

# 修士論文

CMB 観測実験 GroundBIRD の焦点面検出器アライメントと  
長期運用に向けた角度データ取得システムの最適化

京都大学 理学研究科 物理学・宇宙物理学専攻  
物理学第二教室 高エネルギー物理学研究室

片岡 敬涼

2024年1月24日



## 概要

gaiyou(CMB で迫りたい物理 (ニュートリノ質量和、tau)、GB、自分のやったことがどう重要なのかを簡潔に記す)

本論文の構成を述べる。第 1 章で CMB に関する理論的な背景、第 2 章で GroundBIRD 実験の概要を説明する。以降は第 3 章と第 4 章の 2 部構成になっており、第 3 章で GroundBIRD の角度データ取得システムの改善について、第 4 章で焦点面検出器のアライメント較正との結果について述べる。第 5 章で今後の展望を述べ、第 6 章でまとめを述べる。

# 目 次

<b>第 1 章 序論</b>	<b>1</b>
1.1 CMB の異方性と現代宇宙論	1
1.1.1 CMB の温度異方性	1
1.1.2 $\Lambda$ -CDM モデル	1
1.1.3 地平線問題	1
1.2 CMB の偏光とインフレーション理論	1
1.2.1 インフレーション理論	1
1.2.2 CMB の偏光モード	1
1.2.3 偏光 B モードの探索状況	1
1.3 CMB の偏光とニュートリノ質量和	1
1.3.1 光学的厚み $\tau$	1
1.3.2 ニュートリノ質量和との縮退	1
1.3.3 偏光 E モードと $\tau$	1
<b>第 2 章 GroundBIRD 実験</b>	<b>2</b>
2.1 実験概要	2
2.1.1 GroundBIRD 望遠鏡とスキャン戦略	2
2.1.2 超伝導検出器 MKID	3
2.1.3 観測する周波数帯域	5
2.1.4 物理ターゲット	6
2.2 現在の観測状況	7
2.2.1 検出器のフルアレイインストール	7
2.2.2 リモート観測システム	9
<b>第 3 章 仰角データ取得システムの改善</b>	<b>10</b>
3.1 望遠鏡仰角データ取得システムの改善	10
3.1.1 角度情報データ取得の概要	10
3.1.2 仰角データ取得における問題点	13
3.1.3 PYNQ を用いた新システムの導入	16
3.1.4 PYNQ イメージファイルの作成	17
3.2 望遠鏡への実装	18
3.2.1 新データ取得システムのインストール	18
3.2.2 同期信号取得の確認	20
3.2.3 同期信号の分配と仰角データ取得の確認	20

3.3 メンテナンスと安定運用	21
3.3.1 動作の不安定性	21
3.3.2 電源供給方法の見直し	22
3.3.3 Zynq 溫度のモニター	24
<b>第4章 焦点面検出器アライメントの較正</b>	<b>26</b>
4.1 検出器アライメントの問題点	26
4.1.1 スキャン軸に対する傾きと差分解析	26
4.1.2 要求される理想的なアライメント	27
4.1.3 視線軸方向まわりの回転による較正	28
4.2 月を用いた回転角の算出	28
4.2.1 月を用いた理由	28
4.2.2 位相としての検出器 TOD	30
4.2.3 必要な回転角	31
4.2.4 回転する上でのジグの必要性	38
4.3 ジグの設計と現地インストール	38
4.3.1 固定用ジグの作成	38
4.3.2 望遠鏡への実装	38
4.4 天体を用いた較正結果の確認	38
4.4.1 月データによる確認	38
4.4.2 木星データによる確認	38
4.5 検出器間差分で見る大気揺らぎの抑制	38
4.5.1 timing offset の算出	38
4.5.2 差分解析による確認	38
<b>第5章 今後の展望</b>	<b>39</b>
5.1 大気揺らぎに由来するノイズのモデリング	39
5.2 両偏波アンテナを搭載した焦点面検出器のアップデート	39
<b>第6章 まとめ</b>	<b>40</b>
<b>第7章 謝辞</b>	<b>41</b>
<b>参考文献</b>	<b>42</b>
<b>付録A Zynqへのlinux搭載</b>	<b>44</b>

# 第1章 序論

## 1.1 CMB の異方性と現代宇宙論

### 1.1.1 CMB の温度異方性

### 1.1.2 $\Lambda$ -CDM モデル

### 1.1.3 地平線問題

## 1.2 CMB の偏光とインフレーション理論

### 1.2.1 インフレーション理論

### 1.2.2 CMB の偏光モード

### 1.2.3 偏光 B モードの探索状況

## 1.3 CMB の偏光とニュートリノ質量和

### 1.3.1 光学的厚み $\tau$

### 1.3.2 ニュートリノ質量和との縮退

### 1.3.3 偏光 E モードと $\tau$

## 第2章 GroundBIRD 実験

CMB 観測実験には地上から観測する実験と衛星を用いて宇宙から観測する実験に分けられる。ここでは私が参加している GroundBIRD 実験(図 2.1)について実験の概要と現在の観測状況について説明する。



図 2.1: GroundBIRD 望遠鏡の外観。望遠鏡クライオスタットが方位角回転台の上に設置されており、回転台とともに最大で 20 RPM(1 分間で 20 回転) の速度で回転する。

### 2.1 実験概要

#### 2.1.1 GroundBIRD 望遠鏡とスキャン戦略

GroundBIRD 望遠鏡はスペイン領カナリア諸島の 1 つであるテネリフェ島のティデ観測所(高度 2,400 m)に位置する地上 CMB 望遠鏡である。地上からの観測において最も邪魔なのが大気からの放射であるが、ティデ観測所は大気中の積算水蒸気量(Precipitable Water Vapor、以下 PWV と略す)がおよそ 3.5 mm<sup>[1]</sup> と、観測に適した場所である。

GroundBIRD はスキャン戦略に大きな特徴を持つ。地上からの観測では大気放射に由来するノイズが本来見たい CMB に混入する。大気放射は無偏光であるが、観測装置の不完全

性などで誤って偽偏光として観測されるおそれがある。特に、大気は刻一刻と揺らいでいるため、観測する空の領域ごとで観測される大気のノイズも揺らぎ、偽偏光を検出する影響は無視できなくなる。その影響を回避するためには大気揺らぎを抑制する変調が必要になる。GroundBIRD では、望遠鏡を最大で 20 RPM(3 秒で 1 回転)させる独自のスキャン戦略をとることで大気揺らぎを抑制した CMB 観測を実現する。加えて望遠鏡の仰角を  $70^\circ$  に固定し、方位角方向に高速回転させることで、全天の広い領域を観測することができる。望遠鏡の連続回転と地球の自転を組み合わせることで全天の約 45 % を観測することができる(図 2.2)。



図 2.2: GroundBIRD のスキャン戦略。GroundBIRD は視野  $\pm 11^\circ$  で観測する。望遠鏡の回転と地球の自転を組み合わせることで 1 日で全天の約半分をカバーできる。

次に、GroundBIRD の内部構造の概略を図 2.3 に示す。光学系は放物面の主鏡と双曲面の副鏡から成り、CMB がバッフルから入り、光学系で 2 回反射させた後に、焦点面検出器ステージに入る。クライオスタット内は真空中かつ低温になっており、外側のチャンバー部(300 K)、40 K シールド、4 K シールドの 3 層から構成されている。4 K シールドの冷却にはパルスチューブ冷凍機を使用している。GroundBIRD では超伝導検出器 “MKID(Microwave Kinetic Inductance Detector)” を採用しているため、焦点面の温度は極低温に保つ必要がある。焦点面の冷却には He ソープション冷凍機を使用し、温度を 280 mK 付近に保持している。

### 2.1.2 超伝導検出器 MKID

CMB 観測実験で使われる超伝導検出器として MKID と “TES(Transition Edge Sensor)” ポロメータがある。MKID は比較的新しい検出器であり、通常の CMB 実験では TES が使われている。GroundBIRD では高速スキャンのもとで角度分解能を失わないようにサンプリングレートを 1 kHz にしているため、検出器の応答時間が  $< \mathcal{O}(1)$  ms であることが要求される。しかし、TES の応答時間は  $\mathcal{O}(1)$  ms であり [2]、GroundBIRD の要求を満たしていない。一方で MKID の典型的な応答時間は  $< \mathcal{O}(1)$  ms であり [3]、この要求を満たしている。そのため、GroundBIRD ではより時間応答性の良い MKID を採用している。



図 2.3: GroundBIRD 内部の概略図。バッフルを通って CMB が望遠鏡内に入り、主鏡と副鏡で反射されて検出器ステージに入る。

MKID の動作原理の概要を説明する。MKID は超伝導共振回路を応用した高感度な光検出器である。入射する光子のエネルギーに応じて変化する回路内のインダクタンスを、数 GHz で読み出す。MKID の電子顕微鏡写真 [4] と、等価回路を図 2.4 に示す。MKID は読み出し線、超伝導体からなる共振器回路、アンテナからなっている。アンテナから電磁波が入射すると超伝導共振器の状態が変化し、そのインピーダンスの変化を読み出すことで入射エネルギーを測定する。

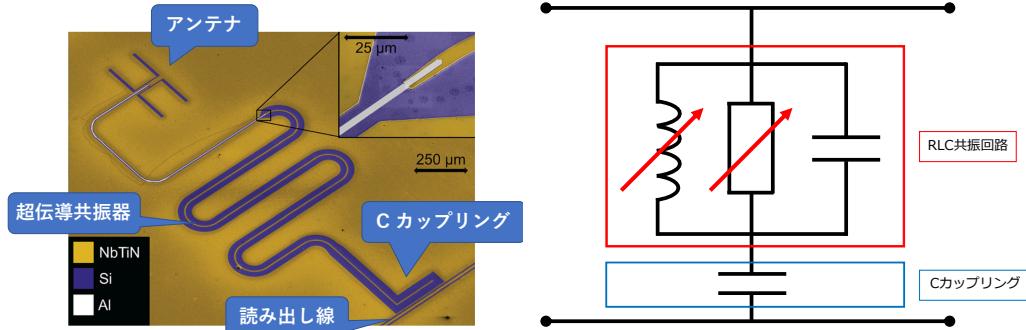
具体的には、検出器の温度上昇やエネルギーが  $h\nu > 2\Delta$  ( $\Delta$  は超伝導ギャップエネルギー) の光子との反応で、超伝導共振器内のクーパー対 (結合した電子対) が壊れる。対になっていた電子はエネルギーギャップより上の準位へと押し上げられる (図 2.5)。この過程で生成される電子を準粒子という。 $\Delta$  と超伝導転移温度 ( $T_c$ ) の関係は  $T=0\text{ K}$  の時に

$$2\Delta(T=0) = 3.52k_B T_c \quad (2.1)$$

と表すことができる [5]。ここで、 $k_B$  はボルツマン定数である。この式より、MKID の材質として用いる超伝導体の転移温度と  $\Delta$  は 1 対 1 対応しており、転移温度を適切に設定することで CMB のエネルギー ( $\sim 160\text{ GHz}$ ) でクーパー対を壊すことができる。例えば、アルミニウムであれば転移温度は  $1.2\text{ K}$  であり、対応する光子の必要エネルギーは  $90\text{ GHz}$  になるため、CMB の検出が可能になる。また、MKID では入射信号によって生成される準粒子の数に比例した応答が得られるため、ギャップエネルギーに近いエネルギーの CMB に対して高感度な検出器になる。

準粒子によって共振器内の超伝導状態が変化し、可変インダクタンスの値が変化する。また、1 つの読み出し線に複数の共振器が容量性カップリング (Capacitive coupling; C カップリング) しており、1 対の読み出し配線を使って  $\mathcal{O}(1000)$  個の MKID を同時に読み出すこと

ができる。



(a) MKID の電子顕微鏡写真 [4]。読み出し線、(b) MKID の等価回路。可変インダクタンス  
超伝導共振器、アンテナからなる。  
と可変抵抗をもつ RLC 共振回路になっている。  
る。

図 2.4: 超伝導検出器 MKID

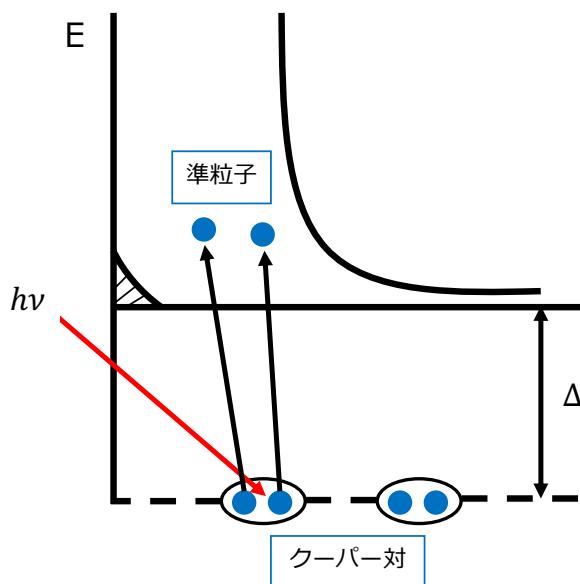


図 2.5: 入射光子による準粒子生成の模式図。縦軸は電子のエネルギーを表す。 $h\nu > 2\Delta$  の  
エネルギーを持つ光が超伝導共振器に入射すると、クーパー対が壊されてエネルギー準位が  
押し上げられ、準粒子になる。

### 2.1.3 観測する周波数帯域

CMB の偏光を観測するためには CMB と、銀河などから来る CMB と同周波数帯の放射である“前景放射”とを分離する必要がある。これらの前景放射はそれぞれ異なる周波数依存性を持つ(図 2.6)。主な前景放射には低周波側で卓越する“シンクロトロン放射”と高周波側で卓越する“ダスト熱放射”があり、CMB にとって大きなノイズとなる。そのため、低周波側  $\mathcal{O}(10)$  GHz から高周波側  $\mathcal{O}(100)$  GHz までの広い帯域での観測を行い前景放射を取り除くことが求められる。2.2.1 で述べるが、GroundBIRD は CMB に感度のある 145 GHz と、

ダスト放射に感度がある 220 GHz の 2 つの帯域で観測する。一方で、低周波側は QUIJOTE<sup>1</sup> のデータを使うことでカバーする。



図 2.6: CMB と前景放射の偏光強度を周波数の関数として表した図。Planck[10] を参照。それぞれの前景放射は異なる周波数依存性を持ち、CMB これらを分離するためには広い帯域での観測が必要になる。

#### 2.1.4 物理ターゲット

GroundBIRD が探る物理ターゲットは 1.3.3 で見た光学的厚み  $\tau$  の地上からの再測定である。光学的厚み  $\tau$  の測定は今までに WMAP や Planck といった衛星実験によって測定がされてきた。図 2.7 に測定された  $\tau$  の値の変遷を示す。誤差が小さくなってきており、最新の測定結果ではその誤差は  $\sim 10\%$  である。しかし、平均値は系統的に下がっている傾向にあり、独立した測定によってこの結果の妥当性を評価する必要がある。そのため、地上実験(例えば CLASS[7] や QUIJOTE[8] など)からの  $\tau$  の精密測定が始まっている。

特に GroundBIRD は独自のスキャン戦略を活かして  $\tau$  の値に迫ることができる。高速スキャンによって大角度スケール ( $6 < \ell < 300$ ) の CMB 偏光を測定することができる。大角度スケールと  $\tau$  の関係を図 2.8 に示す。偏光 E モードのパワースペクトルは大角度スケール ( $\geq 10^\circ$ ) で  $\tau$  に応じて異なる振る舞いをする。GroundBIRD はこの振る舞いを観測することができるため、 $\tau$  の測定に適している。3 年間の観測と、GroundBIRD と QUIJOTE の共同解析によって  $\tau$  を誤差  $\sigma_\tau \sim 0.01$  で測定することを目指す [11]。これによって疑問の残る  $\tau$  の測定を再検証することができ、縮退したニュートリノ質量和の精密測定を達成できる。

<sup>1</sup>QUIJOTE(Q-U-I JOint Tenerife Experiment) 実験は GroundBIRD から 20 m ほどしか離れていない隣に位置する望遠鏡である。2 台の望遠鏡 (11, 13, 17, 19 GHz を観測する QT1 と 30, 40 GHz を観測する QT2) で構成される。

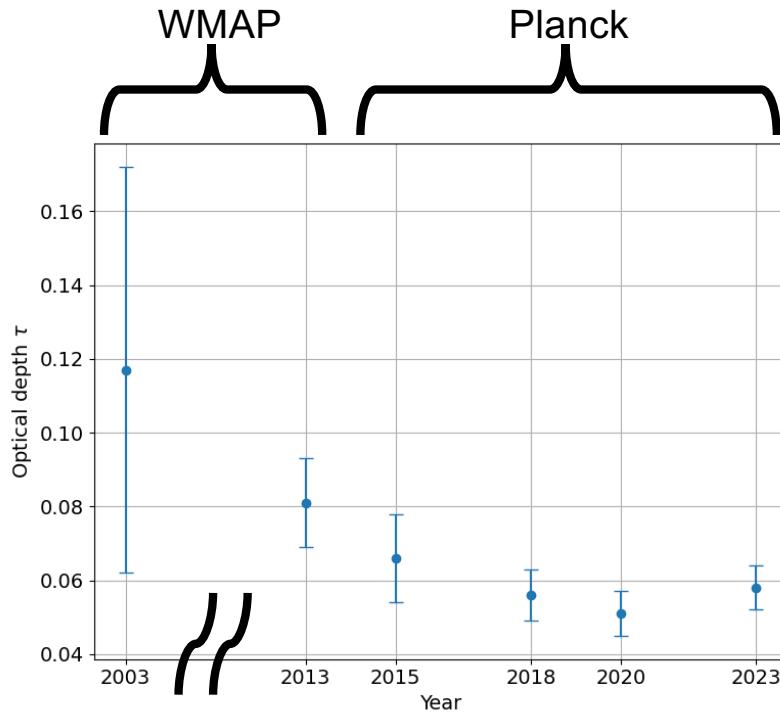


図 2.7: WMAP と Planck によって測定された光学的厚み  $\tau$  の値 [6]。最新では誤差は  $\sim 10\%$  である。

## 2.2 現在の観測状況

### 2.2.1 検出器のフルアレイインストール

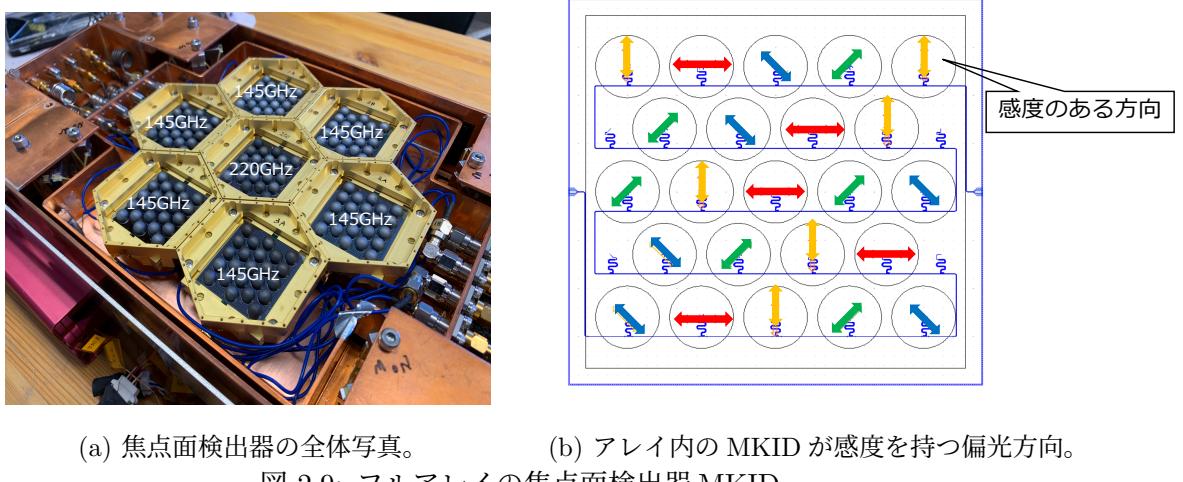
GroundBIRD の現在の状況について説明する。2022年1月から2022年5月まではプロトタイプ検出器を用いたコミッショニング観測が行われた。コミッショニングデータを用いて望遠鏡の視線方向の較正 [12] や、偏光角較正、ノイズ特性の理解 [13] などがされてきた。並行して、2023年5月に全焦点面検出器のインストールが完了し、本格的な物理観測がスタートした。

インストールした焦点面検出器を図 2.9 に示す。23 個の MKID が 1 つのアレイに搭載されており、145 GHz が 6 アレイと 220 GHz が 1 アレイの全 7 アレイからなる。中央に 220 GHz アレイがあり、その周りを 145 GHz アレイが囲むように並んでいる。アレイごとに読み出しを行う。アレイ内での MKID の配置は図 2.9b によくなっている。各 MKID は片偏波アンテナを持つため、1 方向の偏光方向に感度がある。また、偏光方向は  $45^\circ$  ずつで 4 方向あり、異なる方向に感度のある MKID 同士で交互に配置されている。これにより、第 4 章で述べるような検出器間で差分を取る解析によって偏光測定をすることができる。

最後に 145 GHz, 220 GHz のそれぞれの検出器と望遠鏡ビームの特性を表 2.1 に示す。



図 2.8: パワースペクトルの過去の観測結果と E,B モードの理論線、そして GroundBIRD の観測領域 [9] をオレンジで示す。青実線はテンソル・スカラー比を仮定した時の原始重力波由来の偏光 B モード、赤点線は重力レンズ効果由来の B モードを表す。黒実線は E モードであり、大角度スケール ( $\geq 10^\circ$ ) で  $\tau$  の値によってスペクトルに違いが生まれる。



(a) 焦平面検出器の全体写真。  
(b) アレイ内の MKID が感度を持つ偏光方向。  
図 2.9: フルアレイの焦平面検出器 MKID

表 2.1: 145 GHz、220 GHz における検出器とビームの特性 [14]

	145 GHz	220 GHz
MKID 数	6 アレイ (138 ピクセル)	1 アレイ (23 ピクセル)
ビーム幅 (FWHM)	$0.60^\circ (36')$	$0.42^\circ (25')$
ビーム梢円率	< 1 %	< 2 %

### 2.2.2 リモート観測システム

焦点面検出器のフルアレイインストール後、本格的な観測が始まっているが、基本的に全ての操作(ドームの開閉、望遠鏡の回転など)をリモートから行なっている。観測の手順は以下のようにして行う。

1. 天候等が観測に十分適しているか確認
2. 問題がなければドームを開き、望遠鏡の回転を開始
3. 検出器のデータ取得を開始し、観測状況をモニター

望遠鏡をダストや雨、直射日光から守るために観測時間以外はドームを閉めている。観測開始前には天候状況(PWV、湿度、風速等)を確認し、観測するか否かを判断する。また、日光が望遠鏡の視野に入る日中や、冷却に使用するHeのメンテナンス時間以外は基本的に観測を継続する。そのため、観測シフトを日本時間と現地時間で分けることで1日を通して観測を続けている。また、Slackアプリケーションを導入し、アプリ内で対話的に観測状況を把握できるようになっている(図2.10)。



図 2.10: Slack アプリケーションを用いたモニター。Web カメラの映像から観測状況を確認できる。他にも観測中の検出器データのチェックや、サウンドモニターによる異音検知などが可能。

検出器をフルアレイでインストールし、観測システムが整ったことで GroundBIRD 実験としては観測を続けてデータを蓄積する段階にある。そのため、求められることは

- 安定して長期的な運用と観測を続けられる
- 質の良いデータを取得する

ことである。現在の望遠鏡システムではこの条件を満たす上では不十分な点も多く、観測を続ける中でメンテナンスとアップグレードが必要である。以降の第3章ではデータ取得の長期運用のために仰角データ取得システムの改善を行い、第4章では検出器アレイとしてのデータの質を向上させるために検出器のアライメント較正を行なった。

## 第3章 仰角データ取得システムの改善

CMB 観測においては、検出器の時系列データと望遠鏡の角度データを途切れることなく連続的に取得し続けなければいけない。そのため、角度データ取得システムは安定的でかつ操作性がよいものであることが求められる。本章では既存の望遠鏡の仰角データ取得システムを改善し、その動作確認を行なった。

### 3.1 望遠鏡仰角データ取得システムの改善

#### 3.1.1 角度情報データ取得の概要

はじめに、GroundBIRD 全体でのデータ取得システムの概要を述べる。CMB 観測においては検出器の信号を時系列データ（以下、TOD と略す）として取得する。最終的なマップ作成のためには、TOD と同期して望遠鏡の視線情報（角度データ）を取得することが求められる。GroundBIRD では望遠鏡の仰角方向と方位角方向で 2 つの角度データを取得している。連続回転する回転台の上部と下部は回転継手によって電気的に接続されている（図 3.1）。



図 3.1: GroundBIRD の検出器データと角度データ取得系の概要。回転する回転台の上下での信号同期は“回転継手”が担っている。

仰角方向の角度データは、望遠鏡の側面に取り付けられたロータリーエンコーダー（Canon, R-1SL [15]）を使用し、Digilent 製の FPGA ボード Zybo Z7-20 ([16]、以下では単に Zybo と記す）で読み出す（図 3.2）。FPGA とは Field Programmable Gate Array の略で、様々な論

理回路がチップに搭載されており、使用者が配線を自由に組み合わせて論理回路を作ることができるデバイスである。FPGA では特定の演算を行う回路を作成できるため、高速処理を可能にする。また並列処理も得意である。エンコーダーは 4 秒角 ( $1.1 \cdot 10^{-3}$  °) もの高い角度分解能を持つ。

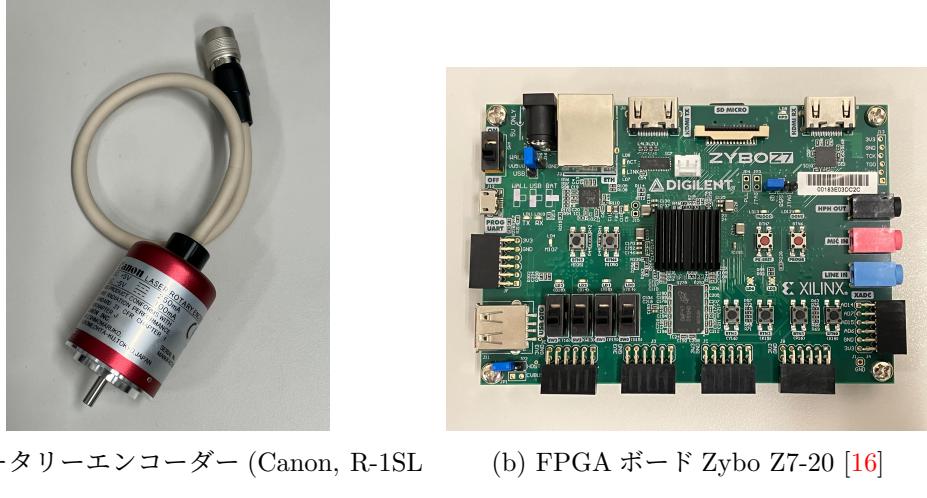


図 3.2: 仰角方向の角度データ読み出し

方位角方向の角度データは、回転台下部に取り付けられたロータリーエンコーダー (HEIDENHAIN, ERM220 [17]) を使用し、Xilinx 製の FPGA ボード Spartan3E [18] で読み出す (図 3.3)。エンコーダー自体の角度分解能は 2.6 分角 ( $4.4 \cdot 10^{-2}$  °) である。さらに、平滑化フィルターを用いた方位角データの補完を行うことで、角度分解能を  $5.7 \cdot 10^{-2}$  分角 ( $9.5 \cdot 10^{-4}$  °) に向上させている [19]。

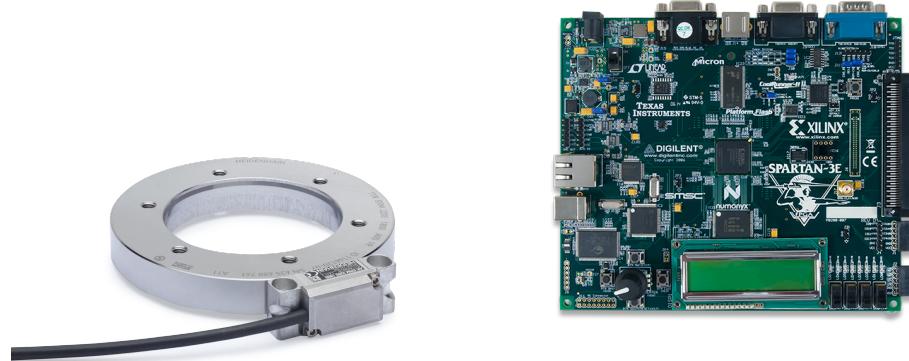


図 3.3: 方位角方向の角度データ読み出し

次に検出器のデータと方位角データに求められる同期精度を見積もる。時刻同期の精度を  $\Delta t$  とする。GroundBIRD の方位角方向の回転速度は最大で 120 °/s になる。方位角方向での角度分解能  $\Delta\phi$  は

$$\Delta\phi = 120^\circ/\text{s} \cdot \Delta t \quad (3.1)$$

になる。時刻同期による角度の決定精度がエンコーダーの角度分解能よりも十分小さいことを課す。時刻同期による角度決定精度をデータ補完によって向上したエンコーダーの角度分解能である  $9.5 \cdot 10^{-4}^\circ$  の 1% 未満と要求すると、 $\Delta t$  の上限は

$$\Delta t < \frac{9.5 \cdot 10^{-4}^\circ \cdot 0.01}{120^\circ/\text{s}} = 79 \text{ ns} \quad (3.2)$$

となる。この正確な時刻同期が必要になるため、仰角と方位角の角度データ読み出しで共に FPGA ボードを使用している。先行研究 [19] により、時刻同期精度を 55 ns に抑え、要求を満たす精度を実現している。

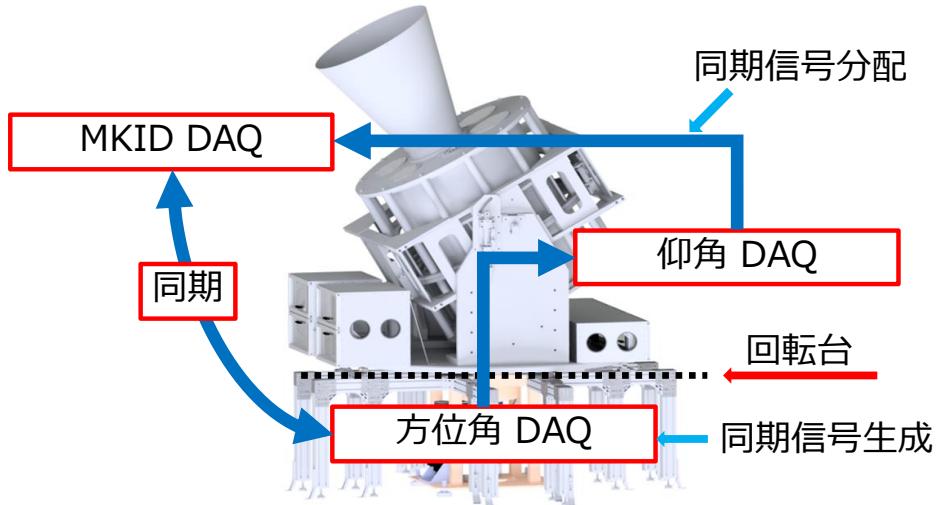


図 3.4: GroundBIRD の同期信号の流れ。回転台下部の方位角 DAQ ボードで生成された同期信号が回転継手を介して回転台上部の MKID の DAQ ボードに送信される。その際、仰角 DAQ ボードで同期信号を分配している。

GroundBIRD での同期信号の流れを説明する (図 3.4)。望遠鏡が連続回転するため、回転台の上下の電気的な接続に同軸ケーブルのような通常の信号線は使用できない。そのため、回転継手を介して回転台上下での信号を共有している。また、回転台の上下のデータ取得系でレートの遅い “同期信号” を回転継手を介して共有することで時刻の同期を図っている。

同期信号によるデータ同期を次のステップで行なっている。

1. 回転台下部の方位角 DAQ ボードから 1 秒に 1 回、基準となる同期信号を出力する。
2. 回転継手を介して回転台上部に届いた同期信号を仰角 DAQ ボードに入力する。
3. 回転台下部では、同期信号を出力した時刻情報を方位角のエンコーダーデータと共に保存する。
4. 仰角 DAQ ボード同期信号を分配し、検出器の DAQ ボードに送る。
5. 同期信号の到達した時刻情報を検出器の TOD と共に保存する。
6. 2 種類の TOD の時刻情報を用いて、各時刻での検出器信号と角度データの同期を行う。

### 3.1.2 仰角データ取得における問題点

本論文の研究対象である仰角のデータ取得に関して詳細に説明する。仰角のデータ取得系が担う役割は以下の2つである。

- 仰角の角度データを取得する
- 同期信号の分配

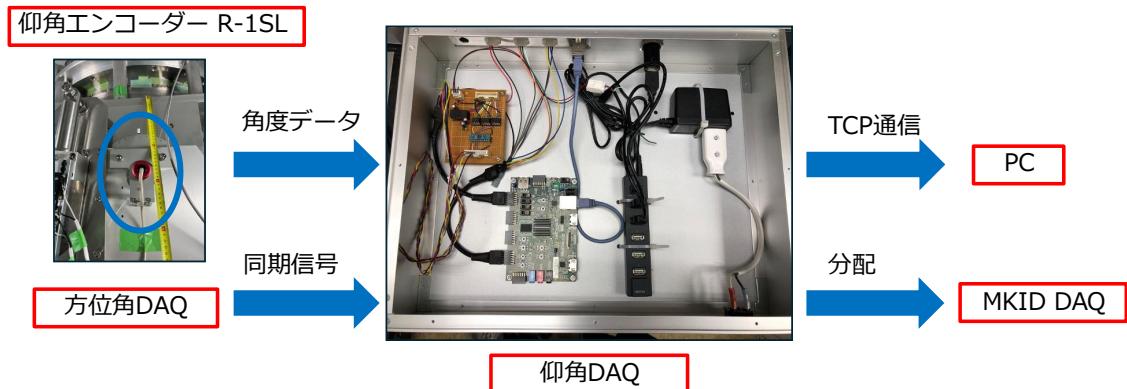


図 3.5: 仰角データ取得システムの全体像。信号処理のほぼ全てを Zybo [16] が担う。信号の電圧変換、Zybo、電源からなるコンパクトなデータ取得系は1つのボックス内に配置されている。

データ取得系は非常にコンパクトであり(図3.5)、信号の処理は Zybo 内に搭載されている“Zynq [20]”と呼ばれるチップで行っている。Zynq は Xilinx が開発した、CPU、FPGA などが1チップに統合された System on Chip (SoC) の1つである。Zynq は CPU を搭載する Processing System (PS) 部分と、FPGA を搭載する Programmable Logic (PL) 部分に大きく分けられ、FPGA が得意とする並列処理や高速処理と、CPU が得意とする複雑な処理とで役割を分担できるため、効率の良い処理が可能となる。先行研究で FPGA に搭載するファームウェア<sup>1</sup>の開発がなされており、実装されている。

FPGA 上でのファームウェアの全体図を図3.6に示す。設計には Vivado という Xilinx 製の開発ソフトを用いる。様々な IP コア(機能ごとの回路のまとめ)を IP インテグレータという GUI 上で配線し、全体のファームウェアを構成する。IP は自作の IP と、Xilinx 製の IP を組み合わせて使用している。

Zynq 内での処理の模式図を図3.7に示す。仰角エンコーダーはインクリメンタル方式を採用しており、エンコーダーからの出力信号は A 相、B 相、Z 相の3相からなる。エンコーダーからの出力パルス数で角度の変化量、A 相と B 相のパルスの立ち上がり順で回転方向が分かる仕様になっている。また、Z 相信号は1回転で1度出力され、回転の原点として使用される。3相の信号は Zynq 内のデコーダー部分で角度情報として翻訳され、1kSPS で “FIFO [21]” に充填される。FIFO とは first-in first-out メモリのこと、データを格納し、取り出

<sup>1</sup>FPGA に組み込む機能を本論文ではファームウェアと呼ぶことにする。

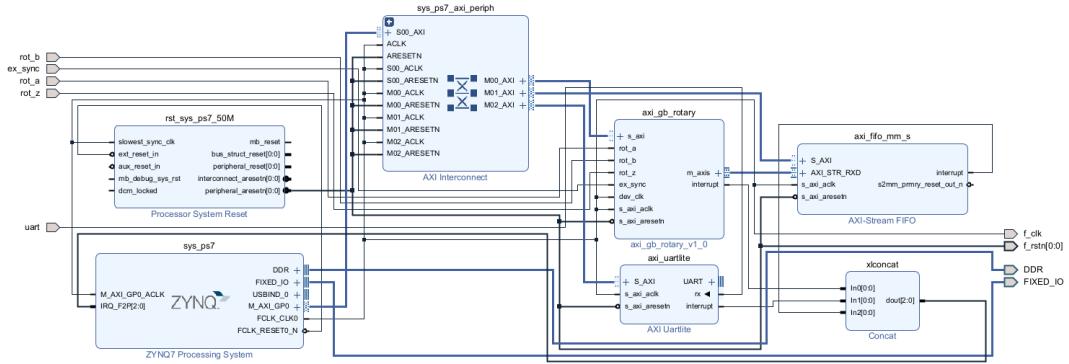


図 3.6: IP インテグレータの配線図。青く囲まれているブロックが IP コアを表す。ブロックの左側から出る線が入力信号で、右側から出る線が出力信号になっている。図中の “axi\_gb\_rotary” と呼ばれる IP が自作 IP で、デコーダーの役割を担っている。

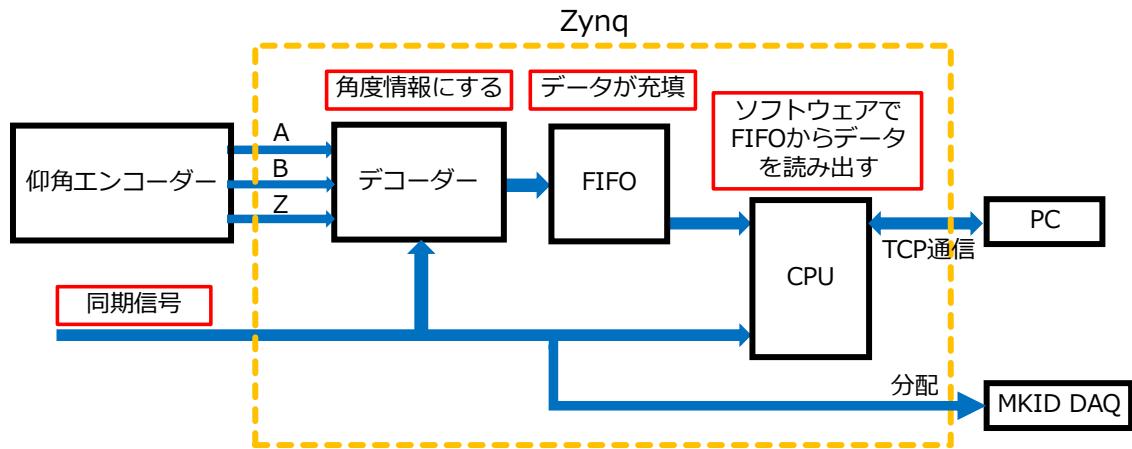


図 3.7: Zynq 内での信号処理。エンコーダーからの角度データ処理と方位角 DAQ ボードからの同期信号の分配を 1 チップ内で行う。

す際は、格納した順番通りに先に格納したデータから取り出す構成になっている。FIFOに格納されたデータはCPUのソフトウェアで読み出し、TCP通信でPCへと送信される。

同期信号の通信方式はUART<sup>2</sup>を使用している。デコーダー部分では測定開始時からの経過時間として1kHzでインクリメントする“タイムスタンプ”を生成しており、同期信号が入力されると同期信号の時間情報を読み取り、このときのタイムスタンプ情報を取得する。タイムスタンプ情報はソフトウェアから読み出す。また、MKIDのDAQは複数のボードを使用するため、同期信号を分配させ、送信している。

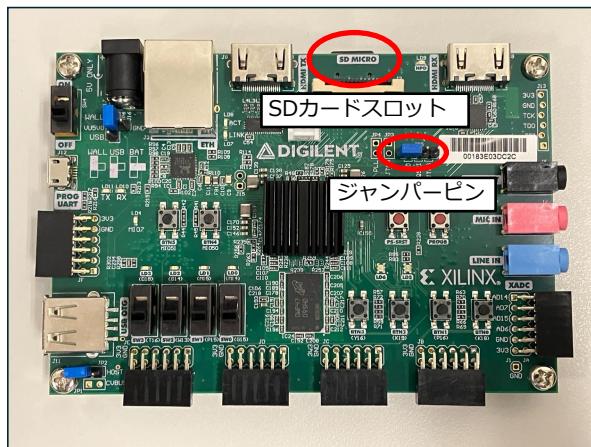


図 3.8: Zybo には SD カードスロットが付いている。システムを起動するために必要なファイルを SD カードに書き込み、スロットに差し込んで電源を入れることで稼働が開始する。ジャンパーピンは SD に設定しておく。

以上の構成で仰角 DAQ システムを稼働させていた(図 3.8)が、観測の長期運用を考えた際に、問題点を抱えていた。Zynq でのデータ処理と PC への通信を OS のない(ベアメタル<sup>3</sup>)環境でソフトウェアを動かすことで行っている。このことにより、システム全体が硬直的になっており、その扱いにおいて柔軟性がない。柔軟性がないことで起きる問題として以下のものが挙げられる。

- データ取得が異常終了した際のメンテナンスが難しい
- OS が介さない通信による信頼性の低下
- ソフトウェア、FPGA 面での改良が難しい

最も大きな問題はメンテナンス性である。観測の有無に問わず、望遠鏡の角度データは常に取得し続けており、安定した仰角データ取得が必要不可欠である。しかし、仰角エンコーダーから PC までのデータの流れが途切れてしまうことがある。それは、通信のエラーや停

<sup>2</sup>UART 通信では、送信側と受信側で通信速度を決めておき、1 byte (8 bits) ずつ情報のやり取りを行う。1 byte のまとめでは、“start bit (1 bit)”、“data (8 bits)”、“parity bit (1 bit、情報の誤検知に使用)”、“stop bit (1 bit)” の 4 つで構成され、全 11 bits になる。

<sup>3</sup>本来は「剥き出しの金属」という意味だが、転じて OS がインストールされていないコンピュータのことを指す。

電等による電源系統の不具合など様々な要因からくるもので、長期運用をする上ではある程度避けられないものである。その際、エンコーダーと Zybo 間の通信が切れて、Zynq 内でのエンコーダー情報が失われる。加えて、エンコーダーの原点情報も失われるため、通信を再開させた際に読み出した角度の値にオフセット値が乗ってしまう。Zynq 内で原点情報を記憶させるには望遠鏡の仰角を動かし、エンコーダーの Z 相信号を取得して値をリセットすることが必要であり、手間のかかる工程になる。この一連のメンテナンスをまとめると以下のステップに分けられる。

1. Zybo を再起動させて再度ソフトウェアを動かす
2. 望遠鏡の仰角を動かして Z 相信号を入力し、原点情報を記憶
3. 仰角を 70° まで動かして固定

この中で、Zybo の再起動と仰角を動かす際にケーブルに変な張力がかかっていないかの目視に現地での作業を要する。リモートでの望遠鏡運用を進める上で、少しでも現地で必要な作業を減らし、リモートでメンテナンスができることが望まれる。

また、Zybo と PC との TCP 通信を行うために、ペアメタル環境では TCP/IP のプロトコルスタックを独自で実装しなければいけない。今回は “lwIP (lightweight IP)” と呼ばれるオープンソースの TCP/IP プロトコルスタックを使用して TCP 通信を実装している。そのため、通信に関わるソフトウェアが複雑化する上、OS が通信を取り仕切るよりも信頼性に欠ける。

加えて、システムに組み込まれたソフトウェアや FPGA のファームウェアは固定化されており、今後の運用で変更点が生じた時に、SD カードに新しいファイルシステムを書き込んで全面的にシステム更新しなければならず、労力を要する。システムのカスタマイズ性を上げ、変更を容易にできるようになれば、運用に関わるコストを削減することができる。

### 3.1.3 PYNQ を用いた新システムの導入

上に述べた問題を改善するために、本研究では既存の Zynq システムに OS を搭載し、ソフトウェアを OS 上で動かすことによってシステム全体の柔軟性向上を試みた。これにより上記の問題点に対して

- データ取得が異常終了した際のメンテナンスをリモート主体で行える
- OS が通信を介することで信頼性が向上
- ソフトウェア、FPGA 面での改良をシステムを起動したまま行える

という改善が期待できる。一方でペアメタル環境に比べて OS をインストールすることで実行時間と使用メモリが増えるが、FIFO へのデータ充填と FIFO からの読み出しあともに 1 kSPS であり、FIFO の容量も十分であるため、性能に問題は出ない。Zynq に搭載する OS は基本的に linux ベースであるが、今回は “PYNQ [22]” と呼ばれる Ubuntu<sup>4</sup>、Jupyter および Python をベースとしたフレームワークを搭載した。PYNQ を採用した理由として

<sup>4</sup> 今回は Ubuntu 22.04 がベースになっている

- ソフトウェアを PYNQ 上の Python スクリプトで動かせるため、通信スクリプトを簡潔に記述できる
- システム起動に必要なブートイメージファイルの作成が容易
- “Overlay” と呼ばれる機能を使用することで FPGA のファームウェアを Python で容易に変更できる

が挙げられる。

### 3.1.4 PYNQ イメージファイルの作成

PYNQ のイメージファイルを作成するにあたって、基本的な手順は [23] を参考にし、作業環境は Docker 上の Ubuntu 20.04 で構築した。その際に使用した開発ツールとバージョンを表 3.1 にまとめる

表 3.1: PYNQ イメージ作成で使用したツールとバージョン

ツール	バージョン
Vivado	2022.1
PYNQ linux イメージ	3.0.1
PetaLinux	2022.1

PetaLinux とは Xilinx が提供する、Zynq をはじめとする SoC 用の linux システムをビルドするためのツールである。この環境のもとで以下のステップで PYNQ イメージを作成した。

#### 1. ベースとなる FPGA ファームウェアの作成

FPGA ファームウェアは Overlay によって Zybo 起動時に変更できるため、この時点では最も単純な回路を Vivado で準備してやれば良い。それをもとに Vivado 上でコンパイルをし、生成された 4 つの回路情報を持つファイル (.xsa ファイル、.bit ファイル、.hwh ファイル、.tcl ファイル) を取得する。

#### 2. Zybo 用のスペックファイルの作成

スペックファイルは Zybo のスペック情報を.spec ファイルとして作成する。その後、作成した全ファイルをビルド用のディレクトリに置く。

#### 3. ビルド

最後に make を実行してビルドを行い、PYNQ イメージファイル (.img ファイル) を生成して完了する。

#### 4. イメージファイルの書き込みと PYNQ 起動

作成したイメージファイルを SD カードに dd コマンド等で書き込み、Zybo の SD カードスロットに差し込むことで PYNQ が起動する。また、起動時に Overlay スクリプトを走らせて図 3.6 で示したファームウェアで FPGA の回路を上書きするように設定した。これにより、本来の回路情報が PYNQ 上で再現される。

Overlay を用いて FPGA のファームウェアを変更する仕組みを図 3.9 に示す。



図 3.9: PYNQ から FPGA のファームウェアを変更する模式図。Vivado で設計した新しい回路ファイルを Overlay クラスに読み込ませることで容易に変更ができる。

Overlay に必要なファイルは Vivado でコンパイルをして生成される.bit ファイルと.hwh ファイルの 2 つである。これらは同じディレクトリに置いておく。Python で Overlay クラスをインポートし、.bit ファイルを読み込ませることで Overlay は実行される。その際、Overlay クラスは同じディレクトリにある.hwh ファイルも読み込んでくれる。図で示したように Overlay スクリプトはたった 2 行で書くことができ、ファームウェアの変更は非常に容易に実行できる。

最後に、Zynq 内で動かすスクリプトを PYNQ 用に Python で再構築した。

## 3.2 望遠鏡への実装

### 3.2.1 新データ取得システムのインストール

新しく導入した仰角データ取得システムを、自分の手元でできる動作確認をした後に実際に望遠鏡に実装、という流れでインストールした。

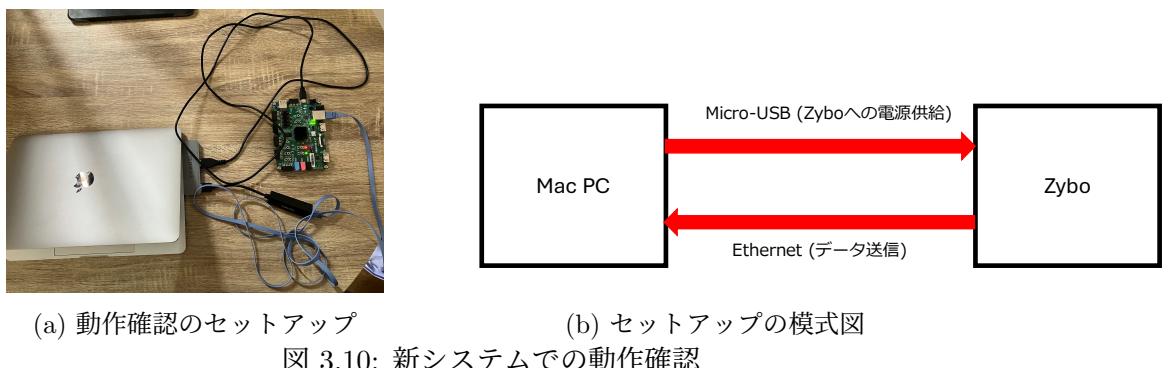


図 3.10: 新システムでの動作確認

動作確認を、図 3.10 に示したセットアップで行なった。ロータリーエンコーダーとは接続せず、同期信号も送らない単純なものである。図 3.10 の左図にあるように、Zybo の電源を入れ PYNQ が起動するとボード右側の “DONE” ランプが緑色に点灯する。自分の Mac PC をデータ取得 PC として、Micro-USB で Zybo への電源供給を行い、データ通信は Ethernet

ケーブルで行った。PC 上でデータ読み出しのスクリプトを動かして、Zynq の挙動に問題がないかをチェックした。期待される挙動は

- Zynq が正しく機能し、PC 側でデータを読み出せる
- エンコーダーデータは 0 として取得されている
- タイムスタンプは時間とともに加算されている

の 3 点である。データの通信が始まると Zybo の Ethernet コネクタ付近のランプが点滅し、PC 側でのデータ読み出しが正常に動くことを確認した。PC に保存したデータファイルをチェックし、期待されるデータを取得できていることも確認した。取得したエンコーダーデータとタイムスタンプデータのプロットを図 3.11 に示す。プロットの横軸はデータのサンプリングナンバー (1 kSPS) を表す。

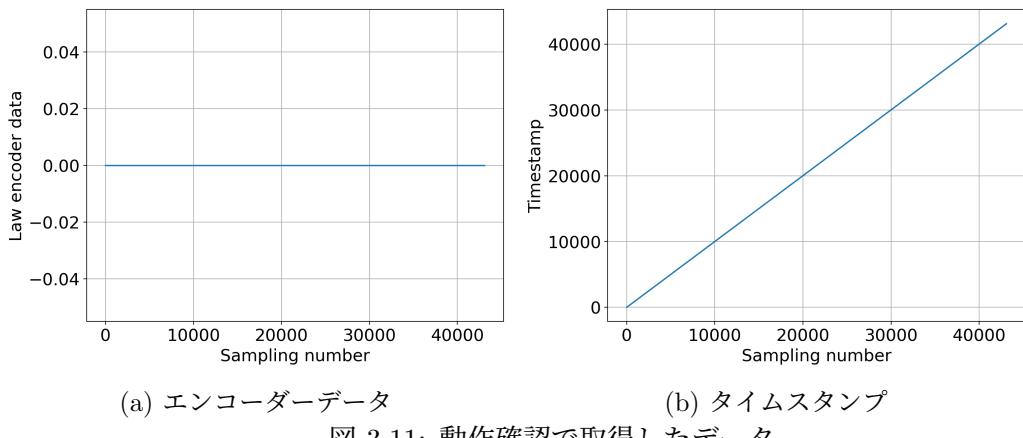


図 3.11: 動作確認で取得したデータ

以上の結果から新システムの挙動が問題ないと判断し、次に実際の望遠鏡にインストールした。インストールは SD カードを新システムのものに交換するだけで完了し、PYNQ の起動を確認した。

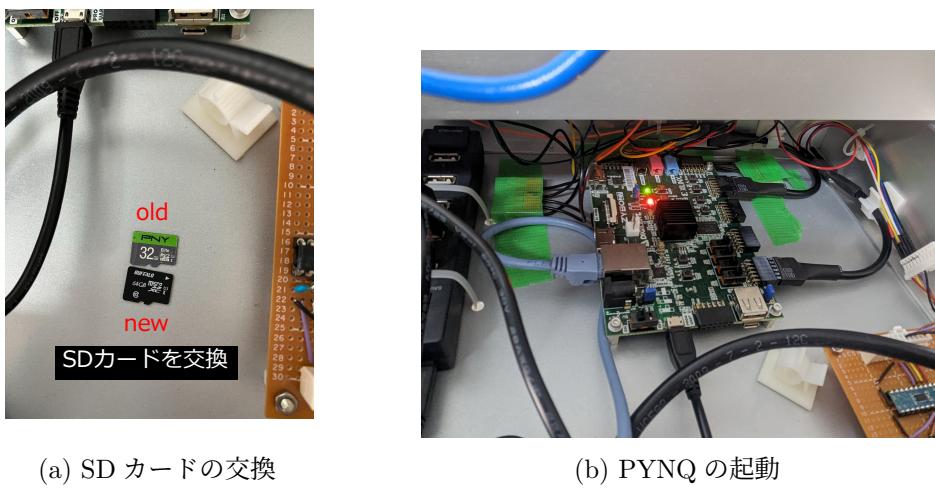


図 3.12: 新システムのインストール

### 3.2.2 同期信号取得の確認

インストール後に実際の望遠鏡システムでデータを読み出せるかの動作確認を行なった。まず、方位角 DAQ からの同期信号を正しく取得して仰角データとして保存できているのかを確認した。結果を図 3.13 に示す。同期信号は 1 秒に 1 回出力されるので、仰角データでは同期信号の番号(図では“Sync\_id”と記す)が 1 秒で 1 ずつ増加する形で見えるはずであり、その結果を確認することができた。

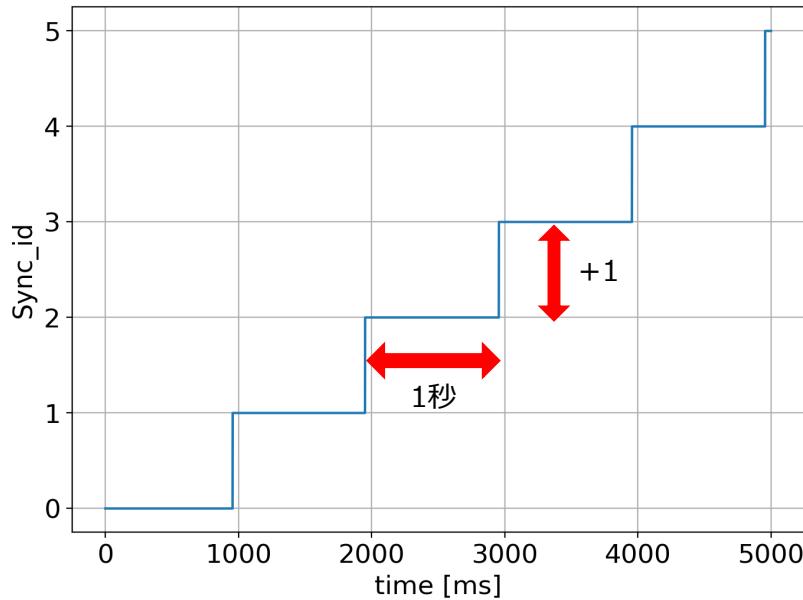


図 3.13: 仰角データからみた同期信号。方位角 DAQ からの同期信号を取得するごとに“Sync\_id”が 1 ずつ加算される。

### 3.2.3 同期信号の分配と仰角データ取得の確認

次に、取得した同期信号を MKID DAQ に分配できていることと仰角データを正しく読み出しているかを確認した。一度失ったエンコーダーの原点情報を再度取得するためにも、望遠鏡の仰角を  $90^\circ$  と  $70^\circ$  の間で何度か動かして、さらに並行して MKID のデータも取得した。それらのデータを使って確認を行なった。確認の手順は以下の通りである。

1. テスト用として取った MKID データを読み出す
2. 正しく動作していれば MKID データに同期した時間情報と仰角データを取得できる
3. その仰角データが正しい値を読み出しているかを確認

結果を図 3.14 に示す。MKID データから同期情報を取得でき、読み出した仰角データが  $90^\circ$  と  $70^\circ$  の間で正しく動いていることも確認した。

以上から新システムが実際の望遠鏡で問題なく動作することを確認した。その後、動作が安定して長期間行われるかをチェックした。

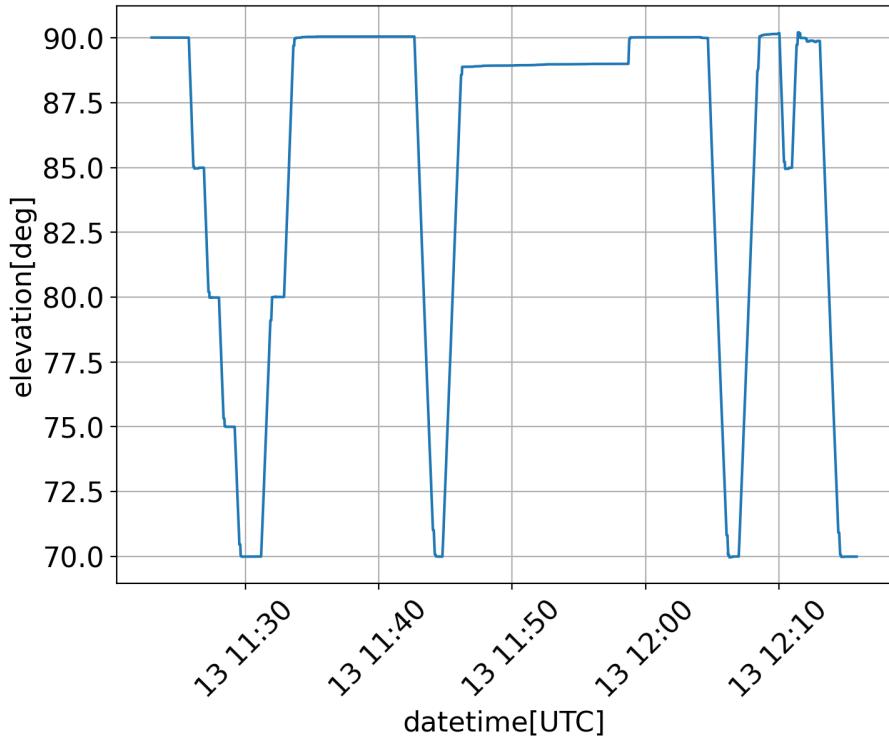


図 3.14: 読み出した仰角データ。横軸 (2024/3/13 の UTC 時間) の時間情報が実際の作業時間とリンクしており、MKID データで同期信号が正しく取得できていることを反映している。

### 3.3 メンテナンスと安定運用

#### 3.3.1 動作の不安定性

インストール後、動作の安定性に問題があり、データ取得が途切れることが何度か発生した。途切れる原因は Zybo の電源が一時的 (データ取得開始後数時間) に落ちることによるものであった。インストール時の動作確認で問題がなかったことから、Zynq 内でのデータ処理と通信自体に問題がある可能性はないと考えた。他に原因となりうるものは

- OS が搭載されたことで Zybo の消費電力が上がり、一時的に電源供給量が足りなくなる
- Zynq での消費電力も上がり、Zynq の温度が許容値よりも高くなってしまう
- そもそも Zybo のボード自体がどこかで劣化している

が挙げられる。3 つ目に関しては、経年劣化や落雷による停電時にダメージを受けたことなどが考えられるが、リモートからボード自体の性能を評価することが難しいため、まずは 1 つ目と 2 つ目の要因について調査した。結果として、1 つ目の電源供給が原因である可能性が高いことが分かった。

### 3.3.2 電源供給方法の見直し

Zybo への電源供給は図 3.12 の右図にあるように、Micro-USB 端子から行なっていた。Zybo への電源供給方法は他にバレルジャックから (図 3.15) がある。

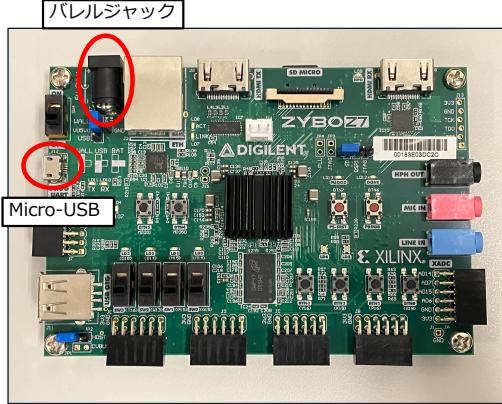


図 3.15: Zybo への電源供給方法。Micro-USB からとバレルジャックからの 2 通りがある。

この 2 種類の供給方法について比較すると表 3.2 のようになる。バレルジャックについては標準的な AC アダプタの規格を参照する。

表 3.2: 各電源供給方法での定格値

供給方法	定格電圧	定格電流	接続の安定性
Micro-USB	5 V	最大 0.5 A	やや不安定
バレルジャック	5 V	最大 4 A	安定

これより、供給できる電力量や接続の安定性に関してバレルジャックの方が優れていることが分かる。特に定格電流の値が大きく異なっており、従来のシステムでは Micro-USB からの給電で間に合っていたが、OS が搭載されたことで給電が足りなくなった可能性が考えられる。スペックシート [24] でもバレルジャックによる電源供給が推奨されている (参考として、電力を大量に消費する処理を Zynq に搭載した場合は 12.5 W 以上の出力が必要)。そのため、Zybo への電源供給方法をバレルジャックに変更して十分な電力を供給することで動作の安定性を図るのが良いと考えた。変更するにあたって新システムを稼働した際の Zybo の動作安定性を給電条件を変えてテストした。行なったテストの様子を図 3.16 に、給電条件と結果を表 3.3 に示す。

表 3.3: 安定稼働テストの給電条件と結果

	給電条件	動作の結果
テスト 1	従来と同じ USB	安定
テスト 2	バレルジャック (5 V、4 A の AC アダプタ)	安定
テスト 3	直流電源 (定電圧モード 5.0 V)	安定
テスト 4	直流電源 (定電圧モード 4.5 V)	不安定 (数時間で Zybo の電源が落ちる)

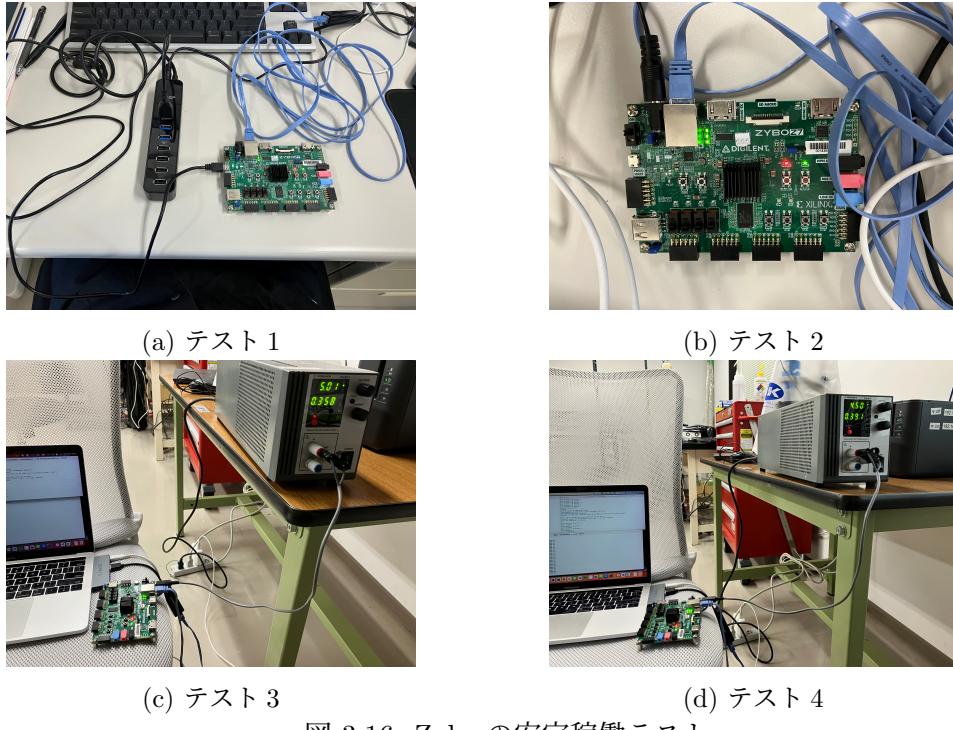


図 3.16: Zybo の安定稼働テスト

基本的なセットアップは図 3.10 と同じで給電の条件のみが異なっている。テスト 1 では従来と同じで Micro-USB からの給電でデータ取得を動かし、数日にわたって動作の安定性を確認し、少なくとも 3 日以上は Zybo の電源が落ちることなく稼働する結果を得た。これは現地で起きている問題と矛盾する結果となった。テスト 2 ではバレルジャックから給電し、こちらも安定した稼働をすることを確認した。稼働が不安定である原因が特定できなかったため、条件を変え、さらに検証を行なった。

テスト 3 と 4 では直流電源(松定プレシジョン製)を使用し、与える電圧値を変えて稼働の安定性を確認した。USB の定格電圧が 5V であることと、Zybo で推奨されている供給電圧が 4.5V~5.5V であることから 5.0V と 4.5V を設定した。電源を定電圧モードにすることで電流値は Zybo でデータ取得システムを動かすために必要な電流量になる。テスト 3 では 5.0V で行い、安定した稼働結果と電流値 ~ 0.36 A を得た。テスト 4 では 4.5V で行い、データ取得開始後数時間で Zybo の電源が落ちる事象が複数回起き、現地での問題と同様の挙動を確認した。また、稼働中の電流値は ~ 0.39 A であった。これより、データ取得システムの稼働に必要な電力量は比較的少ないと分かることは、実際はエンコーダーデータと同期信号を取得し処理しているため、消費電力がこれよりも少なからず増えると考えれば、システムの稼働には Micro-USB の定格電流値の 0.5 A に近い電流量が必要になる。

テストの結果を踏まえると、Zybo への供給電圧が定格通り 5.0 V であればシステム稼働に必要な電流量が USB の定格電流値を下回り十分な電力を供給できるが、電圧値のふらつきで 5.0 V より小さくなると、稼働に必要な電力量を保持するために電流量が増加し、定格電流値に近づくために USB での給電が一時的に不足して Zybo の電源が落ちる可能性があると言える<sup>5</sup>。そのため、性能面に加えて、実際の問題への対処という意味でもバレルジャッ

<sup>5</sup>Zynq 内蔵の “XADC” で Zynq への供給電圧をモニターできるが、Zybo 全体への供給電圧をモニターで

クに変更することの妥当性を検証することができた。

Zyboへの電源供給をMicro-USBからバレルジャックに変更し(図3.17)、システムを再起動させた。その後、動作が安定することを確認した。再起動して以降、半年以上の安定動作を続けている。

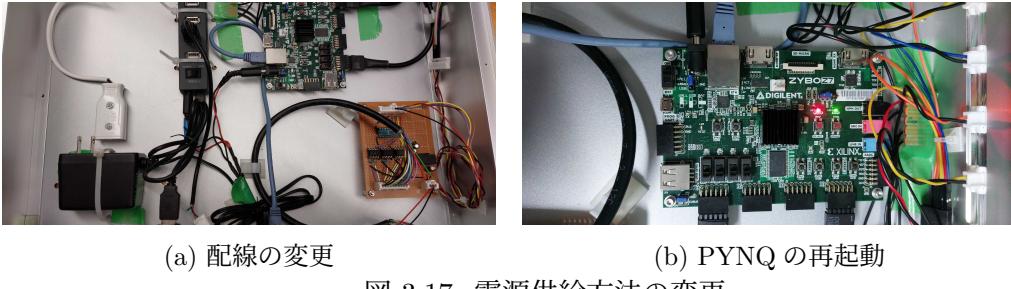


図3.17: 電源供給方法の変更

### 3.3.3 Zynq 温度のモニター

2つ目の要因についても調査し、問題がないことを確認した。ZyboのZynq部分にはヒートシンクが取り付けられており、ある程度の発熱は抑えられるはずだが、消費電力が高いと発熱量が増えてZynqの動作温度の上限( $\sim 85^{\circ}\text{C}$ )を超えることは起きうる。しかし、3.3.2で見たように新システムの消費電力が大きくなり、稼働中にヒートシンクを手で触っても熱くないため上限を超えるほどの発熱をしていることはまずない。そのことをZynq内の温度モニター機能を用いて確かめた。Zynqには“XADC(Xilinx Analog to Digital Converter)[25]”と呼ばれるADCが内蔵されており、供給電圧と温度のモニタリング機能を有している。電圧と温度の値はOSシステムの[ /sys/bus/iio/devices/iio:devices0 ]ディレクトリに出力される。特に温度情報は

- in\_temp0\_offset
- in\_temp0\_raw
- in\_temp0\_scale

の3つの値で与えられ、Zynq温度は

$$T_{\text{Zynq}}(^{\circ}\text{C}) = (\text{in\_temp0\_raw} + \text{in\_temp0\_offset}) \cdot \text{in\_temp0\_scale}/1000 \quad (3.3)$$

で求められる。定期的にZynqの温度を読み出し、値を保存することで温度を確認できるようにした。読み出した温度のプロットを図3.18に示す。気温の影響もあるためZynq温度は夏季になると全体的に高くなるが、それでも動作温度の上限である $85^{\circ}\text{C}$ よりも十分低い温度で動作していることが分かる。

今後は新しく導入したデータ取得システムの動作状況のモニターを継続して、安定的な運用とメンテナンスを続けていく。

---

きるわけではないため、実際のデータ取得システムに供給される電圧、電流値を正確に知る手法は確立できなかった。

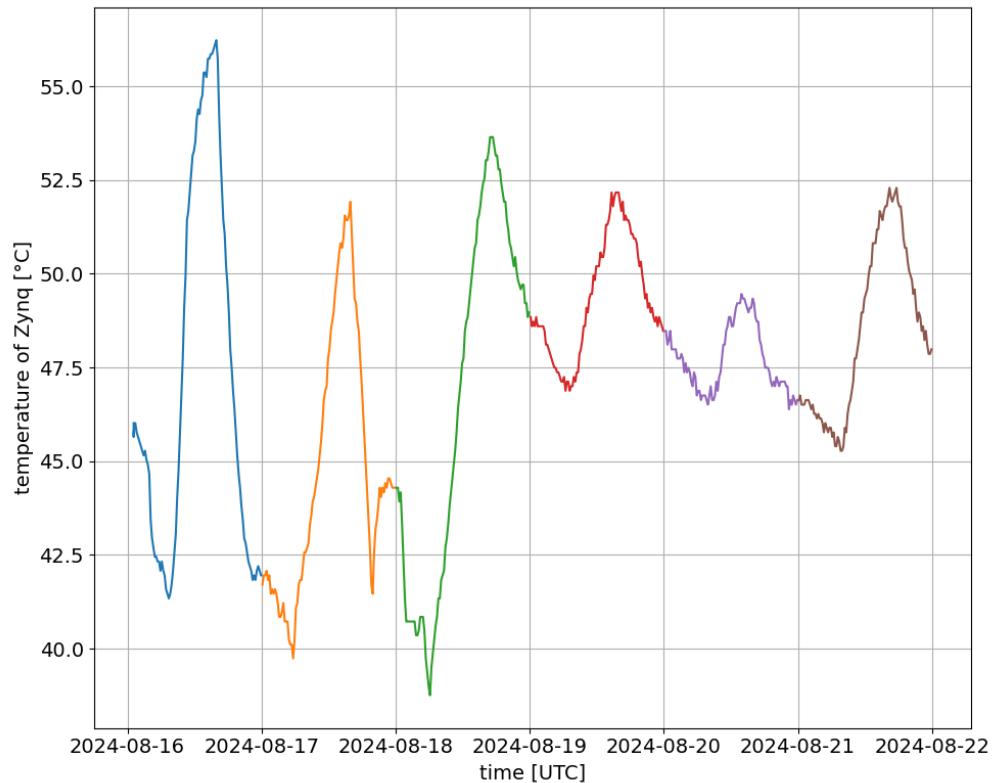


図 3.18: Zynq の温度モニター。2024/08 の 6 日分のデータをピックアップした。1 日の中でも気温の影響を受けて Zynq の温度も変動する。そのため、夏季の温度は冬季よりも全体的に高くなる。

# 第4章 焦点面検出器アライメントの較正

GroundBIRD 実験での偏光測定のためには、検出器間での信号の差分を取ることが重要であり、それに伴って望遠鏡のスキャンに対して最適な検出器のアライメントが求められる。この章では検出器アライメントの最適化に向けた改善を行い、観測データからその効果を確認した。

## 4.1 検出器アライメントの問題点

### 4.1.1 スキャン軸に対する傾きと差分解析

まず、GroundBIRD での解析手法の 1 つである差分解析と検出器アライメントの関係について述べる。[2.9b](#) で示したように、焦点面検出器は異なる偏光方向に感度を持った検出器が交互に配置されている。これらの検出器が検出する信号は空のある点からの放射が望遠鏡内の光学系を経て焦点面へと届いたものである。つまり、焦点面での検出器の配置を空へと射影した時にどう配置されているかが重要になる。各検出器は空のある点を見ており、望遠鏡の方位角回転に伴って同じ仰角の空を回転しながら観測する ([図 4.1](#))。検出器で観測す

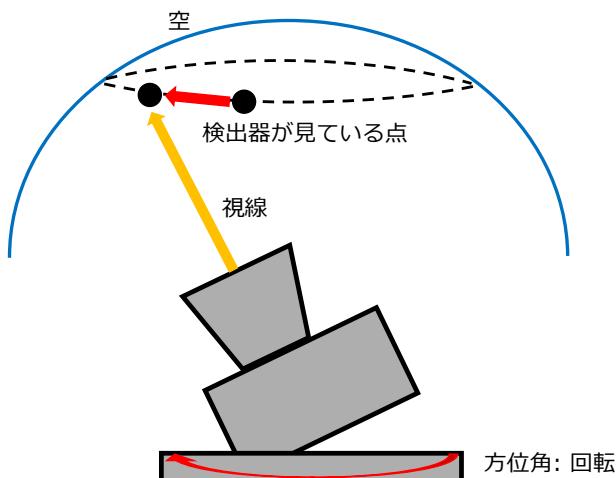


図 4.1: 検出器が空の領域をスキャンする概要図。ある仰角を高速回転しながらスキャンする。

る信号は大きく (CMB + ノイズ) に分けられる。さらにノイズの中でも寄与が大きい成分に大気放射に由来するノイズがある。このノイズは刻一刻と変動する上に空の領域によっても異なっているため、高速回転によるスキャンで異なる検出器が同じ空の領域をスキャンす

ることで抑制できる。具体的には、異なる偏光方向に感度のある検出器が同じ仰角の空をスキャンする時、もう一方の検出器は片方の検出器が観測した空の領域をわずかな時間差で観測することができる。つまり、スキャンの間に大気の情報は変動せず観測される大気由来のノイズも変動しないことになる。そのため、検出器間で信号の差分をとることで、大気ノイズは共通していると考えれば取り除くことができ、偏光成分のみを残すことができる。しかし、検出器の配置がずれていて検出器間でスキャンする空の領域が異なる場合、大気の情報も異なり、差分をとっても大気ノイズを取り除くことができない(図4.2)。

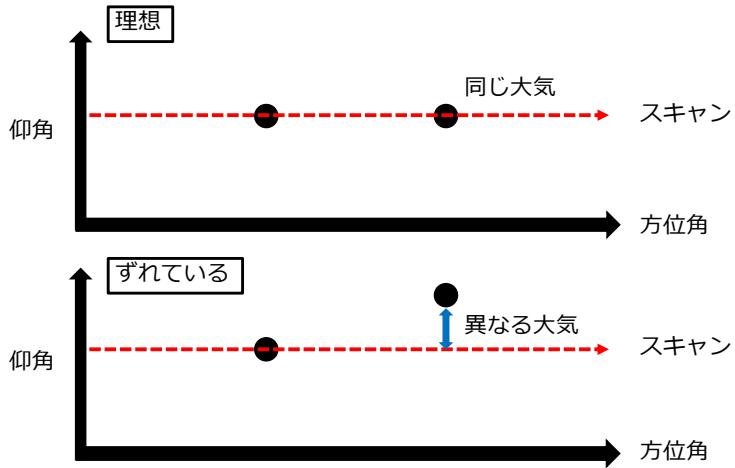


図 4.2: 空での理想的な検出器の配置とずれている場合の配置との比較。スキャン軸(方位角軸)に沿って検出器が並んでいないと観測する大気が検出器ごとに異なる。

以上から空での理想的な検出器の配置は“複数の検出器がスキャン軸に沿って並んでいる”ことである。

しかし、観測データから理想的な検出器の配置からずれていることが示唆されていた。また、そのずれはスキャン軸に対して無視できないほどに有意な角度で傾いているものだと考えられていたが十分な検証と較正(実際に何度傾いているのか、傾きがあることでどれ程大気ノイズの影響が残ってしまうのか、など)がされていなかった。検出器アライメントの問題を改善し、GroundBIRD が持つ観測性能を最大限に引き出すことは質の良いデータを取得するためには不可欠である。

#### 4.1.2 要求される理想的なアライメント

GroundBIRD における理想的な検出器の配置について詳細を見ていく。焦点面検出器は図 2.9a のように 7 つの検出器アレイが平面的に取り付けられているが、この検出器が観測する領域は平面のまま空に射影される訳ではない。観測する空の点は天球面上に張り付いた点と考えられるため、球面として射影される。焦点面検出器が平面を見るときと空を見る時の理想的な配置の違いを図 4.3 に示す。実際には球面から来る歪みの影響を受けた配置として空を観測することになり、望遠鏡の視線中心(220 GHz アレイ)ではほぼ平面だが、中心から離れた検出器は歪みの影響が出る。歪みを考慮した上で複数の検出器をスキャン軸に沿っ

て並べることは焦点面の設計上難しい。また、高周波になるほど大気放射の寄与が大きくなる[26]ことから、本論文では歪みの影響が少なく、大気放射の寄与も大きい中心の220 GHzアレイに対して配置がスキャン軸に沿って並んでいることを理想的なアライメントとする。

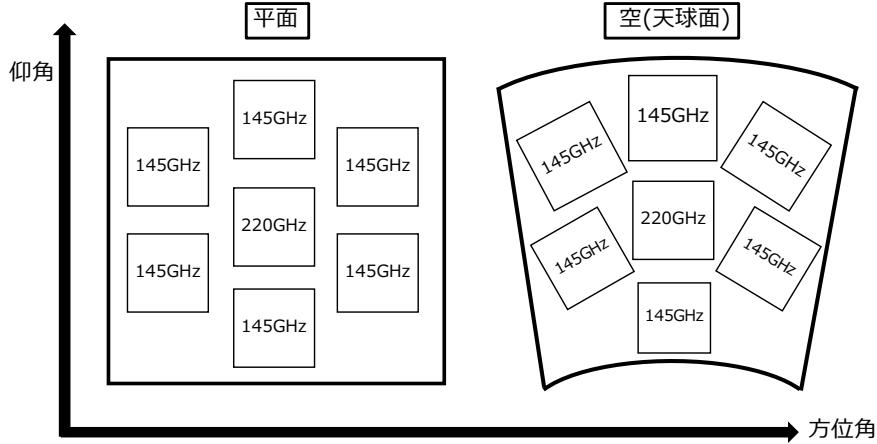


図 4.3: 平面と空(天球面)での理想的な検出器配置の違い。

#### 4.1.3 視線軸方向まわりの回転による較正

スキャン軸に対して傾いた焦点面検出器を理想とする配置にするためには、検出器の視線を回転させてスキャン軸に並べれば良いことになる。また、焦点面検出器は望遠鏡内部で固定されているため望遠鏡全体の視線を回転することに対応する。つまり、望遠鏡のビーム重心を望遠鏡の視線方向とし、視線方向軸周りに適当な角度回転させることで各検出器の視線を回転させる(図 4.4)。

そのため、理想的なアライメントにするための回転角を見積もある必要がある。回転角を求めるには各検出器の見ている点を知る必要があり、天体の観測データを解析することで算出できる(天体の運動が分かっているので検出器データと角度データから各検出器の視線情報を求められる)。

## 4.2 月を用いた回転角の算出

### 4.2.1 月を用いた理由

GroundBIRD で観測できる天体として月や惑星(木星、金星)が挙げられる。しかし、月と惑星ではデータの性質は大きく異なり、それぞれ観測上の長所と短所が存在する。月データと惑星データの特徴を表 4.1 に示す。検出器の視線情報を得るには点源として扱え、正確に点として求められる惑星が適しているが、4.4.2 で見るようによく点源で最も明るい木星の観測データでもノイズの影響をかなり受けてしまう。そのため、データを蓄積するか、PWV や湿度が十分低く観測条件が整った観測データを使用する必要がある。一方、月は高い S/N 比により 1 度の観測データで十分な信号を得られ、データを蓄積することによるノイズが加

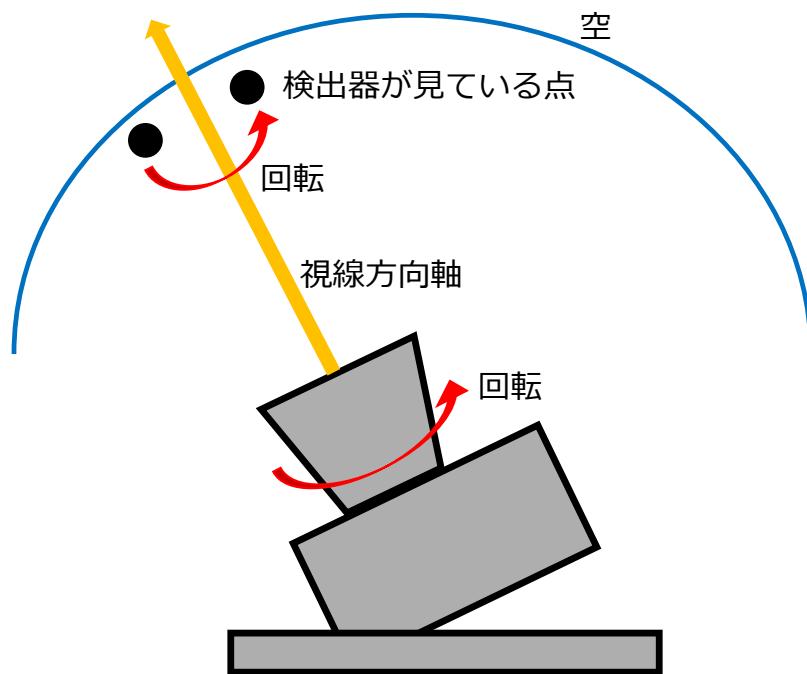


図 4.4: 望遠鏡の視線方向軸と軸周りの回転。検出器の見ている点(視線)も回転する。

表 4.1: 月データと惑星データの比較 [13]

	月	惑星
長所	S/N 比が高い (1 度の観測で十分な信号を得られる)	点源として扱える (視野角がビーム幅に比べて十分小さい)
短所	点源として扱えない (輝度温度の非一様性が生じる)	S/N 比が小さい (データを蓄積しないとノイズに埋もれる)

わることなく解析ができる。点源として扱えないことによる誤差はあるものの、高い S/N 比がそれをカバーできると考えたため、月のデータを選択した。

#### 4.2.2 位相としての検出器 TOD

GroundBIRD における観測データは 2.1.2 で述べた超伝導検出器 MKID の共振状態の変化を入射信号の大きさとして 1 kSPS で取得する TOD のことである。共振状態の変化とはすなわち共振周波数の変化であるが、これを読み出し RF 信号の透過率の変化として測定する。透過率は散乱行列要素の  $S_{21}$  で表す。MKID の  $S_{21}$  は共振の鋭さを表す  $Q_r$ ,  $Q_c$  を用いて

$$|S_{21} - x_c| = \frac{Q_r}{2Q_c} \left( x_c = 1 - \frac{Q_r}{2Q_c} \right) \quad (4.1)$$

で得られ [28]、 $S_{21}$  の軌跡が円状になることが分かる。この円は“共振円”と呼ばれる。共振円と  $S_{21}$  の例を図 4.5 に示す。共振周波数の周りで透過率が鋭いピークを持つことが分か

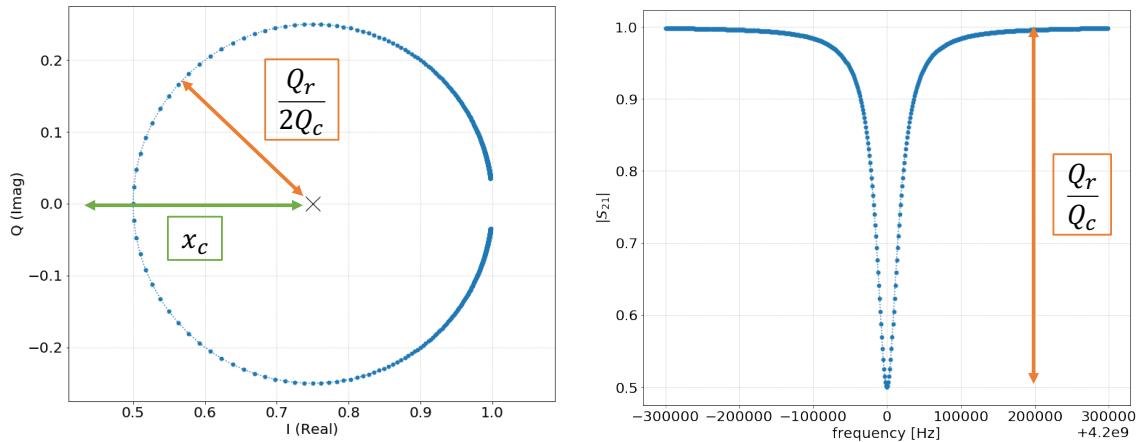


図 4.5: 共振円と共振周波数付近での透過率。[27] より引用。

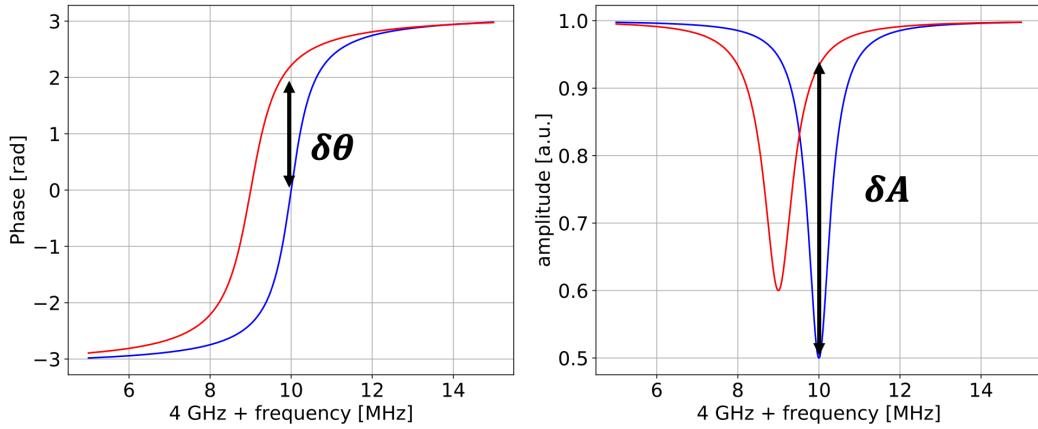
る。共振円を動く  $S_{21}$  の振幅  $A$  と位相  $\theta$  を

$$A = \frac{|S_{21}| - x_c}{1 - x_c} \quad (4.2)$$

$$\tan \theta = \frac{\text{Im}(S_{21})}{x_c - \text{Re}(S_{21})} \quad (4.3)$$

と定義する。これらの値は共振周波数の変化に対して敏感であるため、振幅の変化 ( $\delta A$ ) と位相の変化 ( $\delta \theta$ ) によって入射信号の大きさを読み出せる。入射信号によって振幅と位相が変化する例を図 4.6 に示す。

実際の TOD 取得時には図 4.6 のように各 MKID で固定した共振周波数で測定を行う。そのため、TOD 取得前に各 MKID の共振周波数を知る必要がある。また、共振周波数は観測条件によって変動するため、一定の値をとらない。そのため、1 観測を 1 時間で区切っており、1 観測ごとに TOD 取得前に共振周波数を求めている（その間も望遠鏡は連続的に稼働している）。共振周波数を知るために

図 4.6: 共振周波数付近での  $S_{21}$  の振幅と位相の変化。[13] より引用。

## 1. 周波数スイープ

## 2. フィッティング

の手順で測定をする。周波数スイープとは、読み出し用の RF 信号の周波数を少しづつ変えながら測定をする手法である。測定量は RF の透過率で、スイープで測定されたデータを透過率の関数でフィットすることで共振周波数を求める。フィット関数は  $S_{21}$  に補正項を入れたもので

$$T_{21}(f) = a_0 \exp(-2\pi i f \tau_0) \left( 1 - \frac{Q_r/Q_c e^{i\phi}}{1 + 2iQ_r(f - f_r)/f_r} \right) \quad (4.4)$$

で表される [27]。ここで、 $a_0$ ,  $\tau_0$  は読み出し回路による振幅の減衰と位相のずれを表す。 $e^{i\phi}$  は C カップリングでのインピーダンスを補正する項である。このフィットで共振周波数  $f_r$  と共振の鋭さを表す  $Q_r$ ,  $Q_c$  を得られる。得られた  $f_r$  に RF 周波数を固定し、TOD を取得する。TOD での測定量は  $T_{21}$  であり、補正項の効果を差し引くことで  $S_{21}$  としての振幅  $A$  と位相  $\theta$  を取得できる。本論文ではより応答性の高い位相を TOD として使用する。また、非線形効果を補正したものを最終的に使用する位相 TOD とした。非線形効果の補正は位相の応答 ( $\theta_{\text{res}}$ ) に対して以下の式 [13] を用いた。

$$\theta = 2 \tan(\theta_{\text{res}}/2) \quad (4.5)$$

最終的に使用する位相 TOD の例 (1 観測) を図 4.7 に示す。

## 4.2.3 必要な回転角

月の信号は大きく 1 観測の TOD で鮮明なマップを得ることができる。月を観測した時 (2023/12/02、1 時間) の位相 TOD を図 4.8 に示す。月は空を動いているが、望遠鏡が仰角を  $70^\circ$  に固定してスキャンする間に月が望遠鏡の視野を通り過ぎる時に観測できる。つまり、1 日の間に月が “昇る” 時と “沈む” 時とで 2 回の観測ができる。また、月の視野角は  $30'$  であり点源とはみなせない。そのため図 4.8 にあるように、月の端を観測する時と中心を観測する時では入射する信号の大きさが異なり、中心で最も強い信号を観測する。

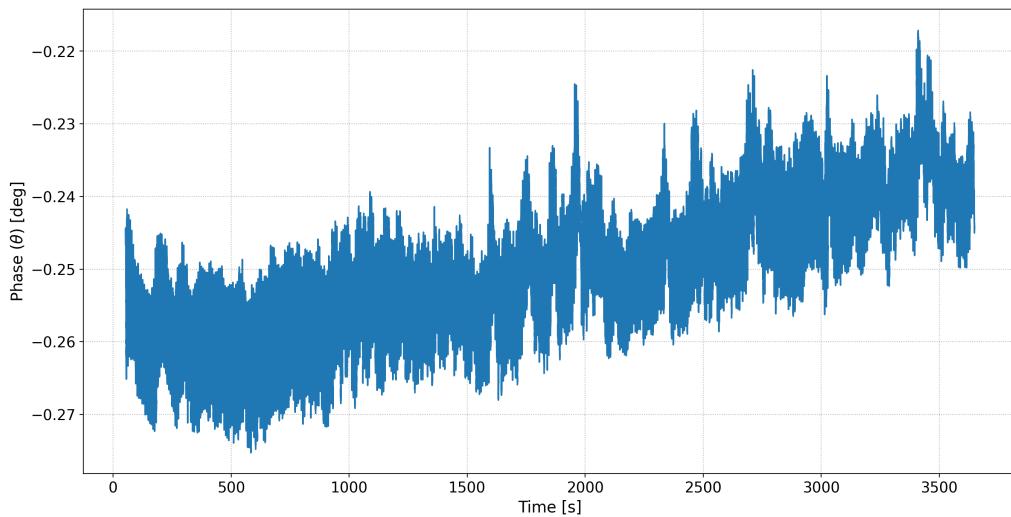


図 4.7: 非線形効果を加えた位相 TOD。本論文ではこの TOD を用いる。

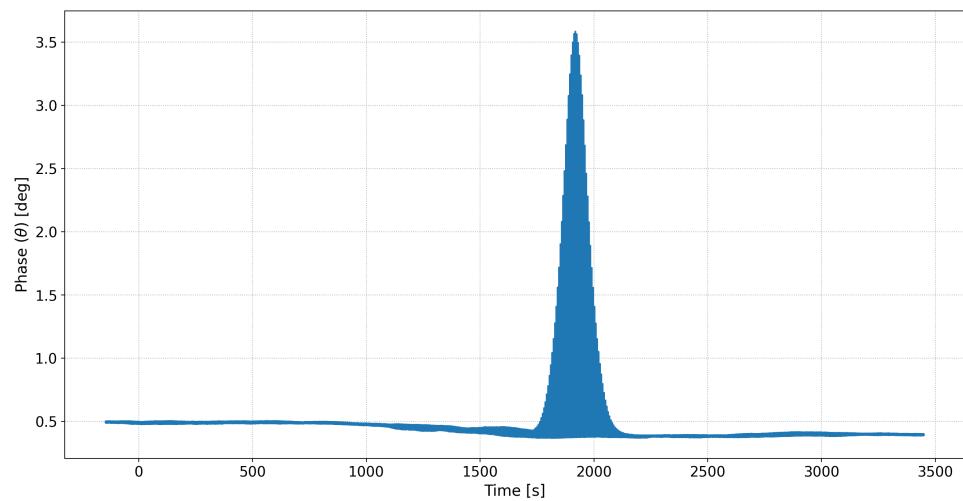


図 4.8: 月観測時の TOD。月の中心を観測した時にピークを取る。

月の運動はよく知られており、Python の “astropy[29]” パッケージを使うことで各時間での月の位置(仰角、方位角)を求めることができる。その情報と望遠鏡角度情報とのずれ(オフセット)を考慮することで TOD を月中心座標で表すことができ、月中心マップを構成することができる。220 GHz アレイの 1 観測から構成した各検出器の月中心マップを図 4.9 に示す。

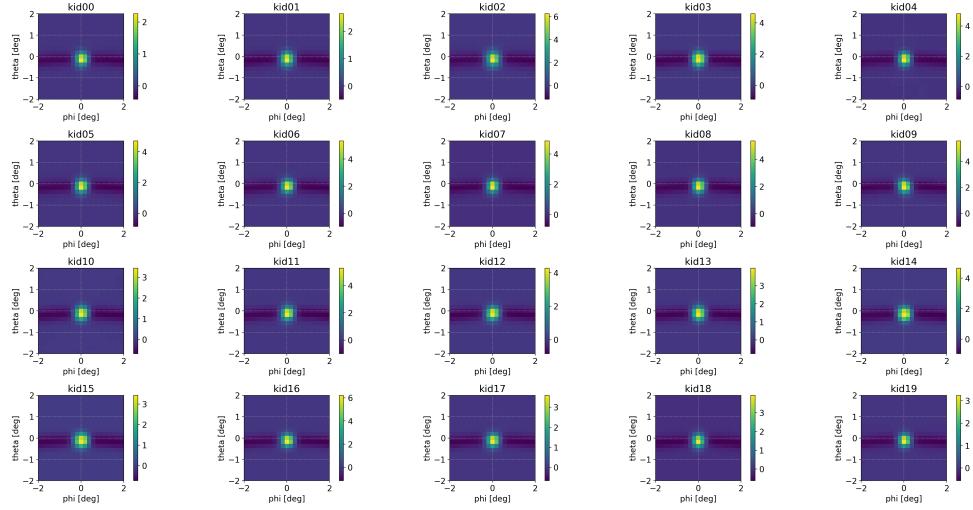
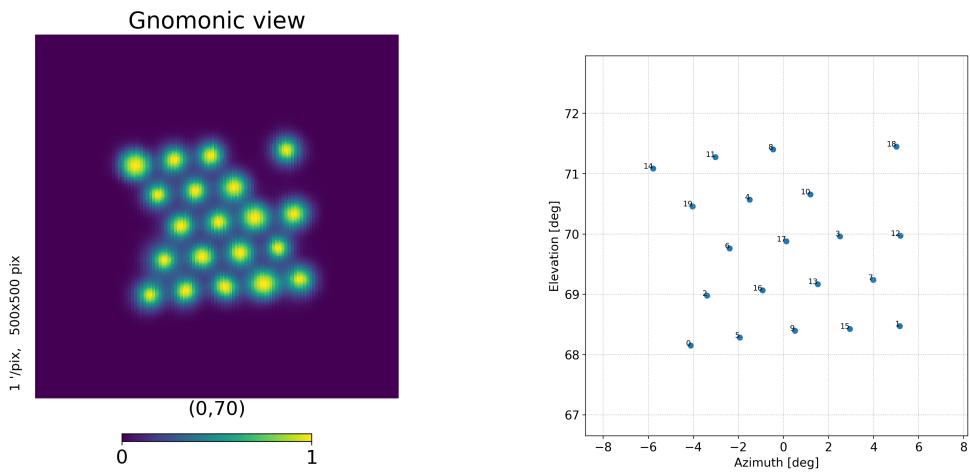


図 4.9: 各 MKID の月中心マップ。座標中心(月中心)で位相が高くなる。

次にこの月中心のマップから各検出器が見ている空(視線)の情報を取得する。月中心マップでは検出器を個別に見ていたが、検出器全体としての視線情報を得るには望遠鏡の視線中心、つまりビームを中心とした時にどの位置で月を観測したかを知る必要がある。そのため、220 GHz アレイの中心にある検出器(kid17 とラベルした)をビーム中心と考え、このビーム中心に対する月のマップを構成した。構成したマップの中で位相が最大となる位置を月の中心を見ていた位置として視線の代表点とする。220 GHz アレイのビーム中心マップと視線のプロットを図 4.10 に示す。ここで 4.10a では “healpy[30]” の gnomview を使って球



(a) 220 GHz アレイのビーム中心マップ。

(b) 各検出器のビーム中心の視線。

図 4.10: ビーム中心マップから取得した検出器の視線

面のマップを平面射影している。また、4.10b には検出器のラベルとして各点に番号を記し

た。この図から検出器の配置がスキャン軸に対して傾いていることが見て取れる。また、中心アレイでは仰角  $70^\circ$  でも球面による歪みが少なく平面的に見えている。

加えて中心以外の  $145\text{ GHz}$  アレイも含めた検出器全体でのビーム中心マップを見て傾きを確認した。TOD は同時刻帯の月を観測した時のものを使用した。使用データを表 4.2 に示す。ビーム中心マップと全検出器の視線は図 4.10 と同じ計算で求めた(図 4.11 と図 4.12)。

表 4.2: ビーム中心マップの構成に使用した月観測データ

観測日 (UTC)	観測時間 [min]	周波数 [GHz]	月の昇降 (rise or set)
2024/2/22 0:04 - 1:04	60	145	set
2024/2/22 0:17 - 1:17	60	145	set
2024/2/22 0:18 - 1:18	60	145	set
2024/2/22 0:37 - 1:37	60	220	set
2024/2/22 0:48 - 1:48	60	145	set
2024/2/22 0:48 - 1:48	60	145	set
2024/2/22 1:10 - 2:10	60	145	set

全検出器で視線を見ると、図 4.3 で示したような天球面での歪みの効果が周りのアレイに表れる。仰角が  $90^\circ$  に近づくにつれて検出器が見ている角度の間隔が広くなる。全体としてビーム中心に対して傾いていることが見て取れる。歪みの効果を補正するために仰角を  $70^\circ$  から  $0^\circ$  に天球面上で回転させた全検出器の視線を図 4.13 に示す。回転角を見積もるにあたってはスキャン軸に対する検出器の傾きを直線的に扱える方が考えやすい。そのため、平面的に見た時の傾き角を計算し、較正のための回転角とする。

回転角の算出には歪みの効果が少ない中心の  $220\text{ GHz}$  アレイのみを使用する。また、中心アレイはスキャン軸に対して 5 列で検出器が並んでいて理想的には全ての列が同じ角度で傾いていることになる。それを踏まえて以下の手順で回転角を求めた。

- 複数日での月観測データ ( $220\text{ GHz}$  アレイ) から検出器の視線を求める
- アレイ内で 5 列に並ぶ検出器のそれぞれの列で位置を直線フィット (データの性質が良くなかった 1 列目は省略した)
- フィットの傾きをその列での回転角としてヒストグラムに詰めて、ヒストグラムをガウシアンでフィット
- ガウシアン中心を最終的な回転角とする

角度決定における信頼性を高めるために複数の観測データを使用した。表 ?? に使用した  $220\text{ GHz}$  アレイの月観測データをまとめる。

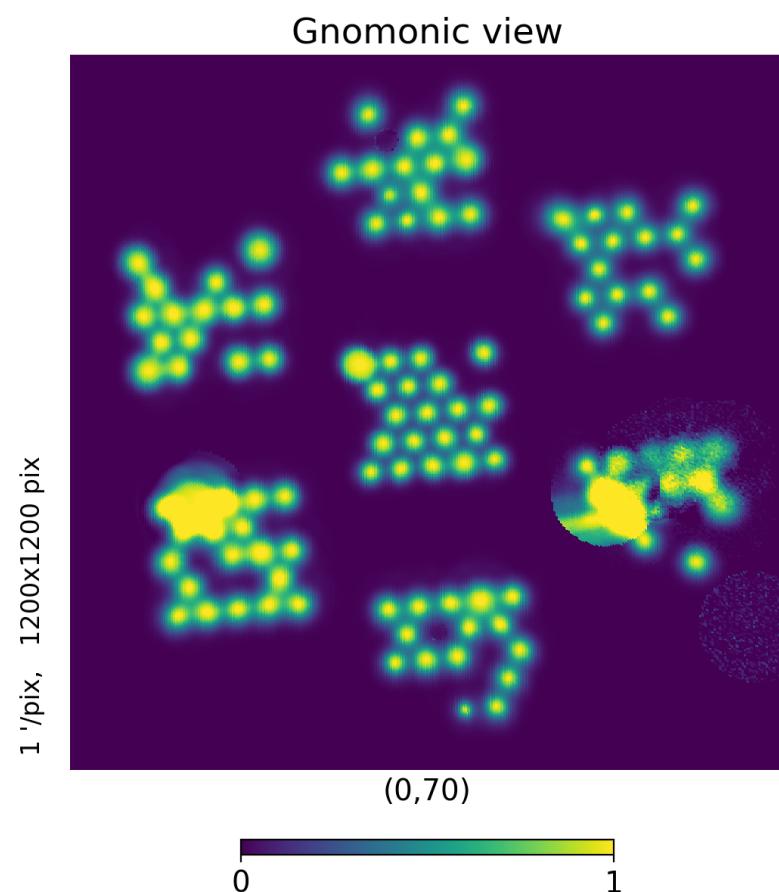


図 4.11: フルアレイのビーム中心マップ。

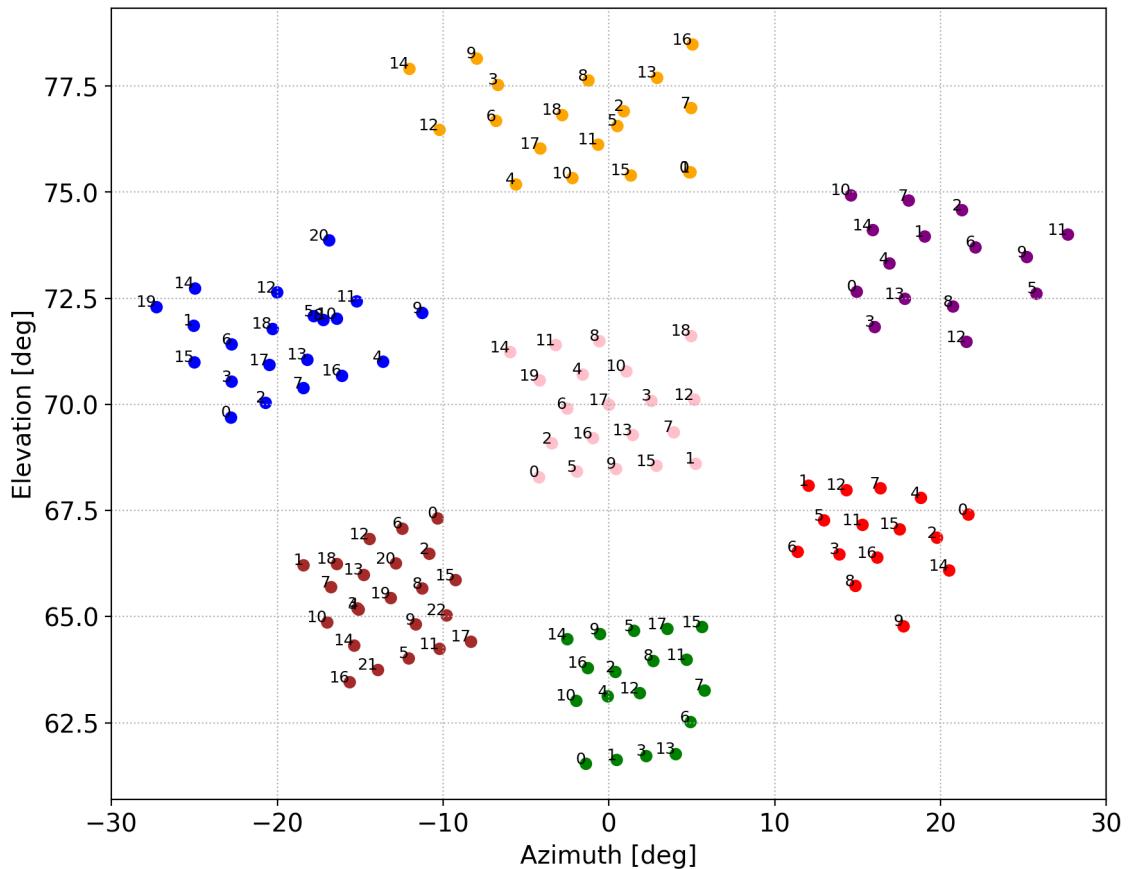


図 4.12: 全検出器のビーム中心の視線。

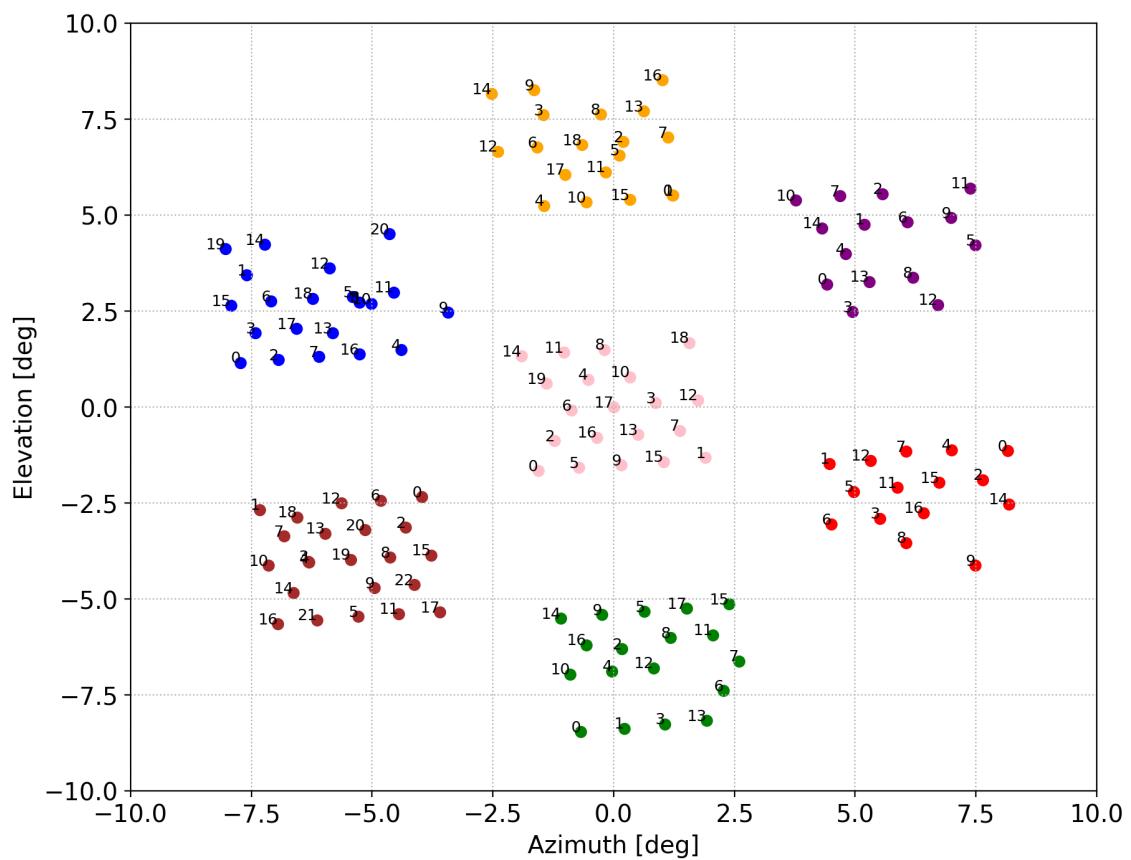


図 4.13: 座標回転により平面的に見たビーム中心の視線。

#### 4.2.4 回転する上でのジグの必要性

### 4.3 ジグの設計と現地インストール

#### 4.3.1 固定用ジグの作成

#### 4.3.2 望遠鏡への実装

### 4.4 天体を用いた較正結果の確認

#### 4.4.1 月データによる確認

#### 4.4.2 木星データによる確認

### 4.5 検出器間差分で見る大気揺らぎの抑制

#### 4.5.1 timing offset の算出

#### 4.5.2 差分解析による確認

## 第5章 今後の展望

- 5.1 大気揺らぎに由来するノイズのモデリング
- 5.2 両偏波アンテナを搭載した焦点面検出器のアップデート

## 第6章 まとめ

matome

## 第7章 謝辞

ありがとうありがとう

## 参考文献

- [1] Julio A. Castro-Almazán, Casiana Muñoz-Tuñón, Begoña García-Lorenzo, Gabriel Pérez-Jordán, Antonia M. Varela, and Ignacio Romero ” Precipitable Water Vapour at the Canarian Observatories (Teide and Roque de los Muchachos) from routine GPS” , Proc. SPIE 9910, Observatory Operations: Strategies, Processes, and Systems VI, 99100P (18 July 2016) <https://doi.org/10.1117/12.2232646>
- [2] P. A. R. Ade et al 2015 ApJ 812 176 <https://doi.org/10.1088/0004-637X/812/2/176>
- [3] P. K. Day, H. G. LeDuc, B. A. Mazin, A. Vayonakis, J. Zmuidzinas, Nature 425, 817-821 (2003) <https://doi.org/10.1038/nature02037>
- [4] R. M. J. Janssen, et al., High optical efficiency and photon noise limited sensitivity of microwave kinetic inductance detectors using phase readout. Appl. Phys. Lett. 103, 203503, 2013.
- [5] J. Bardeen, L. N. Cooper, and J. R. Schrieffer. Theory of superconductivity. Physical Review, 108:1175, 1957.
- [6] [https://lambda.gsfc.nasa.gov/education/graphic\\_history/taureionization.html](https://lambda.gsfc.nasa.gov/education/graphic_history/taureionization.html)
- [7] arXiv:2309.00675 [astro-ph.CO] <https://doi.org/10.48550/arXiv.2309.00675>
- [8] Monthly Notices of the Royal Astronomical Society, Volume 519, Issue 3, March 2023, Pages 3383—3431 <https://doi.org/10.1093/mnras/stac3439>
- [9] S.Honda, et al., On-site performance of GroundBIRD, a CMB polarization telescope for large angular scale observations. Proceedings Volume 11445, Ground-based and Airborne Telescopes VIII; 114457Q (2020)
- [10] Planck Collaboration, A&A 641, A4 (2020) <https://doi.org/10.1051/0004-6361/201833881>
- [11] K. Lee et al 2021 ApJ 915 88 <https://doi.org/10.3847/1538-4357/ac024b>
- [12] Y.Sueno, et al., Pointing Calibration of GroundBIRD Telescope Using Moon Observation Data
- [13] 末野慶徳. Development of calibration and noise characterization methods for a CMB telescope, GroundBIRD, using its commissioning observation data. 京都大学理学研究科 博士論文 2024.

- [14] J. Choi, GroundBIRD: A Telescope for the Cosmic Microwave Background Polarization Measurement, ph.D thesis, Korea University (2015).
- [15] <https://canon.jp/biz/product/indtech/incremental-encoder/lineup/r1sl>
- [16] <https://digilent.com/reference/programmable-logic/zybo/start?redirect=1>
- [17] <https://www.heidenhain.co.jp/製品/角度エンコーダ/組込み型角度エンコーダ/erm-2000 シリーズ>
- [18] <https://japan.xilinx.com/support/documentation-navigation/silicon-devices/mature-products/spartan-3e.html>
- [19] 池満拓司. CMB 望遠鏡のデータ読み出しシステムの時刻同期と較正に関する開発研究. 京都大学理学研究科 修士論文 2020.
- [20] <https://docs.amd.com/v/u/en-US/ds187-XC7Z010-XC7Z020-Data-Sheet>
- [21] [https://japan.xilinx.com/products/intellectual-property/axi\\_fifo.html](https://japan.xilinx.com/products/intellectual-property/axi_fifo.html)
- [22] <http://www.pynq.io>
- [23] <https://wasa-labo.com/wp/?p=1102>
- [24] <https://digilent.com/reference/programmable-logic/zybo-z7/reference-manual?redirect=1>
- [25] [http://www.kumikomi.net/fpga/sample/0008/FPGA08\\_042.pdf](http://www.kumikomi.net/fpga/sample/0008/FPGA08_042.pdf)
- [26] M. A. Janssen, Atmospheric Remote Sensing by Microwave Radiometry (1993)
- [27] 末野慶徳. 超伝導検出器 MKID の評価系構築と TLS ノイズを抑制する研究. 京都大学理学研究科 修士論文 2021.
- [28] 武藤優真. 超伝導検出器 MKID の薄膜純度向上及び高感度化を目指した製作と性能評価. 京都大学理学研究科 修士論文 2024.
- [29] <https://www.astropy.org>
- [30] <https://healpy.readthedocs.io/en/latest/>

## 付録A Zynqへのlinux搭載

時間あつたら書きたい