
- [27] シマフジ電機株式会社, SPW Sampling ADC ボード (双向拡張バス) 取り扱い説明書 ver 0.21, 2008
- [28] シマフジ電機株式会社, FADC Board 回路図
- [29] National Semiconductor 社 ADC128S102 Manual
- [30] Toshiba,TC74LCX541F Data sheet,1997
- [31] サンハヤト社 「『ポジ感光基板』を使ったオリジナル基板製作の手順」
- [32] 気象庁データベース
- [33] XCOM:Element/Compound/Mixture Selection, '<http://physics.nist.gov/PhysRefData/Xcom/html/xcom1.html>'