

# ダークフォトン探索に向けた広帯域分光計の開発と評価

京都大学大学院理学研究科 物理学・宇宙物理学専攻  
物理学第二分野 高エネルギー研究室  
竹内 広樹

2024年2月22日



## 概要

宇宙のエネルギー密度のうち、およそ 1/4 はダークマターと呼ばれる未知の物質が占めていると考えられている。ダークマターはその存在が確信されているものの、エネルギー密度を除いてほとんど性質がわかっておらず、ダークマターの正体解明は宇宙物理学、素粒子物理学双方で重要な課題である。近年、ダークフォトンと呼ばれる粒子が注目されている。特に、 $10\text{--}1000 \mu\text{eV}/c_0^2$  の質量を持つダークフォトンダークマター候補 (DP-CDM) として一部のインフレーションモデルや弦理論からの動機がある。ダークフォトンは電磁場と結合定数  $\chi$  でわずかに相互作用し、質量  $m_{\text{DP}}$  に比例した周波数の転換光子を放出するという性質を持つ。そのため、転換光をアンテナ等で受信し分光計で周波数ピークを探す実験方法が有効である。これまで、様々な質量領域でダークフォトンの探索が行われ、私の所属する研究グループでも既に  $10\text{--}26.5 \text{ GHz}$  帯 ( $41\text{--}110 \mu\text{eV}/c_0^2$ ) での探索を行ってきた。しかし、これらの実験では分光計の帯域幅がアンテナや RF 回路が対応している周波数領域よりも狭いため、分光計の帯域幅が実験の探索速度を制限している。探索速度の向上と統計量の増加による  $\chi$  の感度の向上のため、より広帯域を一度に分光できる分光計が求められる。

そこで、 $4 \text{ GHz}$  の帯域幅をデッドタイムなしに分光できる分光計を新たに開発した。周波数  $f$  の転換光の周波数ピークの幅  $\Delta f$  は典型的に  $\Delta f/f \sim 10^{-6}$  と極めて狭いため、このように非常に狭いピーク信号を検出できるよう  $16 \text{ kHz}$  の周波数分解能を確保した。実装においては RFSoc と呼ばれる ADC や CPU, FPGA などが 1 つのチップにまとめられた SoC を使い、FPGA 上に構築した FFT 回路に光処理を実装した。FFT 回路の設計にあたっては、16 並列の FFT 回路を利用することで  $4 \text{ GSPS}$  の ADC 入力を絶え間なく処理できるようにした。回路を並列化すると回路規模が増大してしまうが、回転演算回路の改良による省メモリ化や、回転演算回路の必要数を削減できるアーキテクチャの採用によって規模の増加を低減した。これらの設計の工夫により設計通りの動作クロックで FPGA への回路実装に成功し、実際に分光計を製作することができた。

さらに、製作した分光計について性能評価試験を行った。まず、単色 RF 源を入力して予想通りのスペクトルが得られているかを確認したほか、入力電力の大きさを変えて電力を測定し、 $0.5\%$  未満の精度で線形性を確認した。また、24 時間に渡って連続してデータ取得を続けられるか検証し、分光計の安定性を確認した。さらに、データ取得にかかった時間の計測と黒体輻射信号を測定する効率の評価から、分光計が実際にデッドタイムなしで分光できていることを確認した。これらの測定により、この分光計がダークフォトン探索に有用であることを実証した。

本実験で開発した分光計を使うことでダークフォトン探索の感度向上が見込まれる。例えば、先行実験から分光計を差し替えるだけで  $\chi$  に対する感度を 4 倍以上向上できる見込みである。さらに、今後探索を計画している  $170\text{--}260 \text{ GHz}$  帯においても、既存の実験よりも 2 桁程度良い感度で探索できる見込みである。なお、製作した分光計や FFT 回路に関しての特許出願も行なった。



# 目次

<b>第 1 章</b>	<b>はじめに</b>	<b>5</b>
1.1	ダークマターとダークフォトン	5
1.2	ディッシュ・アンテナ法	6
1.3	DP-CDM 探索の現状	9
1.3.1	DOSUE-RR 実験	9
1.3.2	これまでの探索結果と課題	11
1.4	本論文の構成	11
<b>第 2 章</b>	<b>設計戦略と実装手法の概略</b>	<b>13</b>
2.1	要求性能の見積もり	13
2.2	フーリエ分光計の概略	13
2.3	ハードウェア設計の概略	15
2.3.1	RFSoc	15
2.3.2	FPGA	15
2.4	ソフトウェア設計の概略	19
<b>第 3 章</b>	<b>各コンポーネントの設計と実装</b>	<b>21</b>
3.1	ハードウェア、FPGA 内回路の構成	21
3.2	アナログ回路 / ADC	23
3.3	FFT ブロック	25
3.4	アキュムレータブロック	27
3.5	FPGA への回路実装 (インプリメント)	28
3.6	ソフトウェア	28
3.7	開発した分光計の仕様	31
<b>第 4 章</b>	<b>FFT 回路の実装</b>	<b>33</b>
4.1	FFT 回路の実装方法と課題	33
4.1.1	Cooley-Tukey のアルゴリズム	34
4.1.2	パイプライン FFT の回路の実装	35

4.1.3	MDF アーキテクチャへの拡張	37
4.1.4	回路のリソース使用量と省リソース化の手法	38
4.2	作製した FFT 回路の構成	43
4.2.1	回転因子テーブルの分割による回転器の省メモリ化	44
4.2.2	Radix-2 <sup>4</sup> MDF アーキテクチャの採用	45
<b>第 5 章</b>	<b>性能評価</b>	<b>49</b>
5.1	単色 RF 源を用いた分光動作の実証	49
5.2	線形性	50
5.3	安定性と所要時間計測	52
5.4	時間効率	53
5.4.1	セットアップ・測定	54
5.4.2	結果	54
<b>第 6 章</b>	<b>DP-CDM 探索における感度見込み</b>	<b>59</b>
6.1	既存の実験セットアップでの感度の向上	60
6.2	高周波数帯の探索	60
<b>第 7 章</b>	<b>結論</b>	<b>63</b>
<b>付録 A</b>	<b>時間効率 <math>\epsilon</math> の推定方法</b>	<b>67</b>
	<b>参考文献</b>	<b>69</b>

# 第 1 章

## はじめに

### 1.1 ダークマターとダークフォトン

宇宙の全エネルギー密度のうち、26.8% はダークマターと呼ばれる未知の物質が占めている [1]。ダークマターの正体を解明することは、宇宙物理学、素粒子物理学の双方において重要な課題である。

ダークマターは宇宙における質量密度を除いてほとんど性質が分かっていないため、様々な質量領域に関心がある。近年、ダークマターの候補として WISP (Weakly Interacting Slim Particle) と呼ばれる粒子群が注目されている [2]。WISP は重力を除いて通常の物質とはわずかな相互作用しかしない、典型的に 1 eV 未満の軽量の粒子を指す。

このような WISP の一つとして、ダークフォトンと呼ばれる粒子が考えられている。ダークフォトンとは、素粒子標準模型に別の U(1) 対称性を追加することで現れる粒子であり、電磁場とわずかに相互作用する質量をもった光子のような粒子である。ダークフォトンのラグランジアンは、電磁場のラグランジアンに次のようなダークフォトン場の項、質量項、相互作用項を導入することによって与えられる。

$$\mathcal{L} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu} - \frac{1}{4}X_{\mu\nu}X^{\mu\nu} + \frac{m_{\text{DP}}^2}{2}X_\mu X^\mu - \frac{\chi}{2}F_{\mu\nu}X^{\mu\nu} - J^\mu A_\mu \quad (1.1)$$

ここで、 $A_\mu$ ,  $F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu$ ,  $J^\mu$  は、それぞれ通常の電磁場のポテンシャル、電磁場テンソル、電磁場カレントである。 $X_\mu$ ,  $X_{\mu\nu} = \partial_\mu X_\nu - \partial_\nu X_\mu$  はそれらに対応するダークフォトンの項である。 $m_{\text{DP}}$  はダークフォトンの質量であり、 $\chi$  はダークフォトン場と電磁場の相互作用の強さを表す結合定数である。

$X_\mu \rightarrow \tilde{X}_\mu - \chi A_\mu$  と書き直すと、

$$\mathcal{L} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu} - \frac{1}{4}\tilde{X}_{\mu\nu}\tilde{X}^{\mu\nu} + \frac{m_{\text{DP}}^2}{2}\left(\tilde{X}_\mu\tilde{X}^\mu - 2\underline{\chi}A_\mu\tilde{X}^\mu + \chi^2 A_\mu A^\mu\right) - J^\mu A_\mu \quad (1.2)$$

と  $F_{\mu\nu}X^{\mu\nu}$  の項を消去できる。式 (1.2) から、通常の電磁場  $A_\mu$  とダークフォトン場  $\tilde{X}_\mu$  が、 $m_{\text{DP}} \neq 0$  かつ  $\chi \neq 0$  であるとき第三項を通じて混合することが分かる。つまり、ダークフォトンは通常光子に転換する性質を持つ。

ダークマターとしてのダークフォトンの質量は様々な質量が考えられるが、特に  $\sim 10\text{--}1000 \mu\text{eV}$  の質量領域は、一部の弦理論やインフレーション理論に支持されている [2, 3]。

## 1.2 ディッシュ・アンテナ法

このようなダークマターとしてのダークフォトン (DP-CDM) の分光探索の手法として、ディッシュ・アンテナ法と呼ばれる探索手法が提案されている [4]。本節でディッシュ・アンテナ法の原理を説明する。

まず、式 (1.2) から周波数  $\omega$  で運動量  $k$  の平面波解は次の運動方程式に従う：

$$\left[ (\omega^2 - k^2) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - m_{\text{DP}}^2 \begin{pmatrix} \chi^2 & -\chi \\ -\chi & 1 \end{pmatrix} \right] \begin{pmatrix} \mathbf{A} \\ \mathbf{X} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (1.3)$$

この解は

$$\begin{pmatrix} \mathbf{A} \\ \mathbf{X} \end{pmatrix} = \mathbf{X}' \begin{pmatrix} 1 \\ \chi \end{pmatrix} \exp(-i(\omega t - \mathbf{k} \cdot \mathbf{x})) \quad (\omega = k) \quad (1.4)$$

$$\begin{pmatrix} \mathbf{A} \\ \mathbf{X} \end{pmatrix} = \mathbf{X}' \begin{pmatrix} -\chi \\ 1 \end{pmatrix} \exp(-i(\omega t - \mathbf{k} \cdot \mathbf{x})) \quad (\omega = \sqrt{m_{\text{DP}}^2 + k^2}) \quad (1.5)$$

の二つが考えられるが、 $\omega = \sqrt{m_{\text{DP}}^2 + k^2}$  のときの解が DP-CDM に対応する。DP-CDM は非相対論的であることに注意すると  $\omega = \sqrt{m_{\text{DP}}^2 + k^2} \sim m_{\text{DP}}$  である。以下のように、式 (1.5) から DP-CDM はごくわずかに通常の電場成分を持っている。

$$\begin{aligned} \mathbf{E}_{\text{DP}} &= -\partial_0 \mathbf{A}_{\text{DP}} \\ &= i\chi\omega \mathbf{X}'_{\text{DP}} \exp(-i(\omega t - \mathbf{k}_{\text{DP}} \cdot \mathbf{x})) \\ &\sim i\chi m_{\text{DP}} \mathbf{X}'_{\text{DP}} \exp(-i(\omega t - \mathbf{k}_{\text{DP}} \cdot \mathbf{x})) \end{aligned} \quad (1.6)$$

ここで、空間中に金属板を配置することを考える。簡単のため、金属板は平坦な板とする。金属板表面では、境界条件として  $\mathbf{E}_{\parallel} = 0$  が課せられるので、金属板内外を DP-CDM が通過する際に、式 (1.4) と式 (1.5) の線型結合によって電場を打ち消す必要がある。つまり、金属板を設置すると、表面から転換光が放出される。

転換光の電場は

$$\begin{aligned} \mathbf{E}_{\text{CP}} &= -\partial_0 \mathbf{A}_{\text{CP}} \\ &= -ik \mathbf{X}'_{\text{CP}} \exp(-i(\omega t - \mathbf{k}_{\text{CP}} \cdot \mathbf{x})) \end{aligned} \quad (1.7)$$

と表せるので式 (1.6) と合わせて、金属板の至る所で  $\mathbf{E}_{\parallel} = 0$  となるためには  $\omega$  が DP-CDM と転換光で共通、すなわち

$$k_{\text{CP}} = \sqrt{m_{\text{DP}}^2 + k_{\text{DP}}^2} \quad (1.8)$$

であり、さらに金属板上で

$$\mathbf{k}_{\text{CP}} \cdot \mathbf{x} = \mathbf{k}_{\text{DP}} \cdot \mathbf{x} \quad (1.9)$$

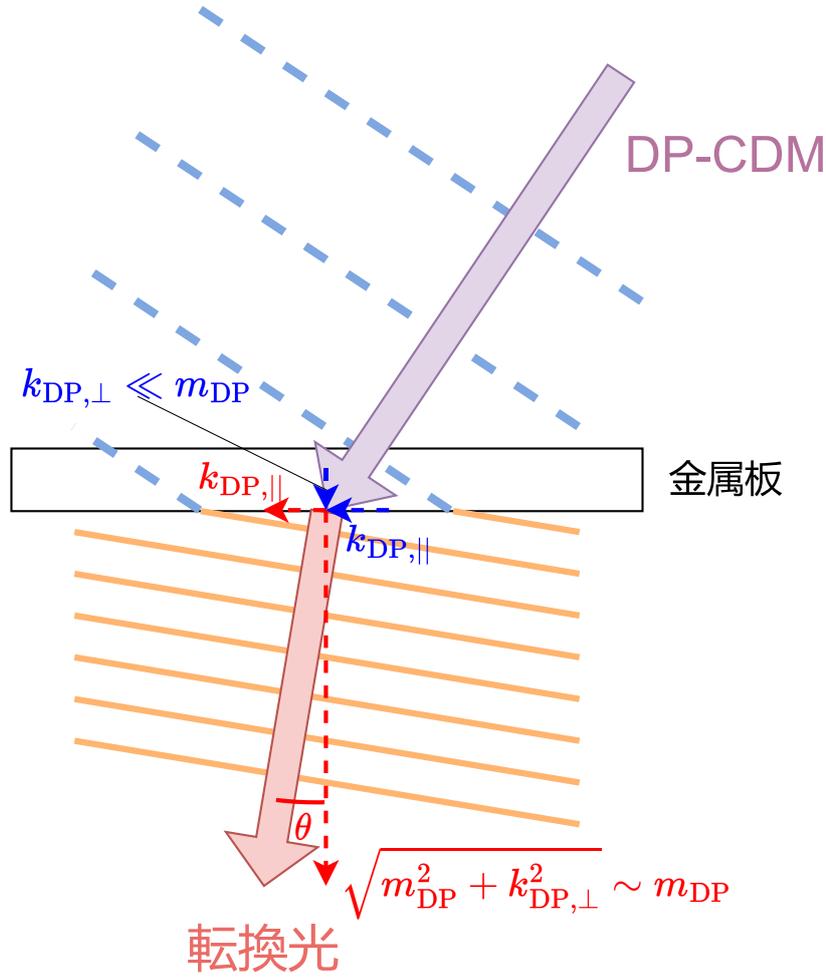


図 1.1: 金属板表面で轉換光が放出されるときの様式図。非相対論的な DP-CDM の金属板方向の運動量成分を保存したまま  $m_{DP}$  の質量エネルギーが全て運動エネルギーに変えられるため、ほぼ垂直に ( $\theta \leq 0.06^\circ$ ) 放出される。

が成り立つ必要がある。 $\mathbf{n}$  を金属板に垂直な単位ベクトルとすると、式 (1.8), (1.9) から、

$$\mathbf{k}_{CP} = \sqrt{m_{DP}^2 + \mathbf{k}_{DP,\perp}^2} \mathbf{n} + \mathbf{k}_{DP,\parallel} \quad (1.10)$$

と表せる。ただし、 $\mathbf{k}_{DP,\perp}$ ,  $\mathbf{k}_{DP,\parallel}$  はそれぞれ  $\mathbf{k}_{DP}$  の金属板に垂直、水平な成分である。DP-CDM は非相対論的であるので、轉換光と  $\mathbf{n}$  のなす角  $\theta$  は  $\theta \sim |\mathbf{k}_{DP,\parallel}|/m$  と、ほぼ垂直に放出される [5] (図 1.1)。

このような光の轉換原理を使えば、図 1.2 のようにアンテナの前に球面鏡や平面板を置くことで、轉換光を発生させてアンテナによって受信できる。アンテナが受信する DP-CDM の轉換光の

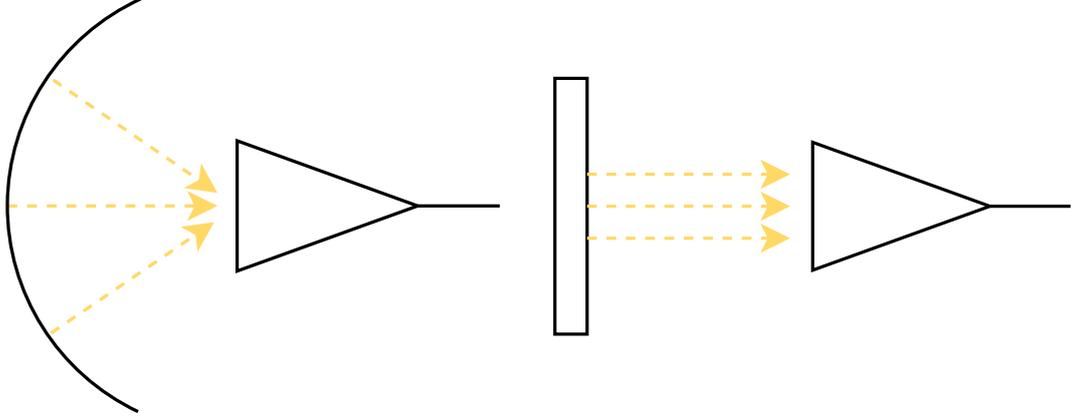


図 1.2: ディッシュ・アンテナ法では金属表面で生じた転換光をアンテナで受信してダークフォトン (DP-CDM) の分光探索を行う。金属表面には球面鏡や金属平板などを用いる。

強度は、

$$P_{\text{DP}} = (6.4 \times 10^{-2} \text{ aW}) \times \left( \frac{\chi}{10^{-10}} \right)^2 \left( \frac{A_{\text{eff}}}{10 \text{ cm}^2} \right) \times \left( \frac{\rho}{0.39 \text{ GeV/cm}^3} \right) \left( \frac{\alpha}{\sqrt{2/3}} \right)^2 \quad (1.11)$$

と与えられる。ここで、 $\rho$  は銀河ハロー内のダークマターのエネルギー密度で、 $\rho = (0.39 \pm 0.03) \text{ GeV/cm}^3$  と推定されている [6]。 $A_{\text{eff}}$  はアンテナが受信できる金属表面の有効面積を表す。 $\alpha$  は DP-CDM の偏極方向に対応した係数で、DP-CDM の偏極がランダムで、片偏波のアンテナで受信する場合、 $\alpha = \sqrt{1/3}$  である。

また、エネルギー保存から、速度  $v_{\text{DP}}$  の DP-CDM の転換光の周波数  $\nu$  は、

$$\nu(v_{\text{DP}}, m_{\text{DP}}) = \frac{1}{h} \frac{m_{\text{DPC}}^2}{\sqrt{1 - (v_{\text{DP}}/c)^2}} \quad (1.12)$$

となる。

DP-CDM の運動量分布はマクスウェル・ボルツマン分布であると仮定することが一般的である：

$$f(\mathbf{v}, \mathbf{v}_E) = \frac{1}{(\pi v_c)^{3/2}} \exp\left(-\frac{|\mathbf{v} + \mathbf{v}_E|^2}{v_c^2}\right) \quad (1.13)$$

ここで、 $v_c$  は銀河の回転速度、 $\mathbf{v}_E$  は銀河に対する地球の速度を表す。式 (1.12), (1.13) から、転換光の周波数分布は  $\nu_0 = m_{\text{DPC}}^2/h$  付近に半値幅  $\Delta\nu/\nu_0 \sim 10^{-6}$  の鋭いピークを持つ。例えば、 $v_c, \mathbf{v}_E$  を典型値の  $220 \text{ km/s}$  とし、 $\nu_0 = 20 \text{ GHz}$  ( $m_{\text{DP}} = 82.7 \mu\text{eV}$ ),  $\chi = 8.0 \times 10^{-10}$ ,  $A_{\text{eff}} = 1.5 \text{ cm}^2$  すると、転換光のパワースペクトルは図 1.3 のようになる。

以上より、金属板から放出された転換光をアンテナで受信して分光し、この周波数ピークを熱ノイズの中から見つけ出すことで DP-CDM の探索が可能である。そして、転換光のピーク周波数から  $m_{\text{DP}}$  が求まり、転換光の強度から  $\chi$  が求まる。

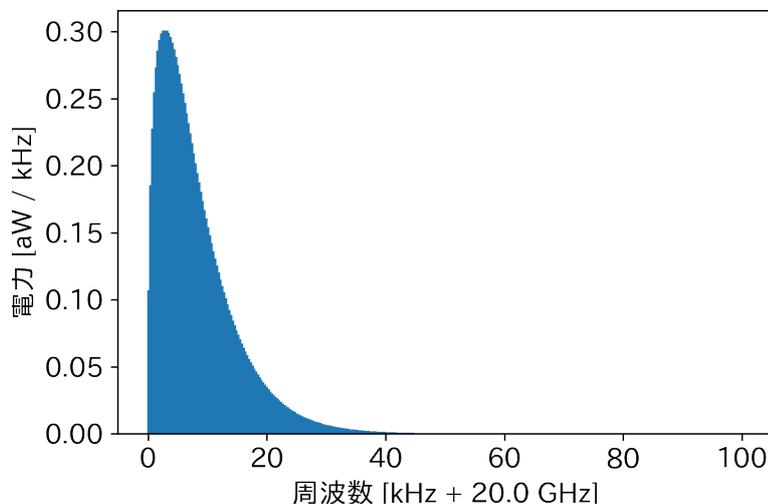


図 1.3:  $\nu_0 = 20$  GHz ( $m_{\text{DP}} = 82.7 \mu\text{eV}$ ),  $\chi = 8.0 \times 10^{-10}$ ,  $A_{\text{eff}} = 1.5 \text{ cm}^2$  としたときに見込まれる転換光のパワースペクトル。

### 1.3 DP-CDM 探索の現状

これまで、ディッシュ・アンテナ法による直接探索の他、Haloscope と呼ばれる共振空洞や、液体キセノン TPC を用いた直接探索、あるいは宇宙論や太陽寿命などによる間接的な制限など、さまざまな DP-CDM 探索が行われてきた。図 1.4 にこれまでの DP-CDM 探索の結果を示す。

色付きの領域は、その実験によって排除された質量  $m_{\text{DP}}$  と結合定数  $\chi$  の組み合わせを示している。図 1.4 を見ると、特に 0.1–1 meV の質量領域では、直接探索がほとんどなされておらず、間接的な制限も弱いことが分かる、この領域で、高感度かつできるだけ広い帯域で直接探索を行うことには大きな意義がある。

#### 1.3.1 DOSUE-RR 実験

この意義深い質量領域を策る実験が、我々が行っている、DOSUE-RR (Dark-photon Observing System for Un-Explored Radio-Range) 実験である。図 1.5 に概略を示すように、DOSUE-RR 実験はディッシュ・アンテナ方式のダークフォトン探索実験である。

DOSUE-RR 実験では、オリジナルの提案 [4] にあった球面鏡を用いて集光する代わりに、金属平板と通常のアンテナの組み合わせ [8, 9] で探索を行う。平面波を集光するアンテナは入手しやすく、また球面鏡に比べて較正もたやすい。アンテナはホーンアンテナを利用し、アンテナが受信した信号は低ノイズアンプ (LNA) やミキサなどで構成される RF 回路によって増幅やダウンコンバートといった処理を行われ、その後分光される。

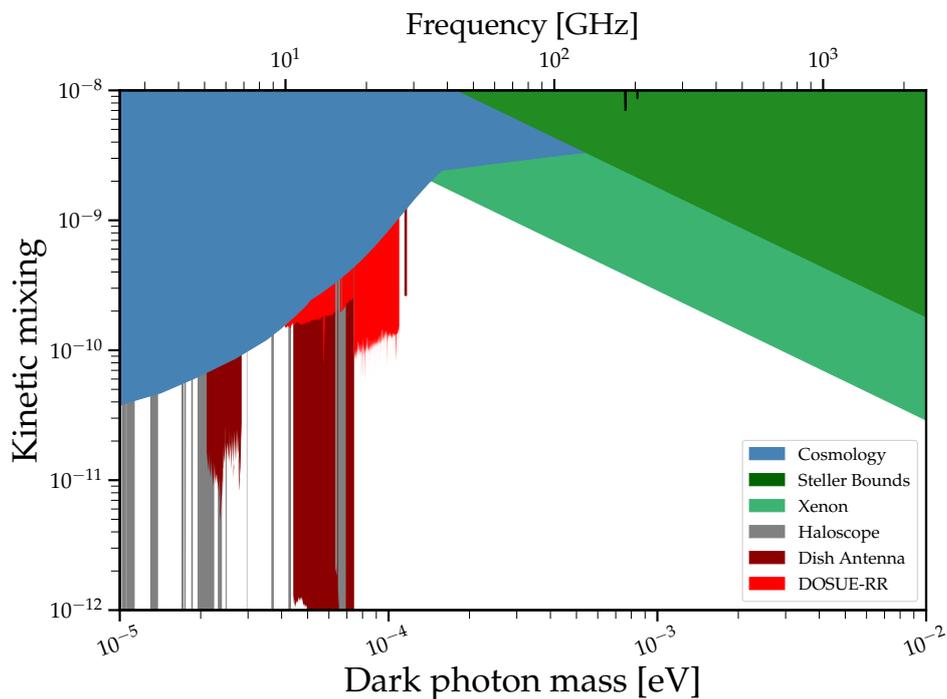


図 1.4: DP-CDM 探索の現状 [7]。色の塗られた領域は探索によって排除されていることを示す。特に、濃赤と赤色の領域はディッシュ・アンテナ法によって新たに探索されたダークフォトンのパラメータ領域である。

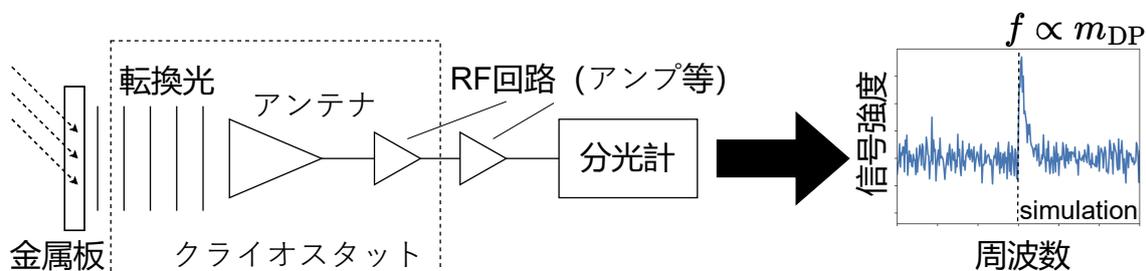


図 1.5: DOSUE-RR 実験のセットアップの概要。金属板で生じた轉換光をホーンアンテナで集光し、RF 回路で増幅などを行った後、分光を行う。

$\chi$  の感度を決める主要なノイズは LNA の熱雑音となる。熱雑音を削減するため、[10] の探索ではアンテナや RF 回路をクライオスタットの中に入れ 3K まで冷却した。また、外来電波が混入して信号と混同されるのを防ぐため、[11] の探索ではコンパクトな電波暗室を製作して遮蔽した。

### 1.3.2 これまでの探索結果と課題

これまで、DOSUE-RR 実験グループでは、図 1.4 のように今まで直接探索がなされていなかった質量領域での、間接探索を超える感度の DP-CDM 探索に成功している。具体的には、転換光の周波数に対して 10–18 GHz 帯 ( $m_{\text{DP}} = 41\text{--}74 \mu\text{eV}$ ) [11] 及び、18–26.5 GHz 帯 ( $m_{\text{DP}} = 74\text{--}110 \mu\text{eV}$ ) [10] の探索を世界最高精度で達成してきた。

一方で、先行実験 [11, 10] では帯域幅に課題がある。これらの探索 [11, 10] では分光計として市販のシグナルアナライザを利用しており、一度に探索できる帯域はわずか 2 MHz であった。一方、アンテナや RF 回路は数 GHz 幅の広い周波数帯に対応しており、原理上はこの範囲を同時に探索することができる。つまり、分光計として用いたシグナルアナライザの狭い帯域幅が、探索実験にかかる時間を律速していた。例えば、先行実験 [10] では 18–26.5 GHz 帯で計 8.5 GHz 幅の探索を行ったが、合計 4250 回もの測定を必要とし、測定に 2 週間も要した。つまり、分光計の帯域幅の拡大によって、大幅に探索速度を向上する余地がある。

また、100 GHz を超える高周波での探索を行う際には、現状の狭い帯域幅がより致命的な影響を及ぼす。例えば、10–20 GHz 帯を探索するには 10 GHz 幅が良いが、100–200 GHz 帯では 10 倍の 100 GHz 幅の探索が必要になる。単純計算で従来の 10 倍の時間が必要になり、実験の遂行が極めて困難になる。

より効率的な探索のためには、数 GHz の広帯域を同時に分光できる分光計が求められる。広帯域分光計を開発する試みとしては、これまでもフーリエ変換を GPU や FPGA 上で行うフーリエ式分光計が製作されてきた。例えば、[9] では GPU を利用して 4 GHz の帯域幅を 60% の時間効率で分光している。その他、FPGA ベースで 2.5 GHz 幅のデッドタイムなしの分光計が販売されているなど、これまで同時に 2–3 GHz 幅を分光可能な分光計が開発されてきた。

本研究では、これらの分光計よりもさらに広い同時帯域幅を実現するため、FPGA を利用したフーリエ式分光計を開発し、性能評価を行った。さらに、開発した分光計を用いることで、DP-CDM 探索で実現できる感度を見積もった。

## 1.4 本論文の構成

本論文の構成について述べる、まず分光計の設計について、第 2 章から第 4 章で説明する。第 2 章では性能要件や大まかな設計戦略を説明し、第 3 章で各コンポーネントの設計について詳述する。第 4 章では、分光計の中核となる FFT 演算回路についてさらに説明する。第 5 章では、製作した分光計の性能評価について述べる。第 6 章では、DP-CDM 探索において、この分光計を用いることで見込まれる感度について議論する。第 7 章は本論文の結論である。



## 第 2 章

# 設計戦略と実装手法の概略

### 2.1 要求性能の見積もり

まず、分光計を設計するうえで必要な性能を考える。探索効率に直結するのは「同時帯域幅 (Real-time Bandwidth)」、あるいは「瞬時帯域幅 (Instantaneous Bandwidth)」と呼ばれる指標である。これは一度の測定で、中心周波数の周りにどれだけの周波数幅のデータを取得できるかを表す。後述するように同時帯域幅の上限はアナログ信号をデジタル信号に変換する ADC のサンプリングレートによって決まる。現状容易に取り扱えるサンプリングレートは数 GHz 程度であるため、本研究ではこのサンプリングレートを最大限活用した数 GHz 幅の同時帯域幅をもつ分光計の製作を目指す。また、時間効率を最大化するために、デッドタイムなしでの分光が求められる。

次に周波数分解能であるが、周波数ピークを探すためには少なくとも転換光の周波数幅以下、できれば一桁以上細かい分解能を持つことが好ましい。周波数  $f$  の転換光のピークの典型的な周波数幅は 1.2 節で述べたように  $\Delta f \sim 10^{-6} \cdot f$  なので、100 GHz 帯を探索する場合は 100 kHz の周波数分解能、10 GHz 帯を探索する場合は 10 kHz の周波数分解能が求められる。サンプリングレートを 4 GHz と仮定すると、10 kHz の分解能を実現するためには帯域内を  $4 \times 10^5$  点に分割する必要があるが、第 4 章で述べるように、分割点数が増えるほどフーリエ変換を行う回路の規模が増大してしまう。回路規模を極限まで削減するアルゴリズムの開発が求められる。

### 2.2 フーリエ式分光計の概略

一度に広い周波数帯の電波を分光する場合、フーリエ変換を用いることが有効であり、このようなフーリエ式の分光計は FFT アナライザやシグナルアナライザなどと呼ばれ広く普及している。前節の要求性能を満たすため、フーリエ式の分光計を製作する。

図 2.1 にフーリエ式の分光計の典型的な構成を示す。まず、入力信号はミキサと呼ばれる回路によってダウンコンバートされる。ミキサは、入力信号  $x(t)$  と基準信号  $l(t)$  を入力して積  $o(t) = x(t) \times l(t)$  を出力するような回路である。ミキサに  $x(t) = x \cos(2\pi ft + \delta)$ ,  $l(t) = \cos(2\pi f_l t)$  の

ような信号を入力すると、

$$o(t) = x \cos(2\pi f t + \delta) \cos 2\pi f_i t \quad (2.1)$$

$$= \frac{1}{2} x \cos \left[ 2\pi \underbrace{(f - f_i)} t + \delta \right] - \frac{1}{2} x \cos \left[ 2\pi \underbrace{(f + f_i)} t + \delta \right] \quad (2.2)$$

という出力が得られる。つまり、ミキサを使用することで、入力信号と基準信号の周波数の差（または和）の周波数の信号にダウン（アップ）コンバートすることができる。ミキサに入力する（単一の周波数の）基準信号を生成する装置は局部発振器 (Local Oscillator, LO) と呼ばれ、また、生成される基準信号は LO 信号と呼ばれる。適切なミキサと LO 信号を用意することで、分光計の帯域が  $0 \sim \Delta f$  であっても、任意の周波数  $f$  に対して、 $f \sim f + \Delta f$  の範囲を分光することができる。

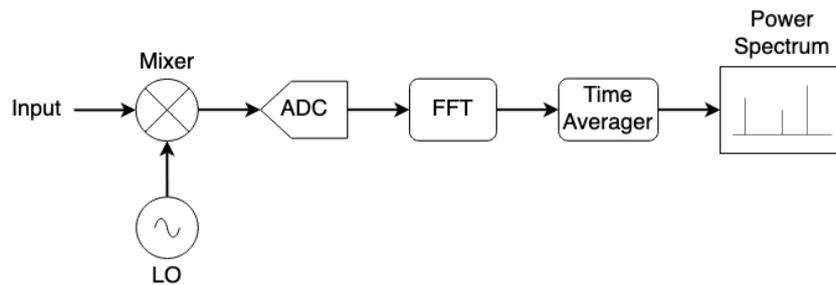


図 2.1: フーリエ式の分光計の典型的な構成。まず、入力信号と局部発振器 (LO) が生成する信号をミキサに入力し、信号をダウンコンバートする。ダウンコンバートした入力信号を ADC でデジタル信号に変換し、高速フーリエ変換 (FFT) によってスペクトルを求める。得られた信号のデータ量が多すぎる場合、時間平均をとるなどしてデータを削減してから外部に転送する。

ミキサによってダウンコンバートされた信号は、次に ADC によってデジタル信号に変換され、さらにフーリエ変換によって周波数情報に変換される。ナイキストの定理から、サンプリングレート  $S$  SPS (Samples Per Second, 1 秒間あたりのサンプリング数) の ADC を使った場合の帯域幅は、 $x(t)$  が実信号、すなわち  $x(t) = x \cos(2\pi f t + \delta)$  と書けるときの  $S/2$  Hz である。これは、LO より高い周波数の信号と低い周波数の信号が足し合わされてしまい区別できないためである。

この問題を回避するため、IQ ミキサが用いられる。IQ ミキサではミキサを 2 つ用意し、片方のミキサにはそのままの LO 信号を、もう片方には  $\pi/2$  ずらした LO 信号を入れることで元の信号の位相情報を保持する。IQ ミキサによる変換は数式としては

$$x \cos(2\pi f t + \delta) \mapsto \frac{1}{2} (x \cos [2\pi(f - f_i)t + \delta], x \sin [2\pi(f - f_i)t + \delta]) \quad (2.3)$$

$$- \frac{1}{2} (x \cos [2\pi(f + f_i)t + \delta], -x \sin [2\pi(f + f_i)t + \delta]) \quad (2.4)$$

と表せる。第一成分がそのままの LO 信号を入れた時の出力、第二成分が  $\pi/2$  ずらしたときの出力である。この信号を 2 つの ADC を用いて同時にサンプリングして複素フーリエ変換を行えば、LO より高い周波数 (Upper Side Band) の信号と LO より低い周波数 (Lower Side Band) の信号を区別することができる。これにより、帯域幅  $S$  Hz での分光が可能になる。

フーリエ式の分光計ではフーリエ変換を高速に行う必要があるため、通常後述の高速フーリエ変換 (FFT) を行う専用の演算回路によって計算される。そして、広帯域のフーリエ式分光計を実現するためには、できるだけ高いサンプリングレートの ADC と、ADC から出力される高いレートのデータを処理できる、高性能な FFT 回路が必要となる。

## 2.3 ハードウェア設計の概略

本節では 2.2 節で概説したフーリエ式分光計を実装するためのハードウェア RFSoc について述べる。また、RFSoc を活用するためのファームウェア設計手法についても俯瞰する。

### 2.3.1 RFSoc

2.1 節の要件を満たすためには、数 GSPS の高速 ADC と、それを処理できる高性能な FPGA が必要である。そこで、本研究ではこの二つを兼ね備えた、RFSoc と呼ばれる AMD Xilinx 製のチップを用いる。RFSoc は AMD Xilinx が開発した、CPU、FPGA、さらに ADC や DAC といった RF 回路などが 1 チップに統合された System on Chip (SoC) である (図 2.2)。また、CPU を搭載する Processing System (PS)、FPGA を搭載する Programmable Logic (PL) のそれぞれが RFSoc 外部のメモリチップ (DRAM) と接続されている。そのほかにも、イーサネットのような豊富な外部 IO 接続を PS, PL 共に持っている。

RFSoc は 5G 通信のような高速無線通信を主要な用途として商業的に生産されているため、FPGA や高速 ADC・DAC を搭載するにもかかわらず、比較的安価に入手可能である。また、量産品であることもあって開発フレームワークなどが整備されている。例えば、後述する Pynq と呼ばれるフレームワークによって Ubuntu や Jupyter を用いた開発が行えるなど、カスタムの SoC やボードを用いた場合と比べて容易に開発を行えるメリットもある。

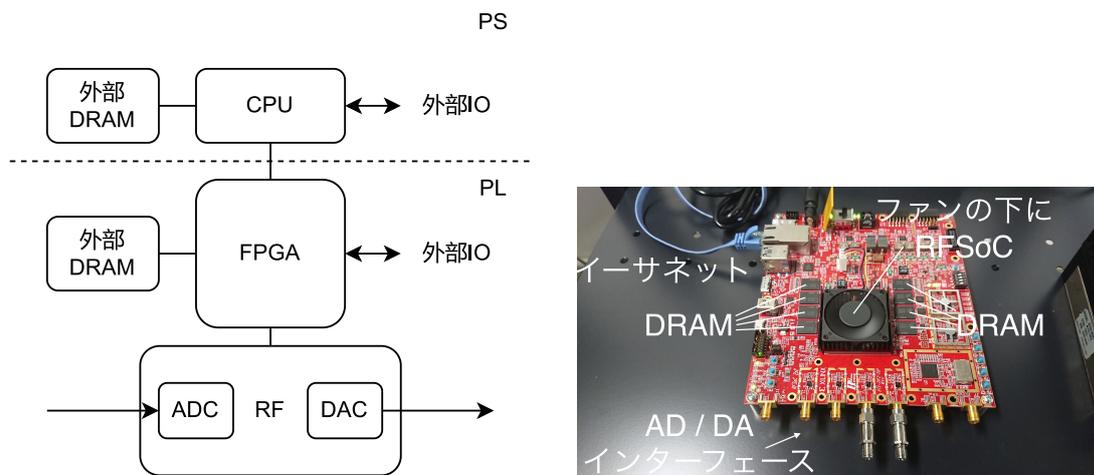
### 2.3.2 FPGA

FPGA は Field Programmable Gate Array の略で、様々な基本的な論理回路やメモリがチップ上に大量に配置されており、(製造後に) ユーザーが自由に配線して論理回路を作製できるデバイスである。FPGA では特定の演算に特化した専用回路を構築できるため、CPU のような汎用プロセッサを用いる場合よりも高速に処理できることがある。本研究では RFSoc に内蔵された FPGA 上に FFT 回路を実装し、内蔵 ADC からの入力信号を処理することで、単一のチップ上に分光計を実装する。

#### 各種リソース

FPGA 上にはユーザーが回路を構築する上での基本パーツとなる、多様な論理回路が搭載されている。本節では FPGA に搭載されている基本的な回路について説明する。

FPGA が提供する最も基本的なパーツは、Look Up Table (LUT) と Flip Flop (FF) である。



(a) AMD Xilinx RFSoc のブロックダイアグラム。CPU、FPGA、ADC / DAC などが 1 枚のチップ上に統合された SoC になっている。また、CPU を搭載する Processing System (PS)、FPGA を搭載する Programmable Logic (PL) のそれぞれが DRAM や、イーサネットなどの外部 IO を持っている。

(b) RFSoc を搭載したボードの例 (RFSoc 2x2 Board)。黒いファンの下に RFSoc が搭載されている。ファンの両脇にある 8 つのチップがメモリチップ (DRAM) である。手前の SMA コネクタを通じて RFSoc 内の ADC や DAC と入出力を行える。例えば、ローパスフィルタが接続されている 2 つのコネクタは ADC へつながっている。

図 2.2: AMD Xilinx RFSoc のブロックダイアグラム (左) と実際に RFSoc を搭載したボードの例 (右)。

LUT は複数の入力値の組み合わせに対して、どのような出力を返すかを記憶するテーブルを備えた回路であり、AND 回路や NOT 回路、XOR 回路のような、組み合わせ回路を実装する基本パーツである。一方、FF はひとつ前のクロックで入力された値を次のクロックの間出力し続けるレジスタ回路で、順序回路を実装するための基本パーツである。FPGA には FF と LUT、そしてそれらを結ぶ配線が大量に用意されており、この二つを組み合わせることで、自在に論理回路を作ることができる。

しかし、LUT や FF のような汎用回路は汎用性を確保するために冗長な実装になっており、よく使う回路については専用の回路ブロックをチップ上に用意し、それを使った方が効率的な場合がある。そのため、実際に販売されている FPGA では様々な専用回路ブロックが搭載されている。

そのような専用回路の一つに、四則演算などの基本的な算術演算を行う回路がある。RFSoc では DSP (Digital Signal Processing) ブロックと呼ばれており、加減算回路や乗算回路などが一つの DSP ブロックの中に搭載されている。算術演算ではできるだけ DSP ブロックを活用することで、あまりリソースを消費せずに回路を実装できる。

また、値を記録するメモリも、専用回路が用意されている。先述の LUT でも値を記憶できるが、大量の値を保存するには適していないためである。RFSoc の場合、Block RAM (BRAM)

と、Ultra RAM (URAM) という 2 種類の大量の値の保存に適したメモリブロックが用意されている。BRAM は FPGA 内に分散して配置されたメモリブロックである。FPGA 内のどこでも比較的短距離でアクセス可能な BRAM ブロックが存在しているため、高速な IO に適している。一方、URAM は FPGA 内の一部に局在して配置されたメモリである。FF や LUT の位置によって URAM へのアクセス時間に大きな差があるなど BRAM ほど柔軟に使えない一方、高密度に値を保存できるため、BRAM よりも大量のデータを保存するのに適している。

FPGA 内の回路ではないが、外部に接続された DRAM を使うことも可能である。RFSoc では PS 側と PL 側のどちらでも DRAM との接続が可能であり、PS 側の DRAM は CPU とのデータ通信に向き、PL 側の DRAM は FPGA と直結しているため FPGA 内での大容量データ保存に向く。DRAM ではギガバイト単位の大容量を確保できる代わりに、BRAM、URAM などよりもさらに帯域が狭く、高頻度の通信には適さない。なお、本研究では PS 側の DRAM のみを使用し、PL 側の DRAM は使用しなかった。

各種メモリ回路は、LUT、BRAM、URAM、そして DRAM と順に大容量になる代わりに帯域が狭くなっていくため、用途に合わせて適切なメモリを使用することが求められる。

これら LUT、FF、DSP、各種メモリ回路などを組み合わせることで、できるだけ少ないリソースで、高い動作クロックの回路を実現することが目標になる。

## 開発の流れ

本節では、FPGA 上に回路を実装する際の開発の流れについて説明する。図 2.3 に本研究で回路を実装した際の開発フローの概略を示す。

まず、回路が行う計算アルゴリズムを一般のプログラミング言語のような高級言語で記述し、コンパイラによってハードウェア記述言語のソースコードに変換する。このプロセスは高位合成 (High Level Synthesis, HLS) と呼ばれている。ハードウェア記述言語とは論理回路の仕様を記述するための専用の言語であり、Verilog や SystemVerilog, VHDL などが挙げられる。基本的には C++, Python といった現代のプログラミング言語と比べると低級な言語である。

高位合成を活用することでハードウェア記述言語の代わりに高級言語で回路を記述することができるので、アルゴリズムのようなより本質的な内容に集中して開発できるメリットがある。AMD Xilinx の FPGA では高位合成のために Vitis HLS という統合開発環境が提供されているため、本研究ではそれを利用した。

Vitis HLS は C または C++ に対応しており、本研究では C++ を使用した。基本的には通常の C++ と同様に記述することで、コンパイラがハードウェア実装を推論してハードウェア記述言語のソースコードに変換してくれる。通常の C++ に加え、任意の  $n$  bit 整数への対応や、プラグマによる実装手法 (パイプライン化や外部インターフェイスの形式など) の指示など、ハードウェア特有の仕様を表すための拡張も用意されており、これらを組み合わせて開発を行う。

高位合成によって C++ で書かれたソースコードをハードウェア記述言語に翻訳した後、IP コアと呼ばれる、回路のブロックにカプセル化・パッケージ化する。この処理はソースコードのカプセル化によって扱いやすくするために行われ、以後は IP コアを接続することによって回路を組み

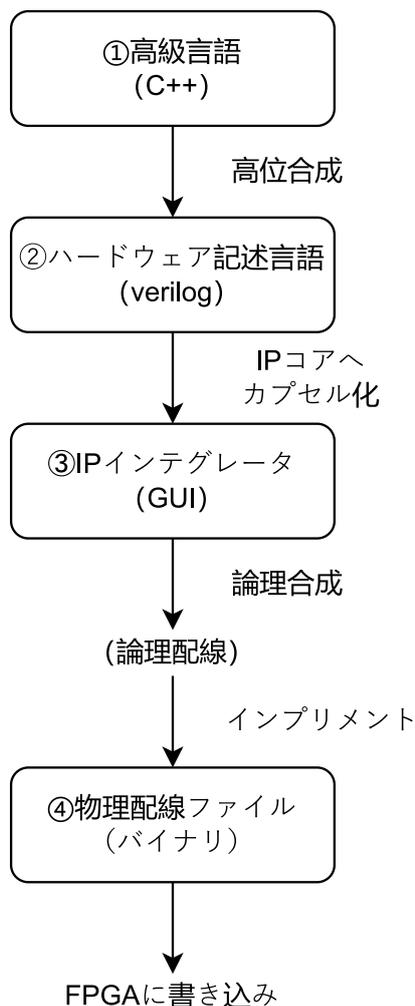


図 2.3: FPGA に自作回路を実装するまでの流れ。高級言語で回路のアルゴリズムを記述し、3 回のコンパイル（高位合成、論理合成、インプリメント）とカプセル化によって物理配線ファイルを作製し、それによって FPGA 上に回路を構成する。Verilog による回路記述や、既製の IP コアを使用することも可能である。

立ていく。ここまでの処理は Vitis HLS 上で行われる。

次に、様々な IP コアを IP インテグレータと呼ばれる GUI 上で接続し、FPGA 上に実装したい回路の全体図を構成する (図 2.4)。この段階以降は Vivado という AMD Xilinx 製の別の開発ツールを用いる。組み合わせる IP は自作した IP の他、AMD Xilinx 製やサードパーティ製の IP を使用することもできる。本研究では FFT 回路などの処理の中核となる部分の多くは高位合成によって独自設計する一方、その他の特別な仕様が不要なパーツでは AMD Xilinx 製の既存の IP を積極的に活用した。詳細は第 3 章で説明する。また、Vivado 上では、動作させるクロックやリセット信号、PS (CPU) との接続など、最終的に実装したい物理回路に必要な仕様を全て記述していく。

IP インテグレータで回路図を作製したら、次に論理合成、インプリメントと呼ばれる 2 回のコン

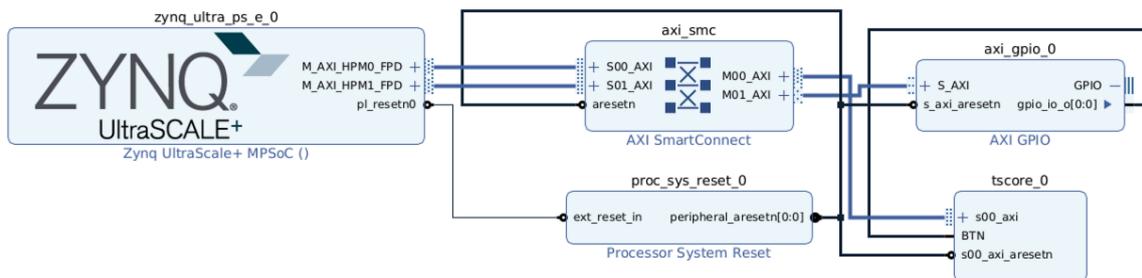


図 2.4: IP インテグレータのスクリーンショット。薄い青色のブロックがそれぞれ IP と呼ばれる回路ブロックを表す。ブロックの左側から出る線が入力や子側のインターフェイス接続、右側の線が出力や親側のインターフェイス接続を表す。このダイアグラムから、左端の PS (CPU) が中央上のインターコネクトを通じて右側の IP と接続していることや、右上の IP の出力が右下の IP に入力されていることが分かる。なお、見やすさのためクロックの配線は非表示にしている。

パイルを行う。論理合成では IP インテグレータ上の回路情報をもとに、配線遅延などを考慮しない論理的な回路に変換し、次のインプリメントで、配線遅延やクロックの乱れ、回路の配置など、ボードの特性を考慮した、FPGA 上での物理配線に変換する。これらの過程では人間はコンパイルオプションの指定などを除きほとんど介入せず、自動的に物理配線へと変換される。

これらの過程により作製した物理配線ファイルを FPGA に読み込ませることで、設計した回路を FPGA 上に実装することができる。物理配線ファイルの FPGA への書き込みは何度でも行うことができ、一回の書き込みにかかる時間は数秒程度である。また、書き込みは外部端子を接続して行うほか、Pynq のようなフレームワークを通じて、PS からソフトウェア的に行うことも可能である。

## 2.4 ソフトウェア設計の概略

FPGA はハードウェア回路として処理を行うため、決まった処理を高速に実行するのに適している。一方で、その場の要求に応じて臨機応変に処理を行うのには適していない。RFSoc には CPU も搭載されているので、そのような処理は PS (CPU が搭載されている側) で行う。本節では、PS で行うソフトウェア処理について説明する。

図 2.2 で示したように、CPU は同一チップ上の PL (FPGA) や、DRAM のような外部チップと接続されており、またイーサネットのような外部 IO も持っている。PL 側とは AXI 規格のバスが用意されており、データを相互にやり取りすることができる。そのため、PS は PL と RFSoc 外部とのやり取りの仲介に適しており、例えば、FPGA で取得した測定データの送信や、ネットワーク経由での FPGA のコンフィギュレーションなどの用途に使用できる。本研究での実際の処理内容は 3.6 節で紹介する。

RFSoc に搭載されている CPU は OS を動かすのに十分な性能を持ち、既に OS (Ubuntu) が入った SD カードイメージも配布されている。このイメージを使うことで、OS 上から容易に PS

と接続されたデバイス・IO にアクセスできる。つまり、PS 用のソフトウェア開発は通常の PC 向けのソフトウェア開発と同様に行える。

さらに、Pynq という PS から PL をより簡単に扱えるフレームワークも提供されている。Pynq は Ubuntu、Jupyter および Python をベースとしたフレームワークで、Python ライブラリを通じて PL 上の各要素にアクセスできる。簡単な Python のスクリプトで、例えば物理配線ファイルの FPGA への書き込みを指示したり、PL 上の IP コアの計算結果の読み取りや設定の変更をしたりすることができる。

## 第 3 章

# 各コンポーネントの設計と実装

本章では、実際に 4 GHz 帯域幅の分光計を構築する手法について詳述する。3.1 節でハードウェアやデータ処理の流れを概説した後、以降の節でそれぞれのコンポーネントの設計と実装について述べる。フーリエ分光計の核心となる、フーリエ演算回路の詳細な回路設計・計算アルゴリズムについては第 4 章で改めて記述する。

### 3.1 ハードウェア、FPGA 内回路の構成

図 3.1 に分光計の設計の概要を、また、図 3.2 に製作した分光計のセットアップ写真を示す。

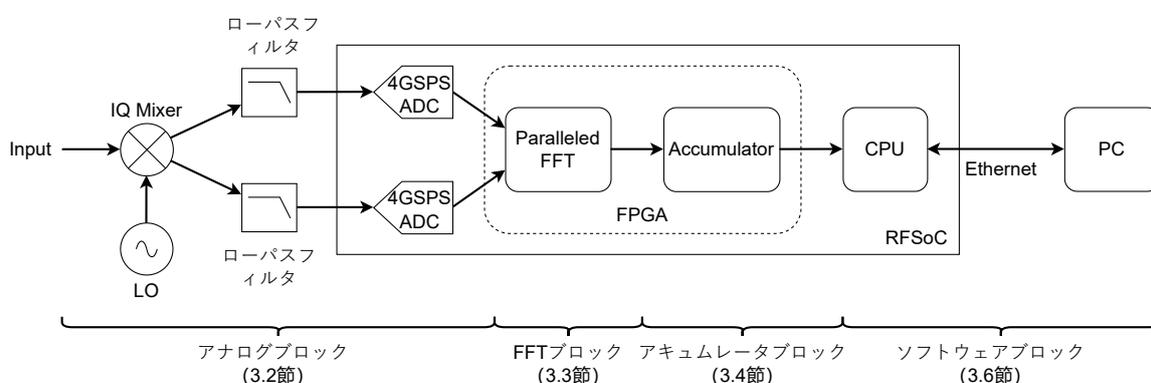


図 3.1: 分光計のハードウェア及び FPGA 内の回路の構成。入力信号は IQ ミキサによって 2 GHz の直交信号にダウンコンバートされ、RFSoc に送られる。RFSoc 上では、4 GSPS の ADC によって時系列のデジタル信号に変換したのち、FPGA 上の並列 FFT 回路によってパワースペクトルを求め、アキュムレータ回路によって時間平均を取る。その後、得られたパワースペクトルは内蔵 CPU へ転送され、イーサネットを通じて外部 PC に送信される。

分光計は複数の部品で構成され、また、RFSoc はさまざまな役割を果たしているが、各コンポーネントが果たしている役割で分けると、次の 4 つのブロックに大きく分けられる。まず、入力信号はアナログブロックに送られる。アナログブロックではまず、IQ ミキサによって入力信号をその

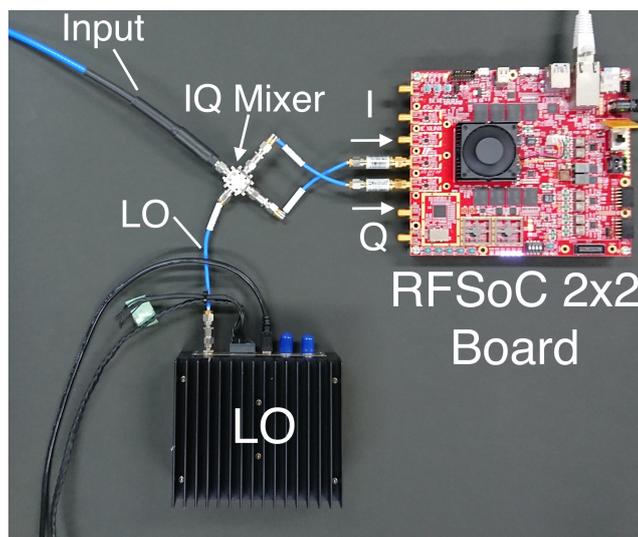


図 3.2: 分光計のセットアップ。入力信号は IQ ミキサによって LO 信号との差分周波数の直交信号にダウンコンバートされ、その後 RFSoc 上で分光処理される。

周波数と LO 信号の周波数との差の周波数の直交信号にダウンコンバートする。LO 信号は局部発振器によって生成された単色光であり、周波数を自由に設定できる。その後、RFSoc 上の ADC によって A/D 変換を行い、FFT ブロックに信号を送る。FFT ブロックでは信号をフーリエ変換し、パワースペクトルを求める。求めたパワースペクトルはアキュムレータブロックで時間平均をとってデータ量を圧縮したのち、得られたデータを PS 上の DRAM へ転送する。PS にデータが送られた後は、ソフトウェア処理によって外部に測定データを送信する。各ブロックの説明は次節以降で行う。

今回製作した分光計では、これまで DOSUE-RR 実験で探索を行っていない 30 GHz をターゲットとして部品を選定した。IQ ミキサとして Marki microwave 製 MMIQ-1040LS、局部発振器として用いるシグナルジェネレータに NATIONAL INSTRUMENTS 製の FSL-0020、RFSoc を搭載するボードとして、ZYNQ UltraScale+ RFSoc XCZU28DR を搭載した HiTech Global 製 RFSoc 2x2 ボードを使用した。RFSoc 2x2 ボードに搭載されたコンポーネントの性能を表 3.1 に示す。また、外部との通信には 1 Gbps のイーサネットケーブルを用いた。

なお、IQ ミキサと局部発振器を変更することで、他の周波数帯域の信号にも適用可能である。

30 GHz の  $10^{-6}$  倍程度の幅の周波数ピークを探索するため、FFT 回路の FFT 点数は  $N = 2^{17}$  とし、31.25 kHz の周波数分解能を確保した。<sup>\*1</sup> 後に、FFT 点数を  $N = 2^{18}$  に拡大して周波数分解能を 15.625 kHz とした高分解能版も製作している。

\*1  $4.096 \text{ GHz 幅} / 2^{17} = 31.25 \text{ kHz}$

表 3.1: ZYNQ UltraScale+ RFSoc 2x2 ボードの搭載リソース

リソース	性能	搭載数	主な用途
CPU			
ARM Cortex-A53		4	コンピューティング
ARM Cortex-R5F		2	システム制御
FPGA			
LUT	-	425280	組み合わせ回路の作成
FF	-	850560	順序回路の作成
BRAM	36 Kbit	1080	メモリ
URAM	288 Kbit	80	メモリ
DSP	-	4272	四則演算回路
DRAM			
(PS)	4 GB	1	
(PL)	4 GB	1	
ADC			
分解能	12 bit		
サンプリングレート	4.096 GSPS	2	AD 変換
DAC			
分解能	14 bit	2	
サンプリングレート	6.554 GSPS		DA 変換
その他 IO			
イーサネット	1000 Mbps		ネット通信
SD カードスロット		1	システムドライブ

## 3.2 アナログ回路 / ADC

複素フーリエ変換によって一つの FFT 回路で  $f_s$  の帯域幅を分光するため、IQ ミキサと、2つの ADC を利用する。2つの ADC は RFSoc の Multi Tile Sync 機能によって同期されている。同期のための基準クロックは RFSoc 2x2 Board に搭載されているクロック生成回路を使用した。IQ ミキサと ADC は SMA ケーブルで接続し、IF 信号を 2 GHz 以下に制限するため、間にローパスフィルタ (VLFG-1800+) を挿入する。ローパスフィルタの挿入損失を図 3.3 に示す。ADC は RFSoc (XCZU28DR) に内蔵の ADC を使用し、4.096 GSPS のサンプリングレートで、12 bit の分解能を持っている。

ADC が取得したデータは FPGA 内の RF Data Converter と呼ばれる AMD Xilinx 製の IP に

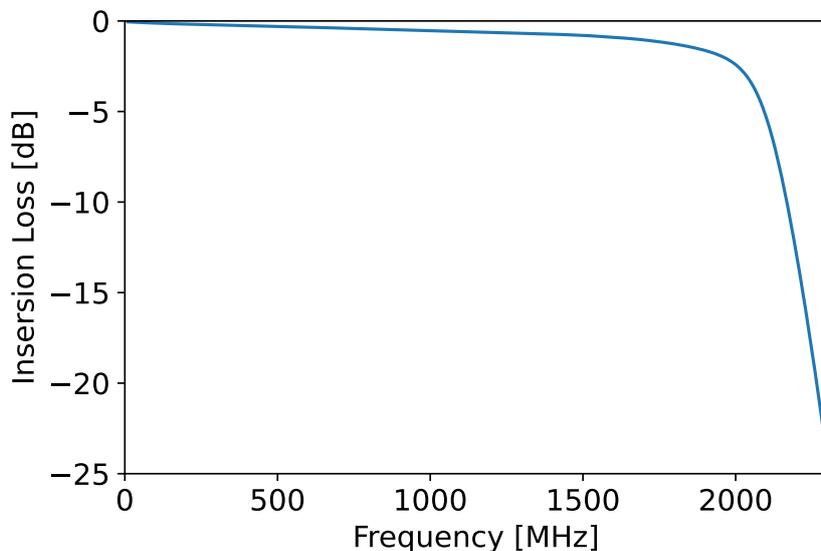


図 3.3: 使用したローパスフィルタ (VLFG-1800+) の挿入損失。プロットの際は公式のデータシート [12] を参照した。

送られる。RF Data Converter は 512 MHz で動作し、1 クロックに 8 つの時刻の ADC 信号をまとめて出力する (図 3.4)。

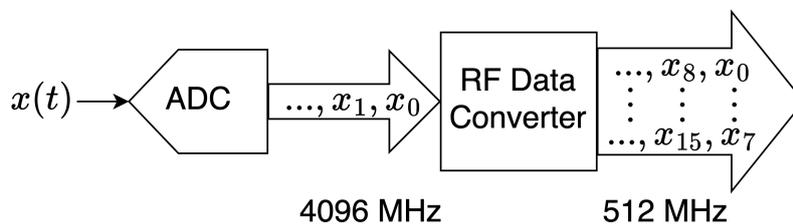


図 3.4: RF Data Converter による信号の受け渡し。ADC は 4.096 GSPS の頻度でデータを取得し、FPGA 上の RF Data Converter へ送る。RF Data Converter はその 1/8 の動作クロックである 512 MHz で動き、1 クロックあたり 8 つの時刻のデータをまとめて出力する。

### 3.3 FFT ブロック

その後、ADC からの信号は FFT ブロックでパワースペクトルへと変換される。

FFT ブロックは大きく分けて、ADC からの入力の分配、クロックの変換、FFT 演算、パワースペクトルの計算の 4 段階に分かれている。ただし、FFT 演算の後段とパワースペクトルの計算はコンパイラによる最適化を狙って同一の IP コアに含めており、IP インテグレータ上でのブロックダイアグラムは図 3.5 のようになっている。クロック変換回路と FFT 演算の前段では既存の IP を活用し、入力分配回路、FFT 後段・パワースペクトル計算回路は高位合成を用いて独自に設計した。

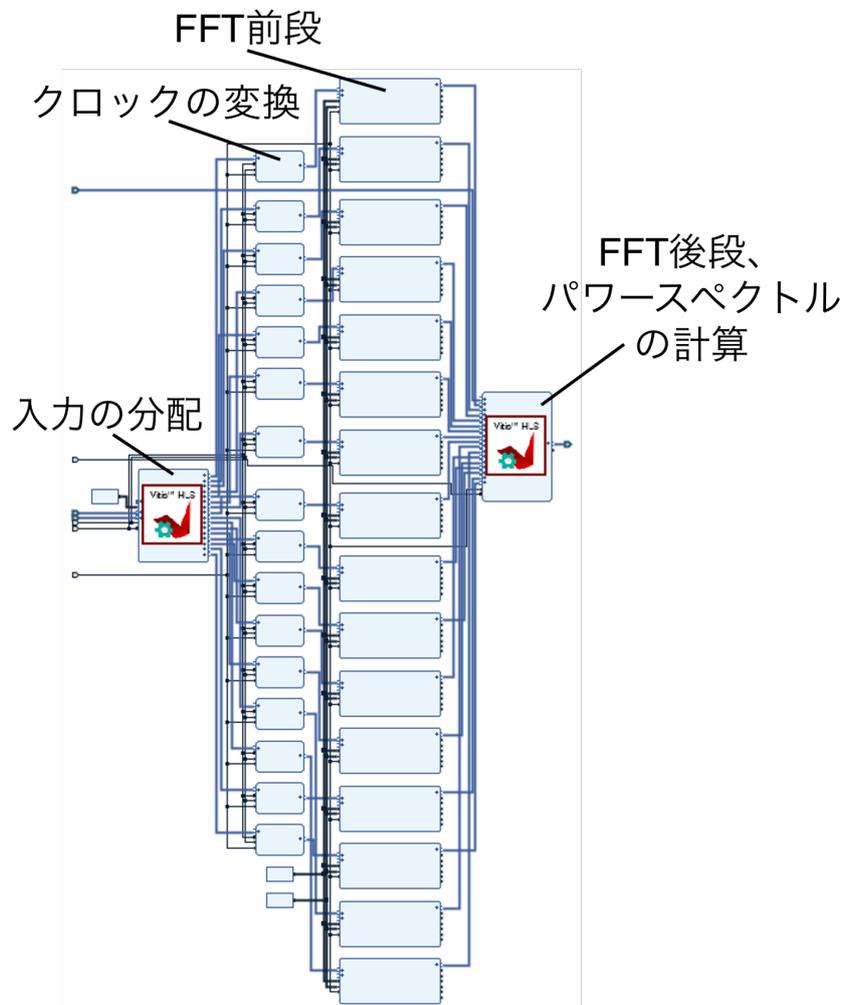


図 3.5: IP インテグレータ上で見た FFT ブロックの構成。図の左から入力の分配回路、クロック変換回路、FFT の前段回路（直列 FFT）、FFT の後段（回転と並列 FFT）およびパワースペクトルの計算回路の 4 種類の IP によって分光処理を行う。

前述の通り 2 つの ADC からの入力信号は 512 MHz で、1 クロックあたり 8 個の時刻の信号がまとめて出力される。しかし、この動作周波数で巨大な回路を動作させるのは困難なため、FFT 計算は 300 MHz で行う。512 MHz、8 個/クロックで ADC からの入力を絶え間なく処理するために、300 MHz で毎クロックあたり 16 入力を受け付けることが可能な 16 並列の MDF アーキテクチャを採用した。この回路では、最大 4800 MSPS のスループットまでデッドタイムなしに分光できる。FFT 演算回路の詳細な実装は第 4 章で説明する。

ADC からの信号を FFT 回路の入力数に合わせるため、FFT 回路の初段に入力を分配する回路を設置する。ここでは、512 MHz で 8 個ずつ送られてくる 2 組の信号を 2 クロック分読み込み、時刻ごとに分けた 16 組の 2 つの ADC の信号を 2 クロックおきにまとめて出力する。

クロック変換部は非同期 FIFO によって実装され、ADC の時刻ごとに分配された 512 MHz の信号が、300 MHz に変換される。FIFO には 2 クロックおきに入力されるため、300 MHz でも詰まることはない。

次に、FFT 部分で変形された ADC 信号を複素スペクトルに変換する。FFT 部分の入力は 12 bit 幅であり、出力は  $N = 2^{17}$  の FFT 回路で  $12 + 17$  となって 29 bit 幅、 $N = 2^{18}$  の FFT で  $12 + 18$  となって 30 bit 幅である。前述の通り、FFT 部分では 16 並列の MDF アーキテクチャを採用しており、1 クロックあたり 16 個の時刻の信号を処理できる。

時間効率を最大化するため FFT の窓関数は矩形窓 (窓関数なし) とした。窓関数には信号の他の周波数への漏れ込みを減らす効果があるが、DP-CDM 探索では興味があるのは周波数ピークであり、漏れ込みは測定に大きな影響を与えない。

最後に FFT 演算を行った後、得られた複素スペクトルからパワースペクトルを求める。この処理は、実部と虚部の 2 乗和を求める回路を 16 個並列することで実装できる。出力先のアキュムレータ回路に合わせ、出力は 54 bit 幅の固定小数点数 ( $N = 2^{17}$  の FFT 回路の仕様)、または 32 bit 浮動小数点数 ( $N = 2^{18}$  の FFT 回路の仕様) とした。これらの仕様を表 3.2 にまとめる。

表 3.2: FFT ブロックの仕様。

仕様	$N = 2^{17}$	$N = 2^{18}$
入力信号数	16	16
入力形式	固定小数点数	固定小数点数
入力 bit 幅	12 bit	12 bit
FFT 終了時点の形式	固定小数点数	固定小数点数
FFT 終了時点の bit 幅	29 bit	30 bit
出力形式	固定小数点数	浮動小数点数
出力 bit 幅	54 bit	32 bit

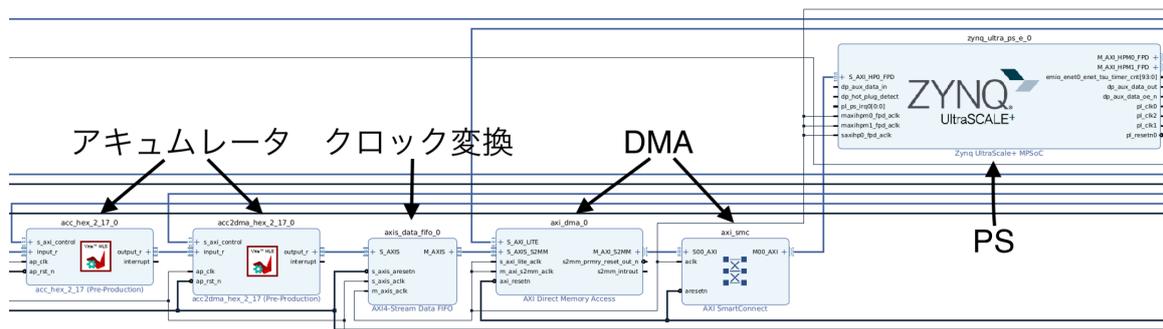


図 3.6: アキュムレータ回路および PS 転送部のブロックダイアグラム。赤色の 2 つの IP がアキュムレータ回路で、それぞれ総和の計算 (左) と並び替え (右) を行う。右端の大きなブロックが PS を表しており、間の 3 つの IP を通じて転送される。

### 3.4 アキュムレータブロック

得られたパワースペクトルのデータレートは 200 Gbps を超えるため、このままでは FPGA 外部に送ることができない。そこで、アキュムレータ回路によってパワースペクトルの時間平均 (総和) を計算し、データレートを削減してから PS へ転送する。アキュムレータブロックの IP インテグレーション上のブロックダイアグラムは図 3.6 のようになっている。

アキュムレータ回路は左の 2 つの IP からなり、それぞれ総和の計算 (左) と並び替え (右) を行う。

図 3.7 にアキュムレータ回路の概略を示す。まず、FFT 回路からパワースペクトルが入力され、小計が前段の URAM に一時保存される。M 個のスペクトルの総和を求め終わったのち、後段の URAM に転送する。ここまでの処理は 16 個の  $k$  について並列に bit reverse order <sup>\*2</sup>で行われる。最後に後段の URAM から natural order に並び替えて出力する。これまで 16 並列でデータを扱ってきたが、PL から PS へ送ることのできるデータ幅に制限があるため、出力は直列で行う。

平均する個数を  $M = 1024$  として、 $N = 2^{17}$  の FFT 回路では入力幅は 54 bit、出力幅は 64 bit で、 $N = 2^{18}$  の FFT 回路では入出力ともに 32 bit の浮動小数点数で製作した (表 3.3)。アキュムレータ回路は、全て高位合成によって製作した。

アキュムレータ回路の出力は図 3.6 中央右の DMA (Direct Memory Access) 回路を通じて PS に転送される。DMA とは CPU での処理を介さずに直接 DRAM に書き込む処理のことで、DRAM の帯域を最大限活用できる。DMA を行うための IP は AMD Xilinx が提供しているので、それを利用した。インプリメントでの物理配線を行いやすくするため、DMA 回路は 100 MHz で動作する。

\*2 通常の数字の数え方では 0, 1, 2, 3, 2 進法で書けば 0000<sup>(2)</sup>, 0001<sup>(2)</sup>, 0010<sup>(2)</sup>, 0011<sup>(2)</sup> と下の桁から 1 ずつ増やして数えていく。これは natural order と呼ばれる。一方、bit reverse order では、0000<sup>(2)</sup>, 1000<sup>(2)</sup>, 0100<sup>(2)</sup>, 1100<sup>(2)</sup> と上の桁から増やしていく。例えば、0 から 15 までを数える場合、0, 8, 4, 12, ... と数えていく。

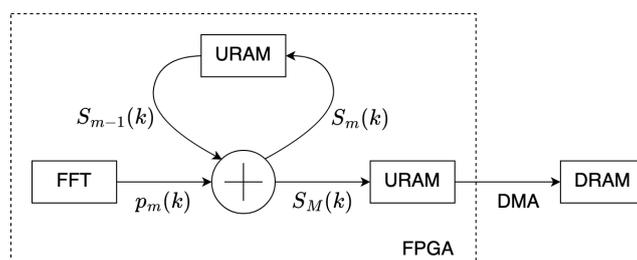


図 3.7: アキュムレータ回路の概略。FFT 回路から出力されたパワースペクトルの時間平均を FPGA 上で計算し、CPU で処理できるように DMA によって PS の DRAM に転送する。

表 3.3: アキュムレータ回路の仕様。

仕様	$N = 2^{17}$	$N = 2^{18}$
平均数 $M$	1024	1024
入力形式	固定小数点数	浮動小数点数
入力 bit 幅	54 bit	32 bit
出力形式	固定小数点数	浮動小数点数
出力 bit 幅	64 bit	32 bit

アキュムレータ回路と DMA 回路との間でのクロックの変換のため、アキュムレータ回路の直後に非同期 FIFO を接続する (図 3.6 中央左)。アキュムレータ回路によるデータ圧縮を行った後なので、ダウンスクロックによって入力データが詰まることはない。

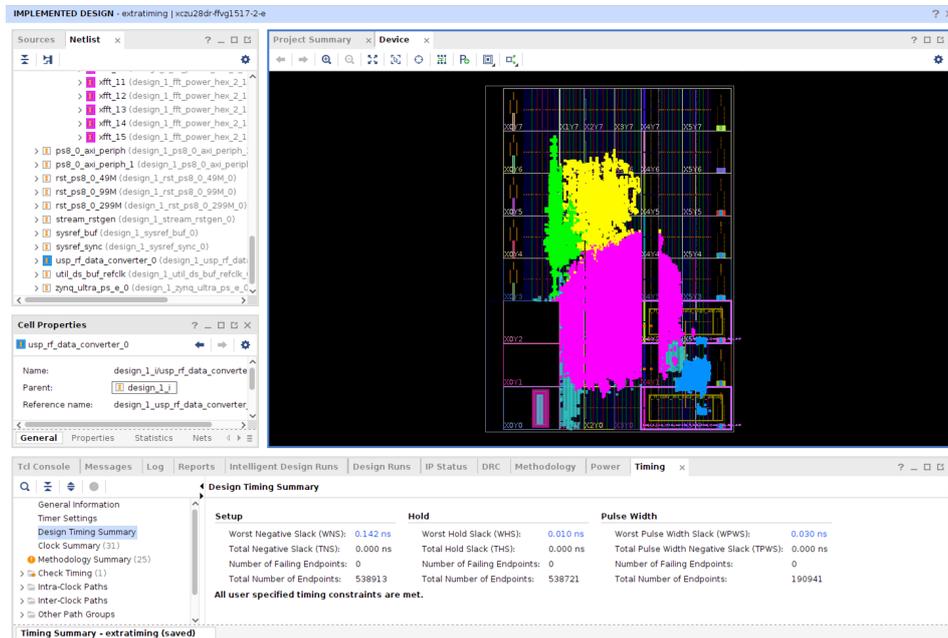
### 3.5 FPGA への回路実装 (インプリメント)

設計したこれらの回路ブロックを Vivado のブロックデザイン上で組み合わせ、FPGA へのインプリメントを行った。インプリメントの実行結果を図 3.8 に示す。

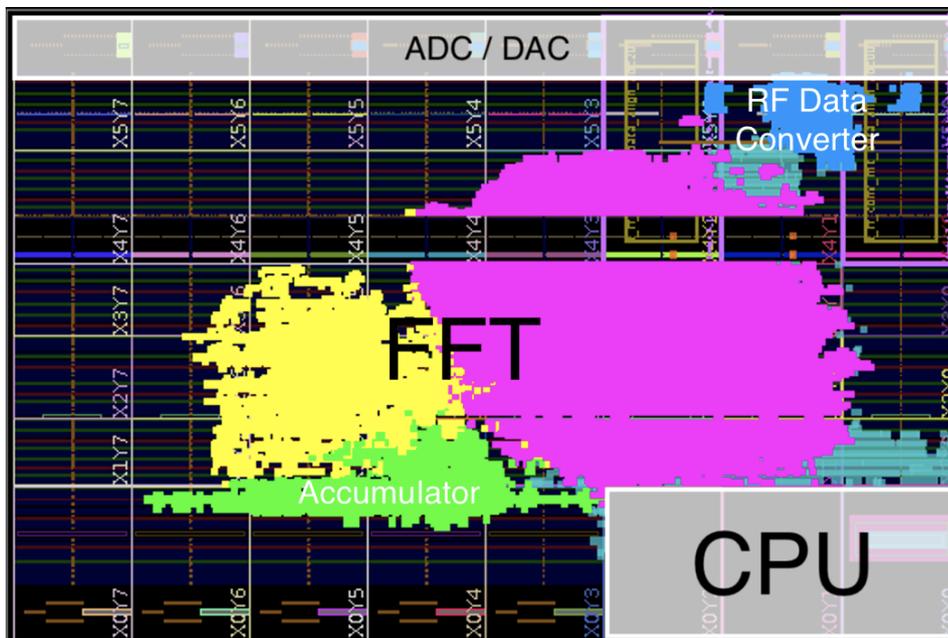
RFSoc 2x2 ボード上に搭載されている RSoC は ZYNQ UltraScale+ RFSoc XCZU28DR であり、RFSoc 内の FPGA の総リソース量および、使用したリソース量を表 3.4 に示す。URAM を除き各リソース種別の消費量は 50% を切っており、並列化や FFT サイズの拡大など、さらなる拡張の余地を残している。実際、 $N = 2^{17}$  の FFT 回路だけでなく、アキュムレータ回路の演算精度を 32 bit 浮動小数点数に変更することでメモリ使用量を削減し、FFT 点数を  $N = 2^{18}$  まで拡大した派生版のインプリメントにも成功した。

### 3.6 ソフトウェア

PS (CPU) では ADC や各種回路ブロックの制御、また、PL (FPGA) から転送されたスペクトルデータの外部への送信などを行う。図 3.9 にファームウェアおよびソフトウェアの構成の概要を



(a) 実行結果のスクリーンショット。下部の結果は、設計仕様通りのタイミングで動作できるように回路を構成できたことを示している。また、右上の図では各回路ブロックがFPGA上でどの位置に配置されているかを示している ((b) 参照)。



(b) 各回路ブロックのFPGA上での位置。上図を90°回転して表示している。それぞれ、青はADCとの通信部分、濃ピンクはFFT回路のうち前段の回路（直列FFT回路）、黄は後段の回路（並列FFT回路）とパワースペクトルへの変換部、黄緑はアキュムレータ回路を表す。

図 3.8: インプリメントの実行結果。

表 3.4: 分光計の FPGA リソース使用量

リソース種別	使用量	搭載数	使用率
LUT	85210	425280	20.04%
FF	157177	850560	18.48%
BRAM	156.5	1080	14.49%
URAM	62	80	77.50%
DSP	674	4272	15.78%

示す。PS と外部 PC はギガビット・イーサネットではつなげられ、制御メッセージや測定データの通信を行う。通信には ZeroMQ ライブラリを用いた。PS 側では 2.4 節で紹介した Pynq フレームワーク上で Python スクリプトを動かす、FPGA や ADC の初期設定や、2 つの ADC のクロック同期の設定、FPGA で求めたスペクトルの PC への転送を行う。また、FPGA に加えて CPU でも再度時間平均を取り、個々のセットアップで要求される時間精度や、イーサネットや HDD の帯域幅に合わせて調整する。

PL から PS への転送は 3.4 節で述べたように DMA によって行われ、PS の DRAM に直接書き込まれる。PS から外部 PC への転送にかかる時間はイーサネットを使用するため不定なので、DMA とイーサネット通信を別のスレッドで行う並行プログラミングモデル<sup>\*3</sup>を採用し、非同期処理を行う。

外部 PC 側は通常の Python スクリプトで、RFSoc と通信し、受け取ったデータをストレージに書き込む。データ量が多いため、受信したデータは都度書き込むようにしてメモリ消費を抑えている。

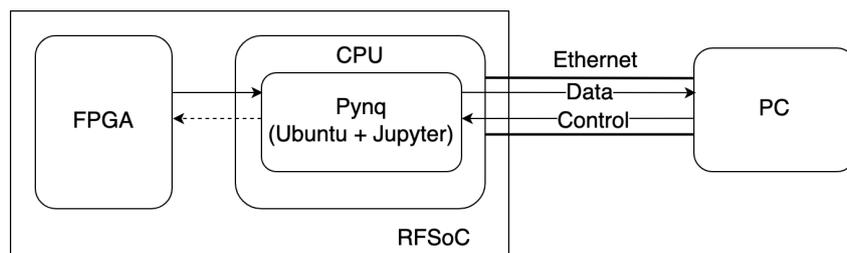


図 3.9: ソフトウェア部分の概要。Pynq 上で Python スクリプトを動かす、FPGA や ADC の初期設定や、2 つの ADC のクロック同期の設定、FPGA が求めたスペクトルの PC への転送を行う。

<sup>\*3</sup> 具体的には threading モジュールを用いた。CPython では同時に 1 スレッドしか実行できないため、threading モジュールではマルチコアによる並列処理は行えないものの、IO 律速な処理であるため threading モジュールの使用が有効である。逆に、並列処理が可能な multiprocessing モジュールを採用してみたところ、スレッド間のデータ共有にオーバーヘッドが生じて低速化した。

### 3.7 開発した分光計の仕様

表 3.5 に本研究で開発した分光計の仕様をまとめる。FFT 回路で ADC のサンプリングレートを上回るスループットを確保できたため、ADC をフルに生かした 4.096 GHz の帯域幅を達成することができた。また、分解能も目標とした 30 GHz 近辺の探索に対して十分な細かさとなる 16 kHz を達成することができた。

次の第 4 章ではこの分光計の核となった FFT 回路について詳述し、第 5 章では作製した分光計の性能評価を行う。

表 3.5: 本研究で開発した分光計の仕様のまとめ

項目	性能値
帯域幅	4.096 GHz
入力	Complex (IQ)
FFT サイズ	$2^{17} / 2^{18}$
周波数分解能	31.25 kHz / 15.625 kHz
ADC 分解能	12 bit
ADC のサンプリングレート	4.096 GSPS
FFT 回路のスループット	4.800 GSPS
時間平均の間隔	32.768 ms / 65.536 ms



## 第 4 章

# FFT 回路の実装

2.2 節で説明したように、分光計の帯域幅や周波数分解能は主に ADC と FFT 回路の性能によって決まる。本章では分光計の中核となる FFT 回路を効率よく実装した方法を詳述する。まず 4.1 節で FFT 演算のアルゴリズムと、それを回路として実装する方法を説明し、その後 4.2 節で本研究で使用した FFT 回路の設計を説明する。

### 4.1 FFT 回路の実装方法と課題

FFT 式の分光計では、時系列の入力データ  $x(t)$  を離散フーリエ変換 (DFT):

$$\tilde{x}(k) = \sum_{t=0}^{N-1} W_N^{kt} x(t) \quad (4.1)$$

によって周波数スペクトル  $\tilde{x}(k)$  に変換する。ここで、 $N$  は正の整数であり、 $k, t$  は 0 から  $N - 1$  までの整数を取る。また、 $W_N$  は  $W_N = e^{-2\pi i \frac{1}{N}}$  で定義される原始  $N$  乗根である。式 (4.1) の形からわかるように、DFT はフーリエ変換を離散化した変換であり、有限のデータで計算できるような  $k, t$  は離散化され、さらに一定の範囲に制限されている。

一見、 $N$  点の  $\tilde{x}(k)$  を全ての  $k$  について求めるためには、 $N$  個の加算・乗算を  $N$  回繰り返して  $O(N^2)$  の計算量が必要に見える。しかし、実際には Cooley-Tukey のアルゴリズム [13] により、 $O(N \log N)$  の計算量で DFT を計算できることが知られており、この手法は一般に高速フーリエ変換 (FFT) と呼ばれている。さらに、FFT 演算を汎用プロセッサで計算するよりもさらに高速に行うために、ハードウェア回路として FFT 演算を絶え間なく実行する手法が提案され [14, 15]、このような回路はパイプライン FFT 回路と呼ばれている。より効率的な回路実装のために、パイプライン FFT 回路は半世紀以上にわたって研究され、様々な用途に合わせた回路が提案されている [16]。

本研究ではこれらの先行研究を踏まえつつ、3.3 節で説明するように独自の回路を設計し FPGA 上に実装することで、デッドタイムなしでの 4 GHz 以上の同時帯域幅の分光を 1 枚のボード上で実現した。本節では、FFT 回路を設計する上で前提となる Cooley-Tukey のアルゴリズムおよび、

基本的なハードウェア FFT 回路の設計について簡単に説明する。

#### 4.1.1 Cooley-Tukey のアルゴリズム

まず、FFT の中核である Cooley-Tukey のアルゴリズムについて説明する。以下では簡単のため、 $N = 2^n$  と書ける点数の DFT で考える。また、本章では非負の整数  $x$  に対して、 $x_i$  を二進法で 0 から数えて  $i$  桁目の数と定義する。つまり、 $x = \sum_{i=0}^{n-1} 2^i x_i$  である。さらに  $x_{i:j}$  を  $x$  の  $i$  桁目から  $j$  桁目までの数、すなわち  $x_{i:j} = \sum_{l=i}^j 2^{l-i} x_l$  と定義する。例えば、 $x = 6$  (2進法で  $x = 110^{(2)}$ ) とすると、 $x_0 = 0, x_{1:2} = 3$  となる。

Cooley-Tukey のアルゴリズムでは、 $2^n$  点の DFT を、まず 2 つの  $2^{n-1}$  点の DFT に、その後、4 つの  $2^{n-2}$  点の DFT、8 つの  $2^{n-3}$  点の DFT、... とより小さな点数の DFT に分割していくことによって効率よく計算する。最初の分割では、式 (4.1) を次のように変形する：

$$\tilde{x}(k) = \tilde{x}(2^{n-1}k_{n-1} + k_{0:n-2}) \quad (4.2)$$

$$= \sum_{t_0=0}^1 \sum_{t_{1:n-1}=0}^{2^{n-1}-1} W_{2^n}^{(2^{n-1}k_{n-1} + k_{0:n-2})(2t_{1:n-1} + t_0)} x(2t_{1:n-1} + t_0). \quad (4.3)$$

ところで、 $W_k$  は原始  $k$  乗根だったので  $W_{2^n}^{2^m} = W_{2^{n-m}}$  であり、特に

$$W_{2^n}^{2^{n-1}k_{n-1} \cdot 2t_{1:n-1}} = 1 \quad (4.4)$$

となる。それゆえ、式 (4.3) は

$$\tilde{x}(2^{n-1}k_{n-1} + k_{0:n-2}) = \underbrace{\sum_{t_0=0}^1 W_2^{k_{n-1}t_0}}_{(1)} \underbrace{\frac{W_{2^n}^{k_{0:n-2}t_0}}{(2)}}_{(2)} \underbrace{\sum_{t_{1:n-1}=0}^{2^{n-1}-1} W_{2^{n-1}}^{k_{0:n-2}t_{1:n-1}} x(2t_{1:n-1} + t_0)}_{(3)} \quad (4.5)$$

と変形できる。ここで、式 (4.5) の計算量を考える。まず、 $k_{n-1} = 0$  と  $k_{n-1} = 1$  の場合 (つまり  $k = k_{0:n-2}$  と、 $k = 2^{n-1} + k_{0:n-2}$  の場合) では、下線部 (2), (3) までは全く同じ計算を  $t_0 = 0, 1$  について行っていることに着目すると、 $2^n$  点の DFT を求めるには、全ての  $k_{0:n-2}, t_0$  に対して (2), (3) を計算した後、(1) の総和を  $2^n$  回求めればよいことがわかる。また、(3) は DFT の形をしており、 $2^{n-1}$  点の DFT を  $t_0 = 0$  と  $t_0 = 1$  の 2 回行えば全ての  $k_{0:n-2}, t_0$  について求められる。さらに、(2) は一回の複素数乗算、(1) は  $W_2 = -1$  から一回の複素数加算で求められる。これらをまとめると、 $N$  点の FFT 演算の計算コストを  $c(N)$ 、一回の複素数乗算・加算の計算コストをそれぞれ  $m, a$  とおくと、

$$c(2^n) = 2^n(m + a) + 2c(2^{n-1}) \quad (4.6)$$

と求められる。 $2^{n-1}$  点の DFT でも同様に計算を分割できるので、分割を繰り返すと、式 (4.5) は

$$\begin{aligned} \tilde{x}(k) = & \sum_{t_0=0}^1 W_2^{k_{n-1}t_0} W_{2^n}^{k_0:n-2t_0} \sum_{t_1=0}^1 W_2^{k_{n-2}t_1} W_{2^{n-1}}^{k_0:n-3t_1} \\ & \dots \sum_{t_{n-2}=0}^1 W_2^{k_1 t_{n-2}} W_{2^2}^{k_0 t_{n-2}} \sum_{t_{n-1}=0}^1 W_2^{k_0 t_{n-1}} x(t) \end{aligned} \quad (4.7)$$

と変形でき、計算コストは

$$\begin{aligned} c(2^n) &= (n-1)2^n(m+a) + 2^{n-1}c(2) \\ &= n2^n a + (n-1)2^n m \end{aligned} \quad (4.8)$$

と求められる。ここで、2 点の DFT が 2 回の加算で計算できることを利用した。この結果から、計算コストは  $c(N) \sim O(N \log N)$  であることが分かる。

#### 4.1.2 パイプライン FFT の回路の実装

前節の通り Cooley-Tukey のアルゴリズムを使えば、 $N$  点 FFT の計算量は  $O(N \log N)$  で計算できる。しかし、例として要求性能である 4 GSPS で入力される 12 bit の信号に対して  $2^{17}$  点 FFT を休みなく行う場合を考えると、 $2^{17} \cdot 0.25$  ns の間に  $\sim 17 \cdot 2^{17}$  回の複素乗算と加算、つまり  $\sim 100$  GFLOPS の速度で演算し、さらにその結果を全て読み書きするために  $\sim 1$  Tbps の帯域と  $\sim 1$  MB のメモリが必要になってしまう。CPU でこの演算を行うのはメモリの帯域・レイテンシを考えると困難である。そのため、高速に FFT を行うためには GPU や FPGA のようなメモリ帯域に富んだハードウェアを用いる必要があり、計算効率を考えると専用の計算回路を設計することが好ましい。

前述の [16] の通り、これまで半世紀以上に渡ってハードウェア FFT 回路について研究が行われ様々な回路実装が提案されてきたが、これらはそれぞれ異なる用途に最適化されており、用途に合わせた回路実装を選択する必要がある。

本研究では、時系列の入力データを絶え間なく FFT 演算していく必要があるため、それに適した MDF アーキテクチャを採用した。

#### SDF アーキテクチャの構成

MDF (Multi-path Delay Feed-back) アーキテクチャは SDF (Single-path Delay Feed-back) アーキテクチャの拡張として構成できるので、先に SDF アーキテクチャの構成について述べる。

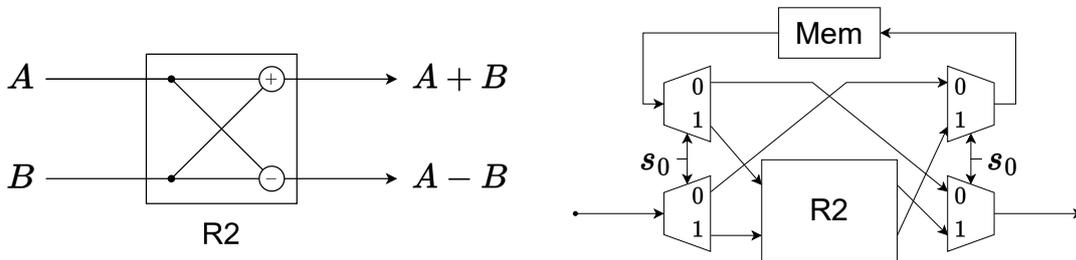
まず、最も簡単な FFT 演算として 2 点 FFT を考える。式 (4.7) から、2 点 FFT は

$$\tilde{x}(k) = \sum_{t_0=0}^1 W_2^{k_0 t_0} x(t) = \sum_{t=0}^1 (-1)^{kt} x(t) \quad (4.9)$$

と書ける。式 (4.9) は図 4.1a の回路に  $x(0), x(1)$  を入力すれば  $\tilde{x}(0), \tilde{x}(1)$  を同時に求められる。この回路は 2 つの変数が交差する様子から (Radix-2 の) バタフライ演算回路と呼ばれており、以

降では R2 回路と略す。全ての FFT 演算は 2 点の FFT 演算と複素数乗算の組み合わせとして書けるので、R2 回路と複素乗算回路の組み合わせによって FFT 回路を構成できる。

この回路は 2 入力 2 出力の回路であるが、例えば ADC から 1 クロックに 1 つの時刻の  $x(t)$  のデータが送られてくるような状況では、1 入力の回路が求められる。そのような場合、図 4.1b のような一時メモリを備えた回路構成にする必要がある。図 4.1b では、 $s_0$  の値によって入力や出力を選択する回路（図中の台形）を使って、クロックによって挙動を変化させる。 $s_0 = 0$  のクロックでは入力データを一時メモリに保存し、 $s_0 = 1$  のクロックでバタフライ演算を行う。また、出力も 1 出力にする場合、R2 回路の出力の片方は一時メモリに保管し、バタフライ演算を行わない  $s_0 = 0$  のクロックで出力する。このように、入出力を一時メモリに退避させることで 1 入力 1 出力のパイプライン回路が構成できる。



(a) Radix-2 のバタフライ演算回路 (R2)。2 点の FFT 演算に対応する。A, B の 2 つの入力を受け取り、 $A + B, A - B$  を並列に計算して出力する。

(b) 1 入力 1 出力化した 2 点 FFT 回路。台形内の 0, 1 は  $s_0 = 0, 1$  の場合の信号の流れをそれぞれ表す。 $s_0 = 0$  のときはバタフライ演算を行わず入力信号を一時メモリに保存し、以前の計算結果を一時メモリから取り出す。

図 4.1: 2 点 FFT 演算を行う回路。左は 2 入力 2 出力、右は 1 入力 1 出力である。

次に、1 入力 1 出力で 4 点のパイプライン FFT 回路を考える。式 (4.7) から、4 点の FFT 演算は

$$\tilde{x}(k) = \sum_{t_0=0}^1 W_2^{k_1 t_0} W_4^{k_0 t_0} \sum_{t_1=0}^1 W_2^{k_0 t_1} x(t) \quad (4.10)$$

と表せる。式 (4.10) は 2 つの 2 点 FFT と 1 個の複素数乗算の組み合わせとして書けるから、それぞれ R2 回路と複素乗算回路 (図中  $\otimes$ ) に対応させ、図 4.2 のような回路を構成すれば計算することができる。図 4.2 では、 $x(t) = A, B, C, D$  とデータが順に入力される時、まず前段の R2 回路で  $A + C, B + D, A - C, B - D$  を計算する。また、そのために一時メモリには  $A, B, A - C, B - D$  を保存する。次に、中段の複素乗算回路では、 $k_0 = t_0 = 1$  となる  $B - D$  のときのみ  $-i$  を乗算し、後段の R2 回路で最終的な  $\tilde{x}(k)$  を求める。

4 点の回路での考え方を一般の  $N = 2^n$  点の FFT 演算に応用すれば、式 (4.7) は図 4.3 のような、図 4.1b の回路  $n$  ステージを、間に  $n - 1$  個の複素乗算回路を挟んで繋げた回路に対応させる

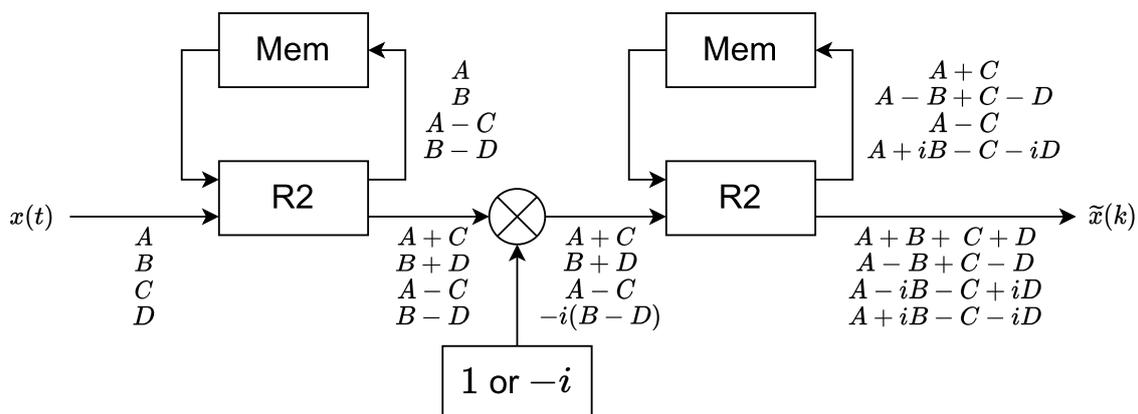


図 4.2: パイプライン FFT 回路の例 (4 点 FFT)。各クロックごとに  $x(t) = A, B, C, D$  と順に入力され、2 回のバタフライ演算 (R2) と 1 回の乗算 ( $\otimes$ ) を行い、 $\tilde{x}(k)$  が出力される。図では省略しているが、クロックによっては R2 回路をバイパスし、入力は一時的メモリに格納し、代わりに一時メモリから以前に格納した値を出力する。

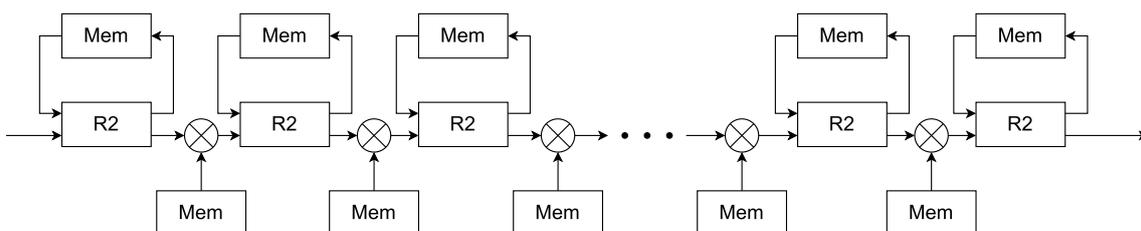


図 4.3: SDF パイプライン回路の構造。

ことができる。図 4.3 の構成は、SDF (Single-path Delay Feed-back) アーキテクチャと呼ばれており、時系列順に入力されるデータを FFT 演算するのに適している。

#### 4.1.3 MDF アーキテクチャへの拡張

SDF アーキテクチャは 1 入力 1 出力の回路なので、動作クロック  $f$  Hz の回路では ADC のサンプリングレートが  $f$  SPS 以下でないと全ての入力データを分光することは出来ない。高サンプリングレートの ADC を利用する場合、FFT 回路を並列化して 1 クロックで複数の時刻の入力を処理できるようにする必要がある。

まず、2 並列の FFT 回路を考える。このとき、入力信号  $x(t)$  は 1 クロックに 2 つずつ、 $[x(0), x(1)]$ ,  $[x(2), x(3)]$ ,  $\dots$ ,  $[x(2t_{1:n-1}), x(2t_{1:n-1} + 1)]$  というように送られる。ここで、式 (4.7) を見ると、 $x(2t_{1:n-1})$  と  $x(2t_{1:n-1} + 1)$  のデータは最後の総和を取るまで独立して計算をしていることが分かる。つまり、図 4.4 のように  $t_0 = 0, 1$  に対応した 2 つの SDF アーキテクチャの FFT 回路を用意し、最後に R2 回路で合わせることで、2 入力 2 出力のパイプライン FFT 回路が構成できる。

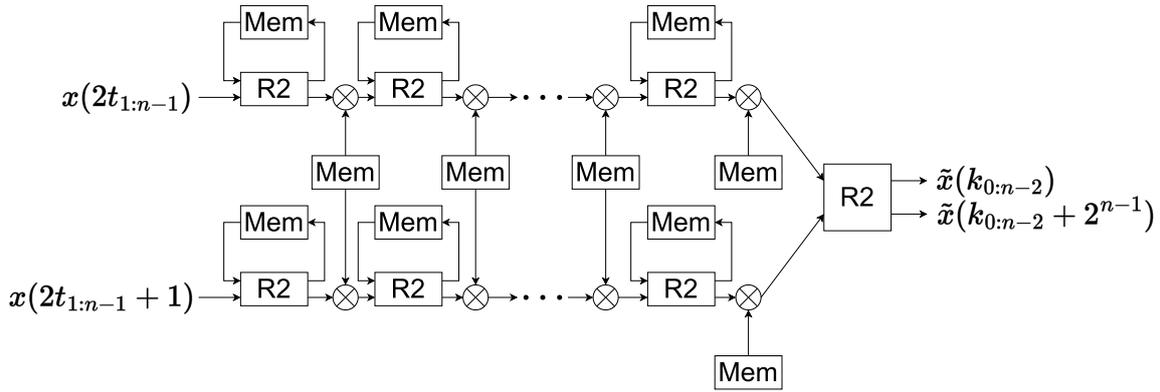


図 4.4: 2 並列 FFT 回路の例。2 並列の MDF アーキテクチャを採用している。 $t_0$  の値別に 2 つの SDF 回路を用意することで、SDF 回路の二倍のスループットで FFT 処理を行える。2 つの SDF 回路での複素乗算では、どのクロックでも二回路で同じ回転因子を用いるため、回転因子のメモリは共有可能である。

同様に、 $P = 2^p$  個の SDF 回路によって各  $x_{0:p-1}$  に対応した  $\sum_{t_{p-1}=0}^1 W_2^{k_{n-p}t_{p-1}} W_{2^{n-p+1}}^{k_{0:n-p-1}t_{p-1}} \cdots \sum_{t_{n-1}=0}^1 W_2^{k_0 t_{n-1}} x(t)$  の値を  $P$  個並列に計算し、最後に  $P$  入力  $P$  出力の  $P$  点 FFT 回路でまとめることで、 $N$  点で  $P$  入力  $P$  出力のパイプライン FFT 回路が構成できる。このような構成を MDF (Multi-path Delay Feed-back) アーキテクチャと呼び、例えば 4 点の MDF パイプライン FFT 回路は図 4.5 のようにして作製できる。 $P$  点の MDF アーキテクチャでは、動作クロック  $f$  Hz に対して、 $P \cdot f$  SPS 以下の ADC であれば、全ての入力信号を分光することが可能である。

#### 4.1.4 回路のリソース使用量と省リソース化の手法

MDF アーキテクチャを使用すれば、十分に並列数  $P$  を増やすことで ADC から送られてくる全てのデータをリアルタイムに分光できるが、実際に回路として実装するためには回路面積や演算器の個数、メモリ量といったリソースの使用量を考慮する必要がある。そこで、パイプライン FFT 回路の作製に必要なリソース量を考える。

##### 各コンポーネントとリソース使用量

式 (4.7) を見ると、パイプライン FFT 回路ではバタフライ演算、複素乗算、そして入出力データや複素乗算器で使用する係数を保存するメモリが必要となる。このうちバタフライ演算は R2 回路によって行われ、図 4.1 のように、2 個の複素加減算回路、あるいは 4 個の実加減算回路によって実装できる。

一方、複素乗算部分ではより大きな回路が必要となる。一般的に、複素乗算は 4 回の実乗算と 2 回の実加算、または 3 回の実乗算と 5 回の実加算が必要である。また、 $n$  bit の実数同士の乗算は典型的に  $n$  bit の加算を  $n$  回行う必要があるため、乗算回路は加減算回路より  $O(n)$  倍大きくなる。さらに、乗算係数の  $W_l^m$  も  $k, t$  の値に応じて用意する必要がある。特に  $n$  が大きい場合には、



で、回転メモリの使用量は  $O(N)$  である。つまり、FFT 点数が大きくなるほど LUT 方式による実装は難しくなる。

逆に粗い回転では、より低コストな回転器を実装できることがある。特に、回転因子が  $W_4^m$  の回転乗算は自明な回転と呼ばれ、 $W_4 = -i$  なので符号の反転や実部と虚部の入れ替えのみで表現可能なため、極めて小規模に構成できる。また、回転因子が  $W_8^m$  のときも、 $W_8 = \frac{1}{\sqrt{2}}(1+i)$  であることに注意すると、 $\frac{1}{\sqrt{2}}$  倍を計算する 2 個の定数乗算器のみで実装可能である。通常より乗算器の個数が少ない上、定数乗算器は一般の乗算器よりも回路規模を小さくできることが知られている [18]。このような低コストな回転器の使用量を増やし、普通の回転器 (以後は汎用回転器と呼ぶ) の使用数を減らすことが FFT 回路の効率化の鍵となる。

回転メモリに加え、R2 回路に入出力される値のための一時メモリ (以降データメモリ) が必要となる。データメモリの使用量は実装方法によって変わってくるが、 $P$  入力  $P$  出力の  $N$  点 FFT 回路の場合、最低でも複素数  $N - P$  個分のデータメモリが必要である。なぜなら、 $\tilde{x}(k)$  を計算するためには全ての  $x(t)$  の値が必要であるため入力が完了するまで出力することができず、最初の  $\tilde{x}(k)$  を出力するときに入力値とデータメモリ合わせて  $N$  個未満の値しかなかったとすると、全ての  $k$  について  $\tilde{x}(k)$  の値を計算できなくなるためである。

既にデータメモリ必要量が理論下限となるアーキテクチャは知られており、例えば SDF アーキテクチャのデータメモリ必要量は、1 ステージ目で  $x(t_{0:n-2})$  と  $x(t_{0:n-2} + 2^{n-1})$  のバタフライ演算をするために  $2^{n-1}$  個、2 ステージ目で  $2^{n-2}$  個、... となって合計  $N - 1$  個となる。

### Radix の変更による回転器の削減

これまで、Cooley-Tukey のアルゴリズムの説明では  $N$  点の DFT を 2 個の  $N/2$  点 DFT に分割する形で説明してきたが、それ以外の分割の仕方も考えられる。式 (4.1) は  $N = R * M$  と 2 以上の整数  $R, M$  で表せる一般の  $N$  に対して、

$$\tilde{x}(Mk' + k'') = \sum_{t'=0}^{R-1} W_R^{k't'} W_N^{k''t'} \sum_{t''=0}^{M-1} W_M^{k''t''} x(t' + Rt'') \quad (4.11)$$

と表せるので、 $R$  個の  $M$  点 DFT に分割できる。この  $R$  のことを Radix と言い、 $\sum_{t''=0}^{M-1} W_M^{k''t''} (\cdot)$  の形の演算を Radix- $R$  のバタフライ演算と呼ぶ。これを 4.1.1 節のときのように繰り返し行えば、 $N = R^k$  で書ける  $N$  に対して、 $N$  点の DFT は Radix- $R$  点のバタフライ演算の繰り返しとして表せ、Radix- $R$  の FFT と呼ぶ。例えば 4.1.1 節の FFT は Radix-2 の FFT である。 $R = 2$  の FFT 回路の代わりに、 $R > 2$  の FFT 回路を利用することで、必要な回転器の個数を減らせることが知られている。

$R = 4, N = 2^n$  の場合を考える。Radix-4 の FFT は、式 (4.1) を

$$\tilde{x}(k) = \sum_{t_{0:1}=0}^3 W_4^{k_{n-2:n-1}t_{0:1}} W_{2^n}^{k_{0:n-3}t_{0:1}} \dots \sum_{t_{n-4:n-3}=0}^3 W_4^{k_{2:3}t_{n-4:n-3}} W_{2^4}^{k_{0:1}t_{n-4:n-3}} \sum_{t_{n-2:n-1}=0}^3 W_4^{k_{0:1}t_{n-2:n-1}} x(t) \quad (4.12)$$

のように変形して計算する。式 (4.12) を SDF 回路として Radix-4 のバタフライ演算回路と回転器で実装した場合、必要な回転器の個数は  $n/2 - 1$  個なので、Radix-2 の FFT 回路と比べ必要数が半減する。同様に、Radix- $R$  の SDF 回路は  $n/R$  個の Radix- $R$  バタフライ演算回路と、 $n/R - 1$  個の回転器で実装できる。

一方で、 $R > 2$  の場合、バタフライ演算回路の実装面積も問題となる。例えば、Radix-4 のバタフライ演算回路を素直に実装した場合図 4.6a のようになるが、4 つの値の総和をとるには 3 回の加算が必要のため、合計 12 個の加減算器が必要となる。Radix- $R$  のバタフライ演算回路に必要な加減算器の個数は  $O(R^2)$  であり、 $R$  が増加するとバタフライ演算回路の必要数の減少を超えて増大する。また、 $W_R^{k_*t_*}$  の乗算も必要になるので、 $R > 4$  の場合はさらに乗算回路も必要になってしまう。

そこで、 $R > 2$  のバタフライ演算回路を R2 回路の組み合わせとして実装することで、バタフライ演算回路の規模を削減する手法が考案された [19, 20, 21]。  $R = 2^k$  のとき、この手法を Radix- $2^k$  と呼ぶ (基本的に、(普通の) Radix- $R$  とは別物として扱う)。例えば  $R = 2^2$  の場合、バタフライ演算回路を図 4.6b のように実装できる。図 4.6b は 8 個の加減算器で実装でき、1 入力 1 出力とした場合は上下の R2 回路を共有できるので必要量がさらに半減する。これは Radix-2 の場合の必要量と全く同程度であり、さらに回転器の個数は (普通の) Radix-4 と同程度という、両者の「いいとこどり」になっている。そのことが Radix- $2^k$  を特別視する理由の一つになっている。

Radix- $2^2$  による SDF 回路は図 4.7 のように実装でき、Radix-2 の SDF 回路から R2 回路の個数を増加させずに、乗算回路の数を半減できる。一般に Radix- $2^k$  の FFT 回路は Radix-2 の FFT 回路と同数の R2 回路で構成でき、回転器の個数の削減に有効である。

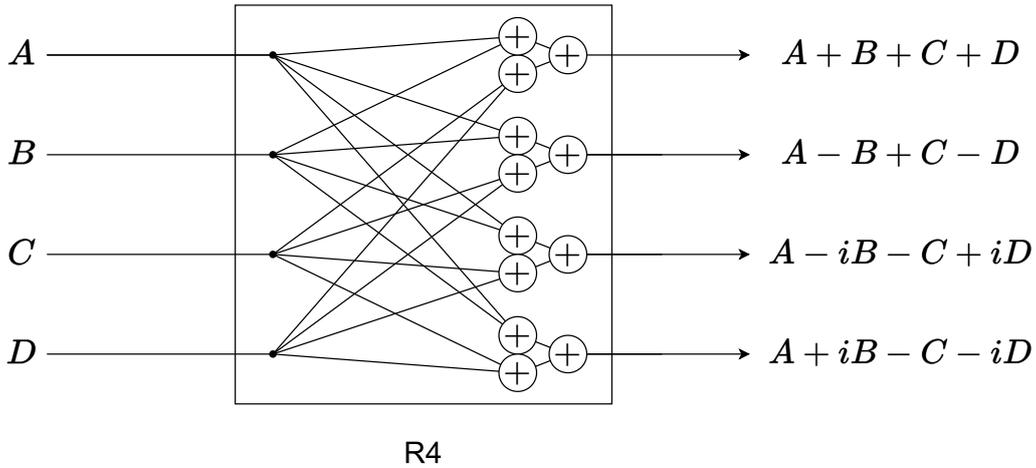
### 対称性によるメモリの削減と限界

データメモリは理論限界以上減らすことができないので、メモリ量を減らすには回転メモリの量を減らす必要がある。  $W_N^m$  の形で書ける回転因子は  $N$  通りの複素数値しか取り得ないので、実数  $2N$  個の回転メモリを用意すれば十分であるが、 $\sin, \cos$  の対称性などを活用することでより少ない回転メモリで LUT 式回転器を構成できる。

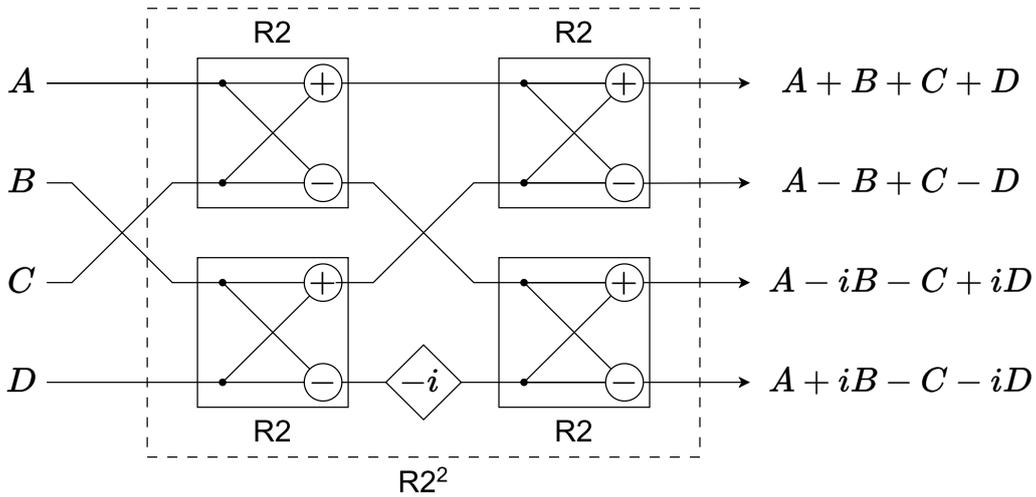
$W_N^M$  は、 $m', m''$  を  $m = \frac{N}{4}m' + m''$  ( $0 \leq m'' < \frac{N}{4}$ ) となるようにおくと

$$W_N^m = W_4^{m'} W_N^{m''} \quad (4.13)$$

と表せる。前述したように  $W_4^{m'}$  の乗算は符号の反転と実部・虚部の入れ替えで表現できるので、 $W_4^{m'}$  を乗ずることによる回路規模の増加は十分小さいと考えられる。 $m''$  の値は  $\frac{N}{4}$  通りしか取り



(a) Radix-4 のバタフライ演算回路。4 個の値の総和には加減算器  $\oplus$  が 3 個必要なので、必要な加減算器は  $4 \times 3$  で 12 個。



(b) Radix- $2^2$  のバタフライ演算回路。必要な加減算器は 8 個。

図 4.6: Radix-4 及び Radix- $2^2$  のバタフライ演算回路 (4 点 FFT 回路)。大きな Radix のバタフライ演算回路は、Radix- $2^2$  のように R2 回路の組み合わせとして実装することで Radix-2 の回路と同数の加減算器で実装できる。

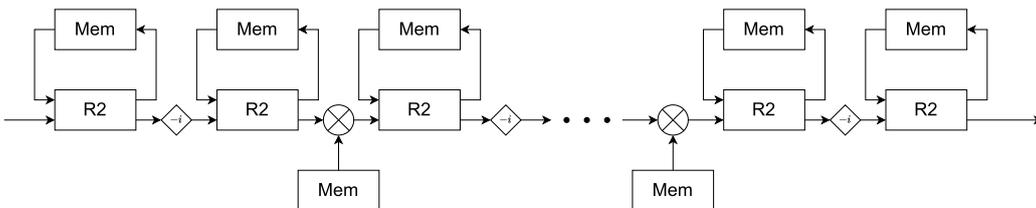


図 4.7: Radix- $2^2$  の SDF パイプライン回路の構造。乗算回路が 2 ステージあたり 1 個に削減されている

得ないので、 $\frac{N}{4}$  個の  $W_N^{m''}$  の値を記憶していれば任意の  $W_N^m$  の回転を計算できる。

さらに、 $W_N^{m''} = \cos\left(-2\pi\frac{m''}{N}\right) + i\sin\left(-2\pi\frac{m''}{N}\right)$  なので、 $\sin$ ,  $\cos$  の対称性から

$$\text{Im}\left(W_N^{m''}\right) = \text{Re}\left(W_N^{\frac{N}{4}-m''}\right) \quad (4.14)$$

と表せる。つまり、実部の値と虚部の値は同じメモリを共有できる。結局、実数  $\frac{N}{4}$  個を記憶するだけで LUT 式回転器を実装できる。

この手法を使うとメモリ量を 1/8 に減らせるが、実際に適用する場合、メモリの同時読み出し数を気にする必要がある。ハードウェア回路では、同じメモリから同時に読み出せる値の個数に制限がある場合があり、制限を超える数を読み出すためにはメモリを丸ごと複製しなければならない。そのため、SDF 回路のような 1 入力 1 出力の FFT 回路ではデータメモリに対して回転メモリは十分無視できるが、MDF 回路のような並列 FFT 回路では回転メモリとデータメモリの消費量は同程度になりうる。

FPGA のようなメモリ量の限られたハードウェアでは、回転メモリのサイズが FFT 点数の拡大のボトルネックになりうる。また、FPGA に実装する場合、仮に BRAM の総量が足りたとしても、データメモリを使いたい回路と回転メモリを使いたい回路が近傍に存在することから競合を起こし、演算回路から遠距離の BRAM を使用しなければならない可能性がある。配線距離が伸びるほどタイミングの調整が困難になるため、これは最大動作周波数の低下の原因になりうる。このように、特に並列 FFT 回路ではメモリ使用量が回路実装する上での課題となる。

## 4.2 作製した FFT 回路の構成

前節を踏まえて、実際の分光計の FFT 回路の設計を検討する。

3.2 節で述べたように、RF Data Converter は ADC 信号を 8 個並列に 512 MHz の動作クロックで出力する。しかし、MDF アーキテクチャのような大規模な回路ではリソース間の配線距離の調整が難しく、このような高クロックで動作させるのは難しい。そのため、3.3 節の通り FFT 回路は 300 MHz で動作させる。動作クロック 300 MHz で 4 GSPS の ADC 信号を処理するためには、16 並列以上の並列数が必要である\*1。

しかし、MDF アーキテクチャは並列数に比例して多くの乗算器、メモリを必要とするため、いかに少ない FPGA のリソースで作製するかが課題となる。そこで、本研究では最小限の並列数である 16 並列の MDF アーキテクチャを採用し、さらに以下のような設計の工夫によってできるだけ消費リソースを削減した。

\*1 FFT では、 $N$  は  $N = 2^n$  の形で表せることが通常求められるので、並列数  $P$  も  $P = 2^p$  と表せる必要がある。そのため、8 並列 FFT 回路の次に並列数の小さいのは 16 並列の FFT 回路である。

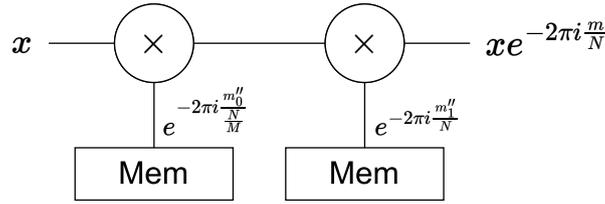


図 4.8: 本研究で開発した分割 LUT 式回転器の概要。回転を 2 回に分けることで、通常の LUT 式回転機と比べて複素乗算の回数が 2 倍に増える代わりに回転メモリの使用量が  $O(N)$  から  $O(\sqrt{N})$  に大幅に削減される。

#### 4.2.1 回転因子テーブルの分割による回転器の省メモリ化

4.1 節で述べた通り、回転メモリやデータメモリの必要量は基本的に FFT の点数に比例する。今回  $N = 2^n$  ( $n = 17$  または  $18$ ) と FFT 点数が十分大きく、FFT 回路を実装するためには  $\sim 10$  Mbit のメモリが必要となる上、 $n$  個の各ステージで 16 個の計算のための入出力が必要なことから  $\sim 100$ – $1000$  Gbps の帯域が求められる。このような高速 IO を扱うために、データメモリや回転メモリの実装には 40 Mbit もあり、かつ広帯域な FPGA 上の BRAM を利用するのが好ましい。BRAM は FPGA 上に分散配置されているため、メモリ消費量が増えるほど遠くに配置された BRAM まで使う必要があり、配線距離の増大によりレイテンシが増加して回路実装が困難になる。BRAM 使用量をできるだけ少なくするため、データメモリのような理論的な下限がなく、メモリ使用量を削減する余地が存在する回転器の省メモリ化を行った。

汎用回転器の実装として、CORDIC アルゴリズムを用いたものは FPGA 上では専用算術計算回路を使えないため効率が悪く、一方で LUT 方式では演算の回路規模が少ないことは 4.1 節で紹介した。そこで、LUT 方式の回転器のメモリ削減を目指す。

回転因子  $W_N$  の回転を実数  $N/4$  個の回転メモリで行う回転器は次のように計算していた：

$$W_N^m = W_4^{m'} W_N^{m''}.$$

さらにメモリ消費を減らすため、 $m'' = Mm_0'' + m_1''$  ( $0 \leq m_1'' \leq M$ ) と  $m_0'', m_1''$  をおいて、

$$W_N^m = W_4^{m'} W_{\frac{N}{M}}^{m_0''} W_N^{m_1''} \quad (4.15)$$

と 3 つの回転因子の積に分ける。 $m_1''$  は  $M$  通りの値を取れるので、 $W_N^{m_1''}$  の値を保存するのに必要なメモリサイズは、実数  $2M$  個である。一方、 $m_0''$  は 0 から  $\frac{N}{4M} - 1$  まで全ての値を取れるので  $\cos, \sin$  の対称性を適用でき、実数  $\frac{N}{4M}$  個分のメモリで  $W_{\frac{N}{M}}^{m_0''}$  の値を取り出せる。そのため、図 4.8 のような回路で式 (4.15) の式を計算することで、実数  $2M + \frac{N}{4M}$  個のメモリだけで回転演算を行うことができる。この構成の回転器を以後、分割 LUT 式回転器と呼ぶことにする。

分割 LUT 式回転器のメモリ使用量が最小になるのは、 $N = 2^n$  に対して  $n$  が奇数なら  $M = 2^{\frac{n-3}{2}}$ 、偶数なら  $M = 2^{\frac{n-4}{2}}, 2^{\frac{n-2}{2}}$  のときで、必要なメモリは奇数なら実数  $\sqrt{2N}$  個、偶数なら実

表 4.1: 回転因子  $W_N^m$  の回転を行う回転器の回路規模

方式	メモリ使用量	複素乗算回数	備考
LUT	$O(N)$	1	
分割 LUT	$O(\sqrt{N})$	2	本研究で開発。図 4.8 も参照。
CORDIC	$O(\log N)$	0	$O(\log N)$ 回のシフト演算と加減算を行う

数  $\frac{3}{2}\sqrt{N}$  個となる。通常のリソースの必要メモリ量は実数  $N/4$  個だったので、オーダーレベルでメモリ量を削減でき、例えば  $N = 2^{17}$  の場合 1/64 倍に削減できる。一方、乗算器などの演算器の規模は回転演算を 2 回行わなければならないため、およそ 2 倍弱に増えてしまう。

表 4.1 に各種回転器のリソース使用量を示す。このように、分割 LUT 式回転器は通常のリソース使用量の LUT 式の回転器と CORDIC アルゴリズムを用いた回転器の中間的な性質を持つ。なお、回転をさらに分割することで、演算器を  $k$  倍にする代わりにメモリ量を  $O(\sqrt[k]{N})$  倍にすることも可能である。

分割 LUT 式回転器は本研究で作製した分光計のように、通常のリソース使用量の LUT 式の回転器で実装するには FFT 点数  $N$  が大きすぎるが、 $\sqrt{N}$  はそこまで大きくない場合に有効である。

#### 4.2.2 Radix-2<sup>4</sup> MDF アーキテクチャの採用

分割 LUT 式回転器では乗算器のコストが 2 倍になってしまうが、最もメモリ消費量の多い  $W_N$  の回転を行うのは FFT 回路のうち 1 ステージのみで、残りのステージの回転器のメモリ消費量は半分以下に過ぎないため、必ずしも全ての回転器で分割 LUT 式を採用する必要はない。つまり、アーキテクチャの工夫によって汎用回転器の個数を減らしたり、メモリ量の多い一部の汎用回転器でのみ分割 LUT 式回転器を採用することで、回路規模の拡大を抑えつつ回転メモリを大幅に削減することが可能である。そこで、16 並列回路で最も汎用回転器の少ない既知のアーキテクチャとして、Radix-2<sup>4</sup> MDF アーキテクチャ [21, 22] を採用した。このアーキテクチャの計算方法と回路構成は次のようにして得られる。

まず、サイズ  $N$  の DFT は

$$\tilde{x}(k) = \sum_{t=0}^{N-1} W_N^{kt} x(t) \quad (4.16)$$

のように定義される。ここで、 $k, t$  は  $[0, N-1]$  の整数で、 $W_N$  は  $W_N = e^{-2\pi i \frac{1}{N}}$  で定義される原始  $N$  乗根である。並列処理を行うため、 $k, t$  を  $k = \frac{N}{16}k' + k'', t = t' + 16t''$  のように分割する。

$$\tilde{x}\left(\frac{N}{16}k' + k''\right) = \sum_{t'=0}^{15} \sum_{t''=0}^{\frac{N}{16}-1} W_N^{\left(\frac{N}{16}k' + k''\right)(t' + 16t'')} x(t' + 16t'') \quad (4.17)$$

$$= \sum_{t'=0}^{15} W_{16}^{k't'} W_N^{k''t'} \sum_{t''=0}^{\frac{N}{16}-1} W_{\frac{N}{16}}^{k''t''} x(t' + 16t''). \quad (4.18)$$

内側の総和はサイズ  $\frac{N}{16}$  の DFT である。この処理は、16 個の SDF パイプライン FFT 回路を並列させ、それぞれ別の  $t'$  の値の DFT 演算を行わせることによって行う。SDF 回路には Xilinx 製の既成 IP を使用した。この SDF 回路の詳細のアーキテクチャは不明だが、データシートから  $k \geq 2$  の Radix-2<sup>k</sup>SDF 回路であると考えられる。一方、外側は回転演算とサイズ 16 の DFT の組み合わせであり、15 個の回転器と、完全に並列化された 16 入力の FFT 回路によって処理する。この FFT 回路では、 $k' = 8k'_0 + 4k'_1 + 2k'_2 + k'_3$ ,  $t' = t'_0 + 2t'_1 + 4t'_2 + 8t'_3$  のように分割し、

$$\sum_{t'=0}^{15} W_{16}^{k't'} = \sum_{t'_0=0}^1 W_2^{k'_0 t'_0} W_4^{k'_1 t'_0} \sum_{t'_1=0}^1 W_2^{k'_1 t'_1} W_{16}^{(2k'_2+k'_3)(t'_0+2t'_1)} \sum_{t'_2=0}^1 W_2^{k'_2 t'_2} W_4^{k'_3 t'_2} \sum_{t'_3=0}^1 W_2^{k'_3 t'_3} \quad (4.19)$$

と変形して計算する。式 (4.19) で行う回転演算のうち、 $W_2^m, W_4^m$  で表される部分は符号反転、実部と虚部の入れ替えで表現できるため乗算器は不要である。また、 $k'_2, k'_3, t'_0, t'_1$  の値が予め分かっているため、 $W_{16}^{(2k'_2+k'_3)(t'_0+2t'_1)}$  の回転演算は定数乗算器で実装できる。 $W_{16}$  では  $W_4$  や  $W_8$  と違い乗算回数を減らすことはできないが、前述の通り定数乗算のコストは安い上、何より回転角がわかっているため回転メモリが必要ない。

式 (4.18), (4.19) から回路を構成すると図 4.9 のようになる。その結果、外側の総和で汎用回転器が必要なステージは 1 ステージのみとなり、図 4.10 のような一般的な構成と比べて汎用回転器の個数が 24 個から 15 個に減少し、実装面積を削減できる。本研究では SDF 回路の直後のステージの回転器 (図中黄●) にのみ分割 LUT 式回転器を採用した。この構成を採用することによって、汎用回転器を全て通常の LUT 式回転器のみとする場合と比較して、乗算回数の増加を 10% 未満に抑えつつ回転メモリの使用量を 1/8 ~ 1/16 倍に削減できる。その結果、この回路を用いて 3.5 節のように分光計を実装することができた。

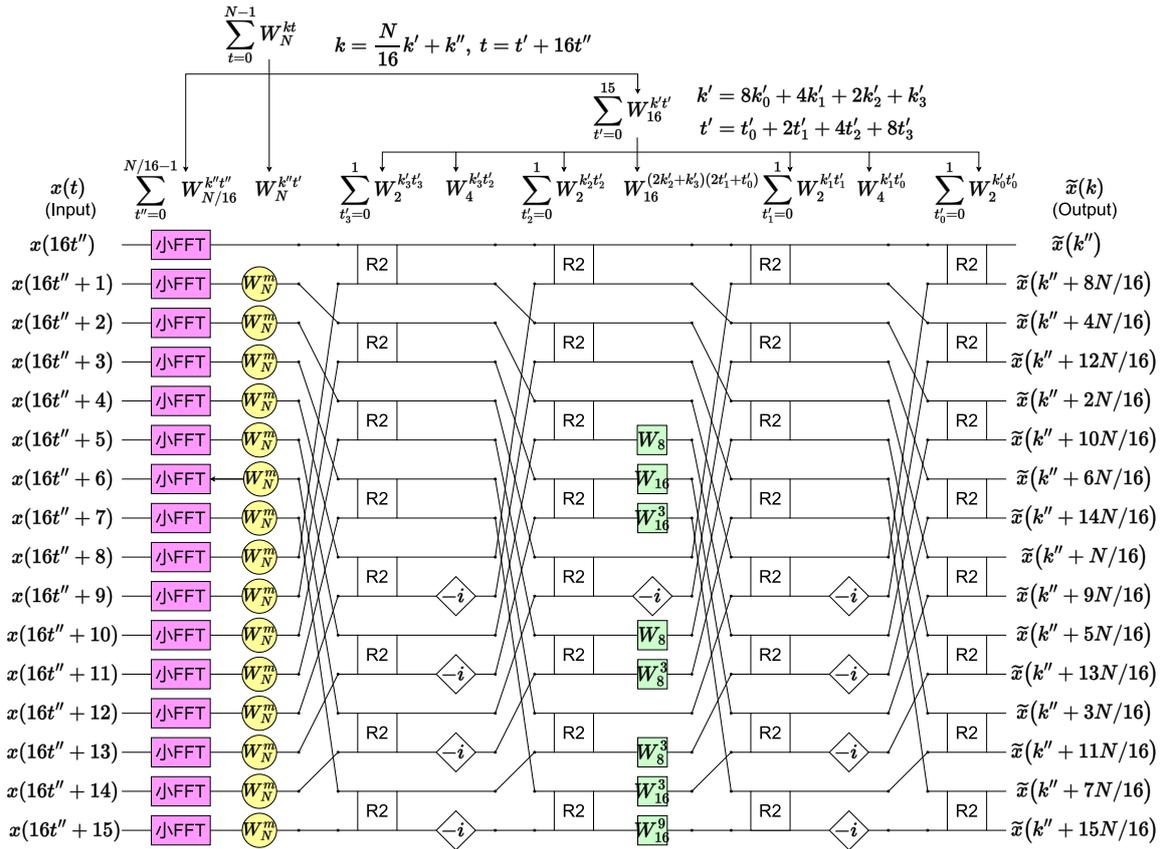


図 4.9: FFT 回路の構成。式 (4.16) の DFT を 16 個の SDF 回路 (ピンク) と、 $N = 16$  で 16 入力の FFT 回路に分割して計算する。後者はさらに式 (4.19) のように分割された Radix-2<sup>4</sup> 回路で、Radix-2 のバタフライ演算器 (R2)、 $-i$  乗算器 (ダイヤの  $-i$ )、回転演算器 (黄色の●または薄緑の角型) が図のように接続されて構成される。回転器のうち、黄色の●は汎用回転器を表し、本研究で開発した分割 LUT 方式を採用した。一方、薄緑の角型で表した回転器は汎用回転器よりも低コストな固定角の回転器 (定数乗算) である。

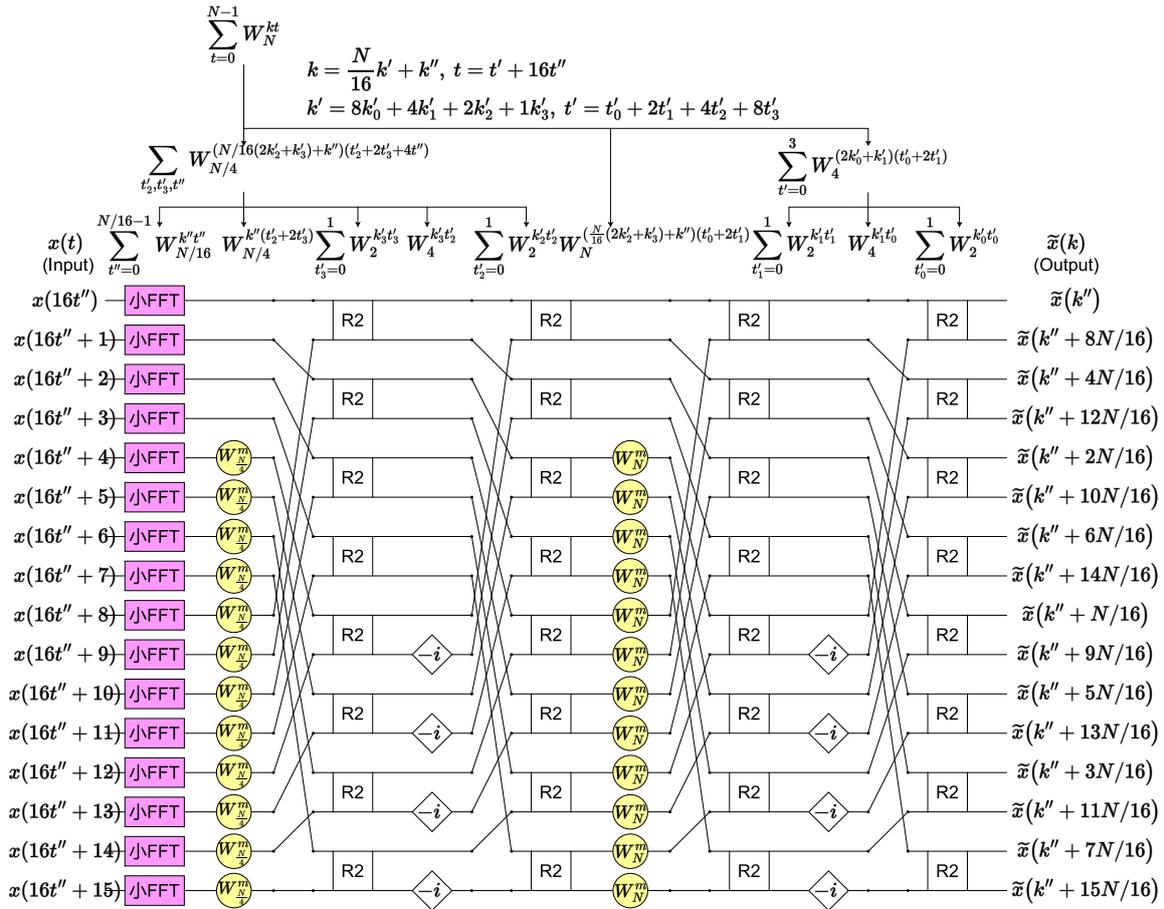


図 4.10: 一般的な Radix- $2^2$  アーキテクチャで 16 並列 FFT 回路を作製した場合の構成例。Radix- $2^4$  で実装した場合と比較して図中央でも汎用回転器 (●) が必要となり、汎用回転器の必要数が 15 個から 24 個に増える。その結果、図 4.9 よりも元々の回路規模が大きくなる上、分割 LUT 方式を採用した場合のデメリットも増加してしまう。

## 第 5 章

# 性能評価

本章では、製作した分光計の性能評価について述べる。これらの性能測定では、イーサネットの帯域に合わせるために、ハードウェア上での時間平均処理に加え、ソフトウェア上でさらに2つのスペクトルデータの時間平均をとった。そのため、 $2 \times 1024$  回の FFT のスペクトルによる、65.536 ms 間の時間平均となっている。

### 5.1 単色 RF 源を用いた分光動作の実証

まず、基本的な動作確認として、16 GHz の単色光を生成しスペクトルを観測した。入力信号、LO 信号ともにシグナルジェネレータ (FSL-0020) を使用した。LO 周波数は 15 GHz に設定し、RFSoc には 1 GHz の IF 信号が入力されることになる。

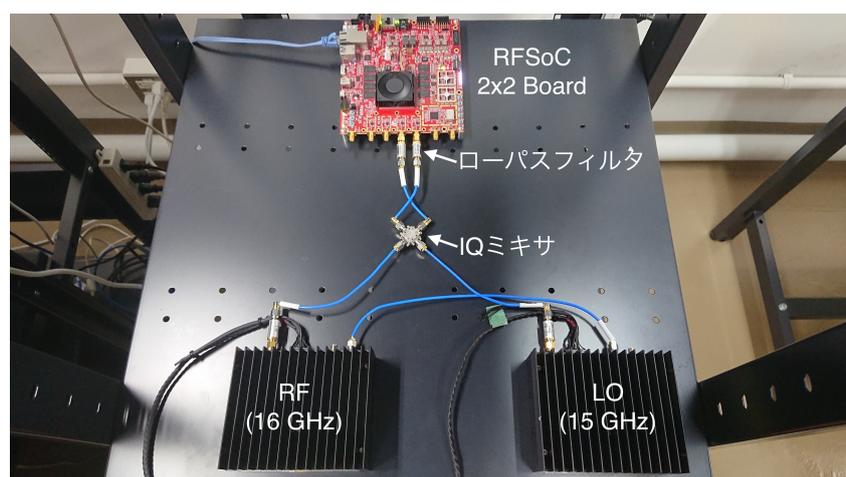


図 5.1: 測定のセットアップ。入力信号 (RF) (左、16 GHz)、LO 信号 (右、15 GHz) はどちらもシグナルジェネレータで生成した。

図 5.2 に観測されたスペクトルを示す。ピークは正確に 16 GHz のビンに観測された。また、IQ ミキサによるリークや、倍波成分はいずれも 20 dB 以上弱く、IQ ミキサのカタログ値と整合す

る。ノイズフロアは  $\sim -150 \text{ dB}_{\text{fs}}/\text{Hz}$  にあり、ADC のカタログ値と同程度であった。これにより、FPGA 内の FFT 回路で追加のノイズが発生していないことが確認できた。

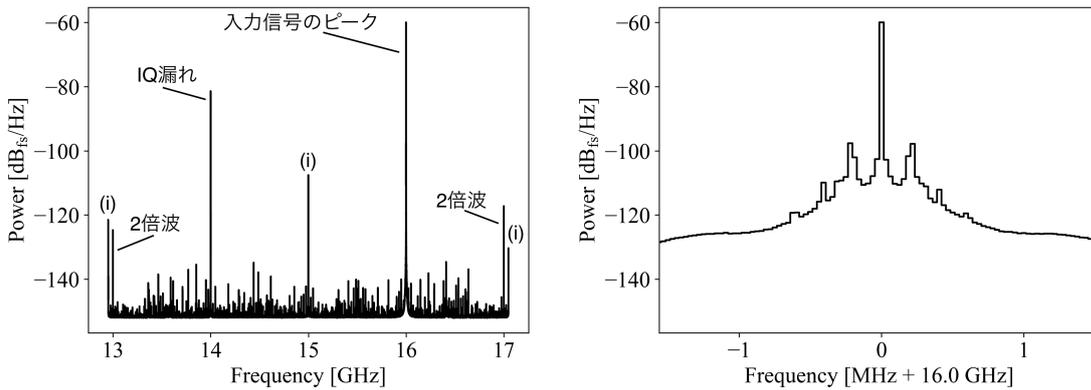
シグナルである 16 GHz を拡大したものを図 5.2b に示す。中心付近に対称な構造が見えるが、これは離散フーリエ変換が有限区間・個数のデータのみで計算していることによるもので、矩形窓(窓関数なし)とした場合の予想と一致している。

得られたスペクトルには RFSoc の性質上避けられないノイズが確認できた：

- (i) DC 近傍の  $1/f$  ノイズ由来の成分 (0 MHz および 2048 MHz)
- (ii) (i) と並列に計算される成分への (i) の漏れ込み (512 MHz, 1024 MHz, 1536 MHz, ...)
- (iii) クロック周波数 40.96 MHz およびその高調波

これらのノイズのある周波数領域で信号がノイズに埋もれてしまうため解析から除外する必要がある。ノイズの周波数があらかじめ分かっているので、探索の際は複数の LO 周波数で測定することで回避可能である。

以下の解析では (i) に関しては 10 MHz 幅、(ii) に関しては 1 MHz 幅、(iii) に関しては 0.25 MHz 幅の合計 50 MHz の帯域幅を除外して解析を行った。



(a) 得られたパワースペクトル。15 GHz の LO 入力による 14 GHz の IQ 漏れ、13.17 GHz の 2 倍波はいずれもピークより 20 dB 下回った。  
(b) ピーク付近のパワースペクトル。ピークは正確に 16 GHz にあり、漏れは 30 dB 以上低かった。

図 5.2: 16 GHz の単色信号を入力した時のパワースペクトル。ピークは正確に 16 GHz に観測され、その他のピークはいずれも 20 dB 下回った。ノイズフロアは  $\sim -150 \text{ dB}_{\text{fs}}/\text{Hz}$  だった。

## 5.2 線形性

次に、電力の線形性の評価を行った。図 5.3 にセットアップを示す。ダークマター探索では熱ノイズ程度の電力の周波数ピークを発見することが目標となる。微小電力での線形性を見るため、シグナルジェネレータによって 14 GHz の単色光を 0.05–2.00 mW で生成した後、アテネータで

-60 dB 減衰させた。この電力は 60 K の熱雑音をアンプで 60-80 dB 増幅させたときにスペクトルの各ビンで観測される電力に相当し、先行実験 [10] に近い。入力電力の大きさを変更できるように、入力信号のシグナルジェネレータを KEYSIGHT 製 E8257D に変更した (図 5.4)。なお、LO のシグナルジェネレータは変更していない。また、LO 周波数は 13 GHz とした。

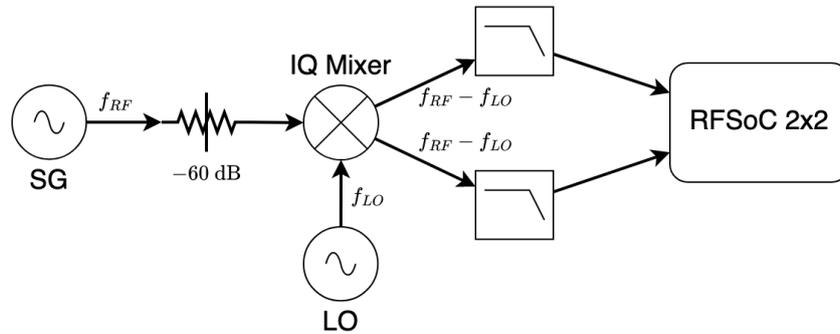


図 5.3: 線形性の測定のためのセットアップ。シグナルジェネレータが生成した  $f_{RF} = 14$  GHz の単色光をアテネータで 60 dB 減衰させ IQ ミキサに投入した。LO 周波数  $f_{LO} = 13$  GHz として 1 GHz にダウンコンバートし、RFSoc 2x2 ボード上で分光した。

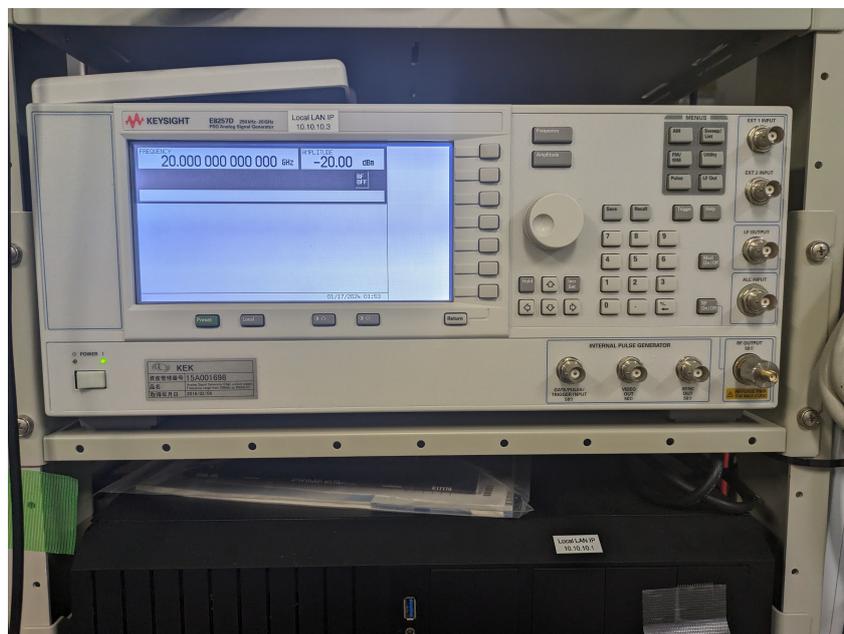


図 5.4: シグナルジェネレータ E8257D の写真。この測定では入力信号の電力を変更する必要があるため、シグナルジェネレータを変更した。

得られた電力値をオフセット付きの一次関数で fit し、線形直線からの残差を求めた。fitting の際、電力の相対誤差は一定であると仮定した。得られた入力電力と観測電力の関係は図 5.5 のようになり、0.5% 未満の精度で線形性を確認した。 $\chi \propto p_{DP}^{\frac{1}{2}}$  なので、非線形性による感度への影響は

< 0.25% と、主要な系統誤差に比べ一桁以上小さく無視できる。

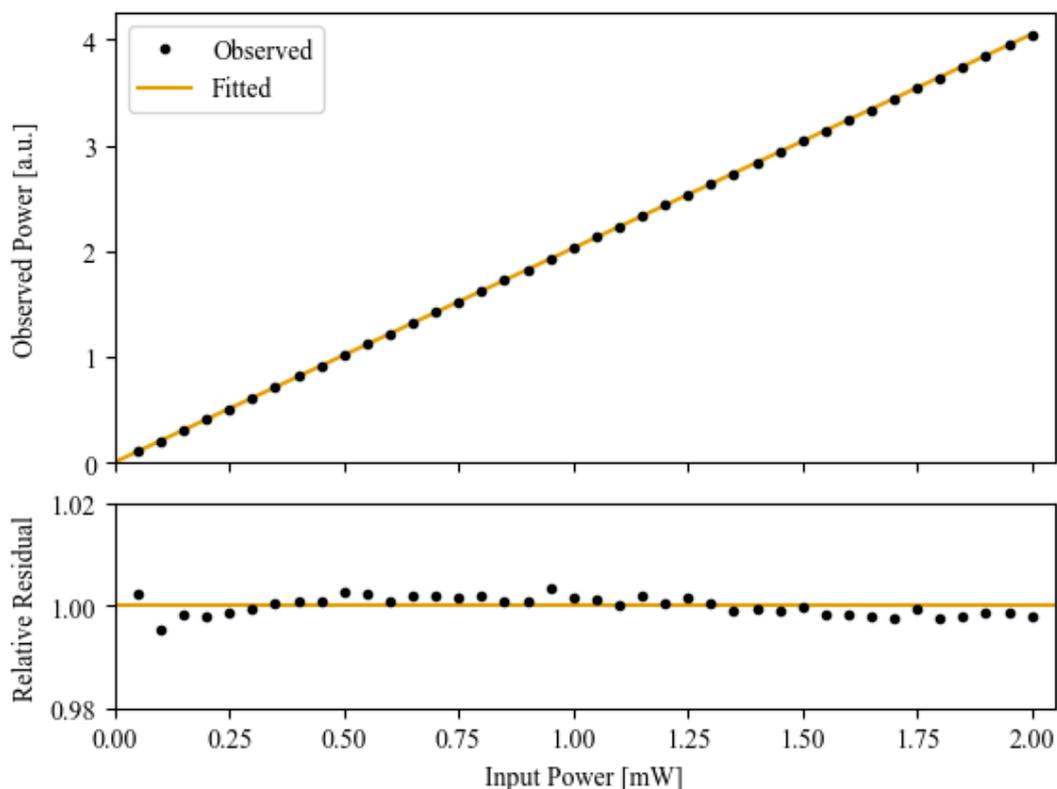


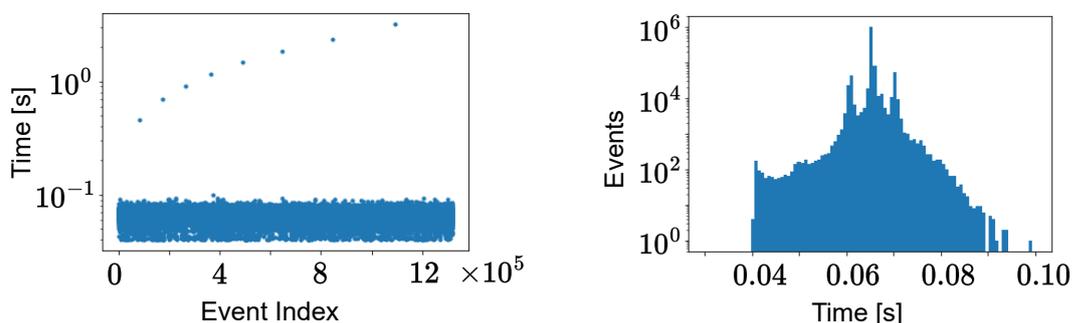
図 5.5: 入力電力と観測電力の関係。0.5% 未満の精度で線形性が確認された。

### 5.3 安定性と所要時間計測

長時間観測する上で、分光計が安定稼働できるかは重要である。また、データ取得処理の中にソフトウェア処理が存在していることから、処理の所要時間が不定である。実際にデッドタイムなくデータを取得できているかも評価する必要がある。

そこで、長時間にわたって安定してデータを取得できるか、また、実際にデッドタイムなしに分光できているかを確認するため、24 時間連続して稼働させる試験を行った。この測定では得られたスペクトル自体には興味がないため、RFSoc 2x2 ボードの 2 つの ADC 入力に対し、片方にアンテナ、もう一方には終端抵抗を接続するのみとした。FPGA からは 65.536 ms ごとにデータが送られる設定となっているため、ちょうど 24 時間となる 1318360 回データを連続に受信する設定し、所要時間を測った。所要時間の計測には CPU のシステム時間を使用した。

合計の所要時間は 24 : 00 : 44 秒であり、また、一回のデータ取得にかかる時間は図 5.6 のようになった。デッドタイムの割合は  $(5.09 \pm 0.12) \times 10^{-2}\%$  であり、デッドタイムが存在する一方、99.94% 以上の期間ではデータを取得できていることが確認できた。なお、システム時間の誤差は保守的に 1s として計算した。



(a) FPGA からのデータの取得にかかった時間 (b) FPGA からのデータの取得にかかった時間のヒストグラム

図 5.6: FPGA からのデータの取得にかかった時間。FPGA からは 65.536 ms ごとにデータが送られる設定となっているが、CPU 上で行うソフトウェア処理の時間が不定であるため、多少のふらつきが生じる。

図 5.6a を見ると、100,000 回に 1 回弱の頻度で、特異的にデータ取得に時間がかかっている。これは  $\sim 1$  s の所要時間と、徐々に増えていることから、ソフトウェア関連、特にメモリ関係が原因と予想される。ただし、これらの事象は極めて数が少なく総観測時間の 0.02% 以下に過ぎない上、正常な事象との区別が容易であるため、実用上問題になることはないと考えられる。

一方、図 5.6b を見ると、所要時間が多い事象は平均処理時間の 65 ms 以上では指数関数的に減っていくものの、0.1 s 程度の所要時間がかかっているものも存在する。例えば 90 ms 以上かかってしまった場合、次の FPGA からのデータ送信をブロックしてしまう。その場合、ADC からのデータの一部分が FPGA に送られず、次のスペクトルに送られるデータに一箇所時間不連続点が生ずる。ただし、この時間不連続点は FPGA 内のアキュムレータ回路での時間平均期間をより長くとることによって容易に解消できるため、実上の問題はない。

以上より 99.94% 以上の期間でデータを取得できていることが確認でき、ほぼデッドタイムなしでの分光が行えていることが確認できた。

## 5.4 時間効率

前節の測定で ADC が出力する信号をほぼロスなく FPGA・CPU で処理できていることを確認できた。それに加えて ADC が本当に全期間のデータを取得でき、さらに FFT 演算によって情報をロスしないことが言えれば、デッドタイムなしでの測定ができているといえる。そこで、黒体輻射の実効的な測定時間と測定されたパワーのゆらぎの間に関係があることを用いて、時間効率の測定を行った。黒体輻射を観測したときの電力  $P$  と電力のふらつき  $\Delta P$  は  $\nu$  によらず次の式で表さ

れる [23] :

$$P = Gk_B T_{\text{sys}} \Delta\nu, \quad (5.1)$$

$$\Delta P = \frac{Gk_B T_{\text{sys}} \Delta\nu}{\sqrt{\Delta\nu \epsilon \Delta t}}, \quad (5.2)$$

ここで、 $G$  はシステムのゲイン、 $T_{\text{sys}}$  は黒体放射の温度とアンプの熱雑音などに由来して決まるシステム温度、 $\Delta\nu$  は観測する周波数領域の幅、 $\Delta t$  は測定時間、 $\epsilon$  は観測の時間効率を表す。黒体放射のスペクトルを何度も同じ時間だけ測定することを考えると、その平均値は  $P$  となるが、測定値は統計的に標準偏差  $\Delta P$  でふらつく。 $\Delta P$  は測定時間を延ばすことで小さくなるが、このときの測定時間とは真に測定できた時間、すなわち  $\epsilon$  がかった  $\epsilon \Delta t$  である。この性質から、 $\epsilon$  を

$$\epsilon = \frac{1}{\Delta\nu \Delta t} \left( \frac{P}{\Delta P} \right)^2. \quad (5.3)$$

と求めることができる。

#### 5.4.1 セットアップ・測定

図 5.7 に測定のセットアップを示す。異なるパワーで測定するため、室温の黒体と液体窒素に浸して 77 K まで冷やした黒体からの放射を受信し、パワースペクトルを測定した。電波黒体には E&C エンジニアリング製の CV-3、ホーンアンテナには PASTERNAK 製の PE9856/SF-20 を用いた。なお、ホーンアンテナと LNA の間にはアイソレータ (PASTERNAK 製 PE8304) が挿入されている。LNA は前段に Low Noise Factory 製の LNF-LNR6-20A、後段に Mini-Circuits 製の ZVA-18G-S+ の計 2 台構成とした。図中では省略しているが、ゲインの調整や反射の低減のため、適宜アテネータを挿入した。分光計の構成はこれまでと同様で、LO 周波数は 13 GHz とした。測定は 65.5 ms 間を 1 測定として 1000 回、計 65.5 s をそれぞれの温度で行った。図 5.8 は室温の黒体、図 5.9 は 77 K の黒体での測定の様子である。

#### 5.4.2 結果

図 5.10 にそれぞれの温度での放射スペクトルを示す。図に見える構造はホーンアンテナやアンプによって決まるものである。また、WiFi 5.6 GHz 帯の 100ch、2.4 GHz 帯の 11ch の折り返しによってノイズが生じた帯域を解析から除外した。具体的には、5.6 GHz 帯由来で 14.394–14.414 GHz 帯、11.586–11.606 GHz 帯を、2.4 GHz 帯由来で 14.624–14.644 GHz 帯、11.356–11.376 GHz を解析から除外した。

$P$  がホワイトノイズであるとき、 $1/\epsilon$  がカイ 2 乗分布に従うため、 $\epsilon$  の期待値にはバイアスが生じる。 $\epsilon$  を直接計算する代わりに、バイアスのない  $1/\epsilon$  を求めてから  $\epsilon$  を計算する必要がある (付録 A)。そこで各周波数で、パワースペクトルの時間平均と測定値のふらつきを求め、 $1/\epsilon(\nu)$  を計算し、その平均値から  $1/\epsilon$  を求めた。図 5.11 に各周波数で求めた  $1/\epsilon$  の値を示す。室温の黒体輻

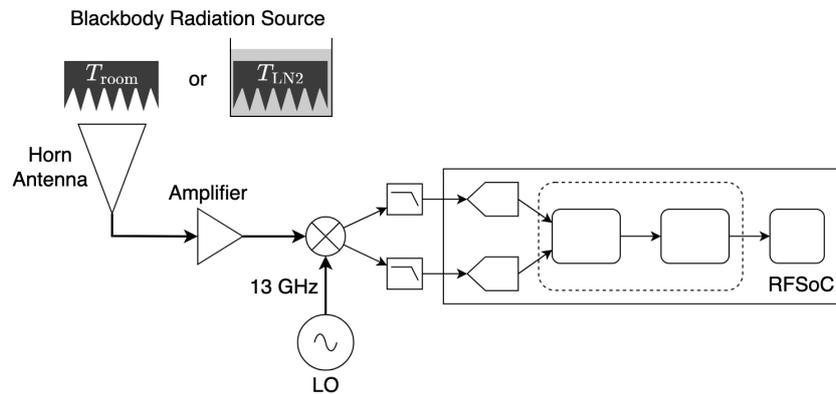


図 5.7: 時間効率の測定のためのセットアップ。黒体からの輻射をホーンアンテナで受信した。信号は LNA により増幅された後、LO 周波数 13 GHz を入力した IQ ミキサによってダウンコンバートされ、RFSoc 2x2 ボード上で分光される。室温の黒体と液体窒素に浸して 77 K まで冷やした黒体のそれぞれでパワースペクトラムを測定した。

射の測定で  $1/\epsilon = 1.00257 \pm 0.00013$ 、77 K の黒体の測定で  $1/\epsilon = 1.00169 \pm 0.00013$  という結果が得られ、どちらの測定でも  $\epsilon$  は 99.7% 以上であることがわかった。

前節の測定とこの測定の結果から、分光計が実際にデッドタイムなしに分光できていることが確認できた。

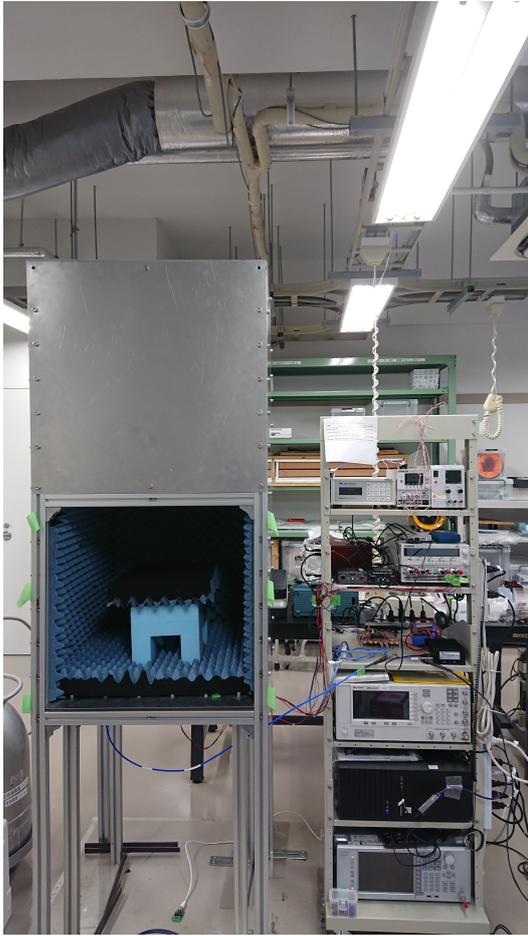


図 5.8: 測定の様子 (室温)。周囲のからの雑音を抑えるため、周囲を電波黒体を貼り付けたアルミ版で覆っている。写真では見やすさのため前面パネルを解放しているが、測定中は閉じられている。

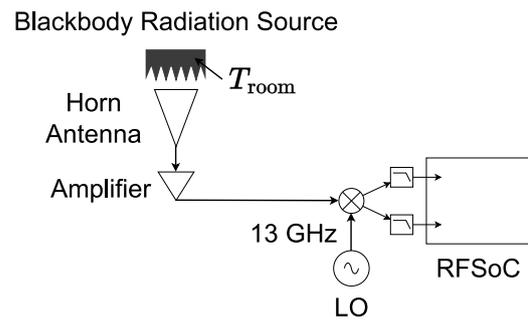
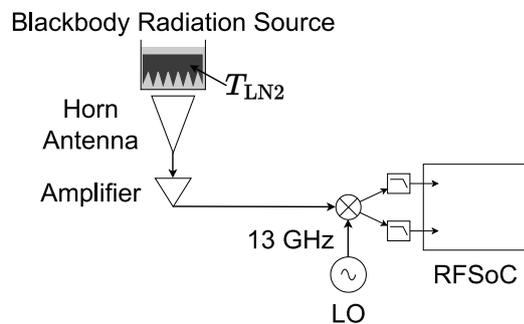


図 5.9: 測定の様子 (77 K)。電波黒体を液体窒素につけ、77 K に冷やした。容器は電波透過率の高い発砲スチロールを用いた。



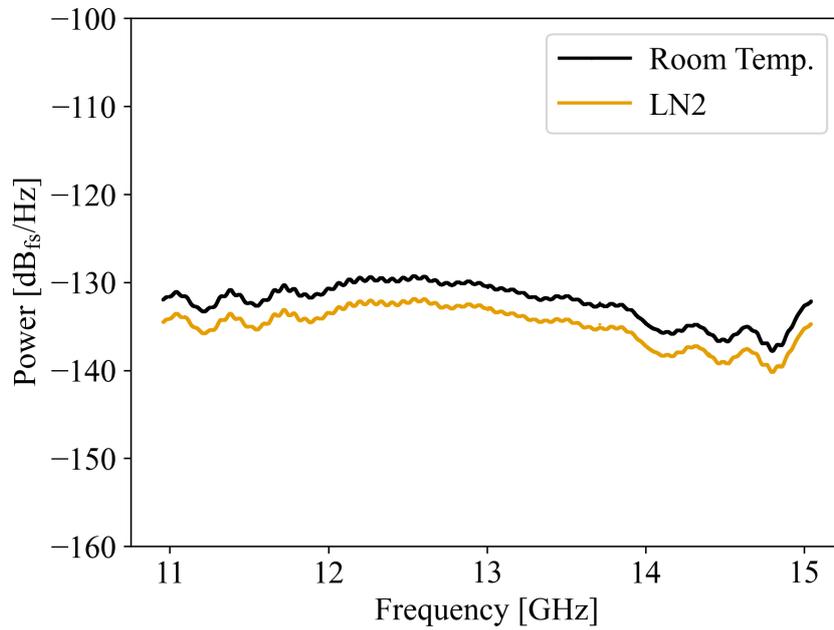
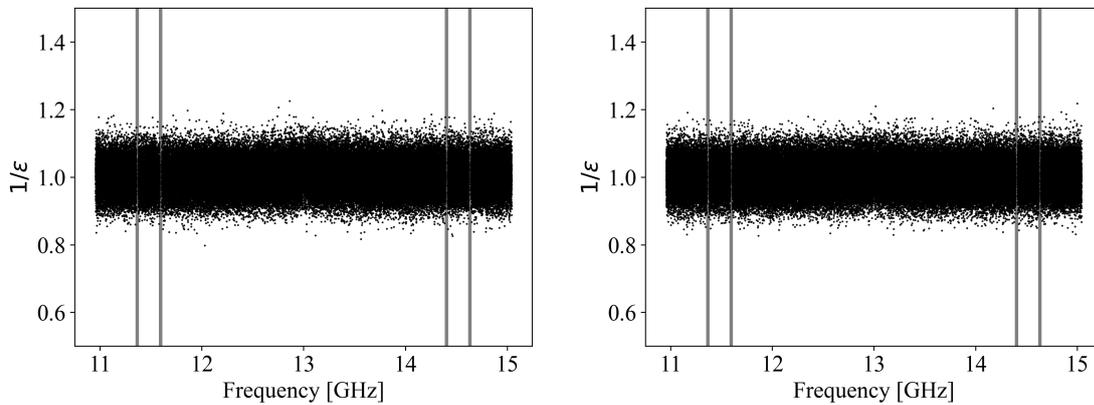


図 5.10: 室温および液体窒素下での黒体輻射のパワースペクトル。



(a) 室温の黒体輻射から求めた各周波数における  $1/\epsilon$  の測定値

(b) 77 K の黒体輻射から求めた各周波数における  $1/\epsilon$  の測定値

図 5.11: 各周波数における  $1/\epsilon$  の測定値。室温および 77 K の黒体のパワースペクトルを 65.5 s 間測定し、1000 個のパワースペクトルを得た。それぞれの周波数で、パワースペクトルの時間平均と時間変動から時間効率の逆数  $1/\epsilon$  を求めた。WiFi の折り返しノイズが入る周波数帯 (灰色の太線部分) は除外した。



## 第6章

# DP-CDM 探索における感度見込み

本章では、開発した分光計を用いることで、実際の DP-CDM 探索においてどの程度の感度の向上が見込まれるかについて議論する。

DP-CDM 探索の解析は以下のような手順で行う。まず、得られたパワースペクトルから、ある  $\nu$  (または  $m_{\text{DP}}$ ) を仮定したときの  $P_{\text{DP}}$  の大きさを fitting によって求める。次に、 $P_{\text{DP}}$  とそのふらつき  $\Delta P_{\text{DP}}$  から、 $P_{\text{DP}}$  がバックグラウンドからどれくらい卓越しているか見積もり、統計的に有意な信号といえるかを判断する。信号が見つかった場合、DP-CDM が存在する証拠となる。逆に信号が見つからなかった場合、統計的にどの程度の  $P_{\text{DP}}$  まで排除できるかを、求めた  $P_{\text{DP}}, \Delta P_{\text{DP}}$  から算出し、 $\chi$  の上限を求める。

今はある条件で測定したときに、 $\chi$  がどれくらいあれば発見できそうかを見積もりたいので、以下のように計算した。式 (1.11) より、 $P_{\text{DP}}$  の信頼度 (Confidence Level, CL) 95% での上限を  $P_{\text{DP},95\%}$  としたとき、転換光の周波数  $\nu = m_{\text{DP}}c^2/h$  における  $\chi$  の信頼度 95% での上限  $\chi_{95\%}$  は、

$$\chi_{95\%}(\nu) = 10^{-10} \times \left( \frac{P_{\text{DP},95\%}(\nu)}{6.4 \times 10^{-2} \text{ aW}} \right)^{\frac{1}{2}} \left( \frac{A_{\text{eff}}}{10 \text{ cm}^2} \right)^{-\frac{1}{2}} \left( \frac{\rho}{0.39 \text{ GeV/cm}^3} \right)^{-\frac{1}{2}} \left( \frac{\alpha}{\sqrt{2/3}} \right)^{-1} \quad (6.1)$$

と表せる。 $\nu$  から  $\nu + \Delta\nu$  の範囲のスペクトルから  $P_{\text{DP},95\%}$  を求める場合を考える。 $\Delta\nu = 1 \times 10^{-6} \times \nu$  とし、転換光は  $\nu$  から  $\nu + \Delta\nu$  の区間に一様に分布しているとする。 $P_{\text{DP}}$  を測定するときの主要なノイズは熱ノイズなので、 $\nu$  から  $\nu + \Delta\nu$  の間のノイズ  $P_{\text{noise}}(\nu, \Delta\nu)$  を

$$P_{\text{noise}}(\nu, \Delta\nu) = k_B T_{\text{sys}} \Delta\nu \quad (6.2)$$

$$\Delta P_{\text{noise}}(\nu, \Delta\nu) = \frac{k_B T_{\text{sys}} \Delta\nu}{\sqrt{\Delta\nu \Delta t}} \quad (6.3)$$

のホワイトノイズとして考えると、

$$P_{\text{DP},95\%}(\nu) = 1.96 \times \frac{k_B T_{\text{sys}} \Delta\nu}{\sqrt{\Delta\nu \Delta t}} \quad (6.4)$$

と求められる。

## 6.1 既存の実験セットアップでの感度の向上

この結果を用いて、まず先行実験 [10] で探索した 18–26.5 GHz 帯の探索において、開発した分光計を用いることで見込まれる探索感度を求める。

探索の条件は分光計を差し替える以外変化させないと仮定し、 $A_{\text{eff}} = 17.4 \text{ cm}^2$ ,  $T_{\text{sys}} = 250 \text{ K} \times 23.1\% = 57.8 \text{ K}$  とした。ただし  $\alpha$  は片偏波アンテナであることを考慮に加え、 $\alpha = \sqrt{1/3}$  とした。

$\Delta t$  は先行実験 [10] の総計測時間が 102 000 s であったことから、同等の総計測時間になるように  $\Delta t = 2.4 \times 10^4 \text{ s}$  とした。分光計の帯域幅はスプリアス領域を解析から除外することを想定して 4 GHz とした。また、測定時間を変えた場合の例として、総観測時間 30 日、300 日での予想感度も求めた、これらの計算では、これまで分光計の対応帯域の制限によって探索できなかった 26.5–33 GHz 帯も探索領域に含めた。

求めた  $\chi$  の予想感度を、図 1.4 に示す。先行実験と同等の機器・測定時間でも、分光計部分を交換するだけで先行実験の結果から  $\chi$  の感度を 4 倍以上向上できる見込みである。帯域幅 2000 倍から予想される  $\sqrt[4]{2000} = 6.69$  倍よりも少なくなっているのは、先行実験 [10] では  $\alpha = \sqrt{2/3}$  としていたことによる。また、30 GHz 付近のこれまで直接探索がなされていなかった多くの質量領域において、30 日程度の探索で宇宙論による間接的な制限を 2 桁以上超える見込みである。

## 6.2 高周波数帯の探索

DOSUE-RR グループでは現在 170–260 GHz 帯の探索を計画している。この周波数領域で見込まれる感度を考える。

この探索では楕円鏡を用いて集光させることによって  $A_{\text{eff}}$  を増大させることを予定しており  $A_{\text{eff}} \sim \pi \times 1 \times 10^2 \text{ cm}^2$ 、雑音温度は  $T = 50 \text{ K}$  を見込む。また、さらなる探索速度の高速化のため、複数台の分光計で並列に読み出すことも計画している。そこで、同時読み出し帯域幅 4 GHz の場合と 32 GHz の場合の 2 通りで比較した。探索帯域全体での探索期間は低周波数帯と同様に 102 000 s, 30 日、300 日の 3 通りを考えた。

これらの条件で求めた感度を見込みを図 6.2 に示す。30 日程度の同時読み出し帯域幅 32 GHz での探索により、太陽ダークフォトンの探索による制限 [24] と比較して 2 桁程度上回る感度での探索が行える見込みである。

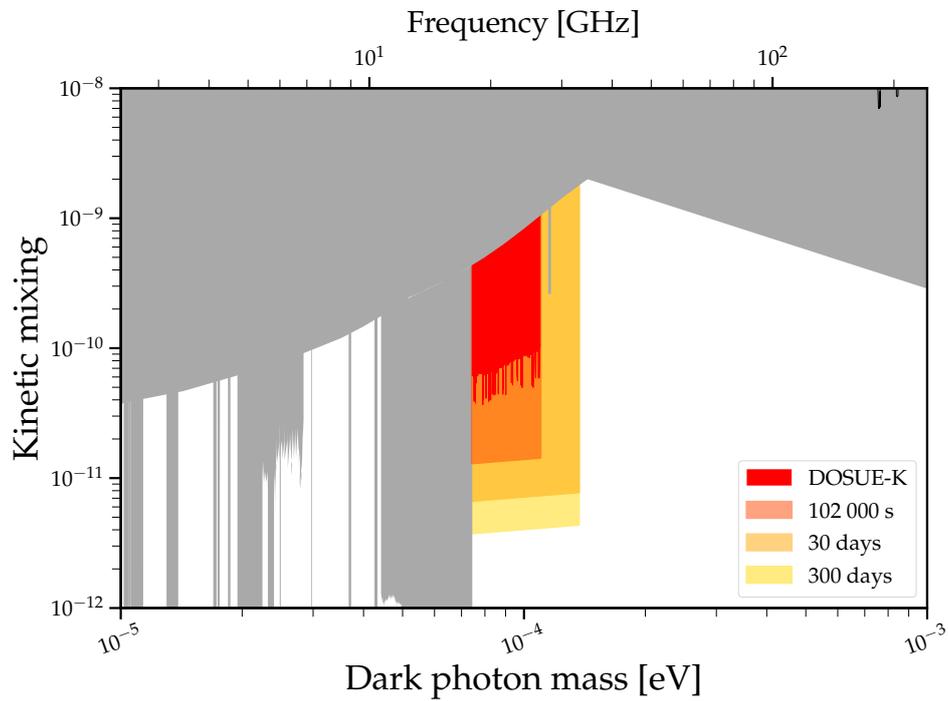


図 6.1: 分光計の更新による  $\chi$  の感度向上の見込み。各領域はそれぞれ総観測時間を 102 000 s、30 日、300 日としたときの予想感度 (95%CL) である。赤色の領域は先行実験 [10] による制限で、観測時間は合計 102 000 s である (一部の周波数領域で追加測定の結果を含んでいることに注意)。灰色の領域はその他既存の制限 [7]。

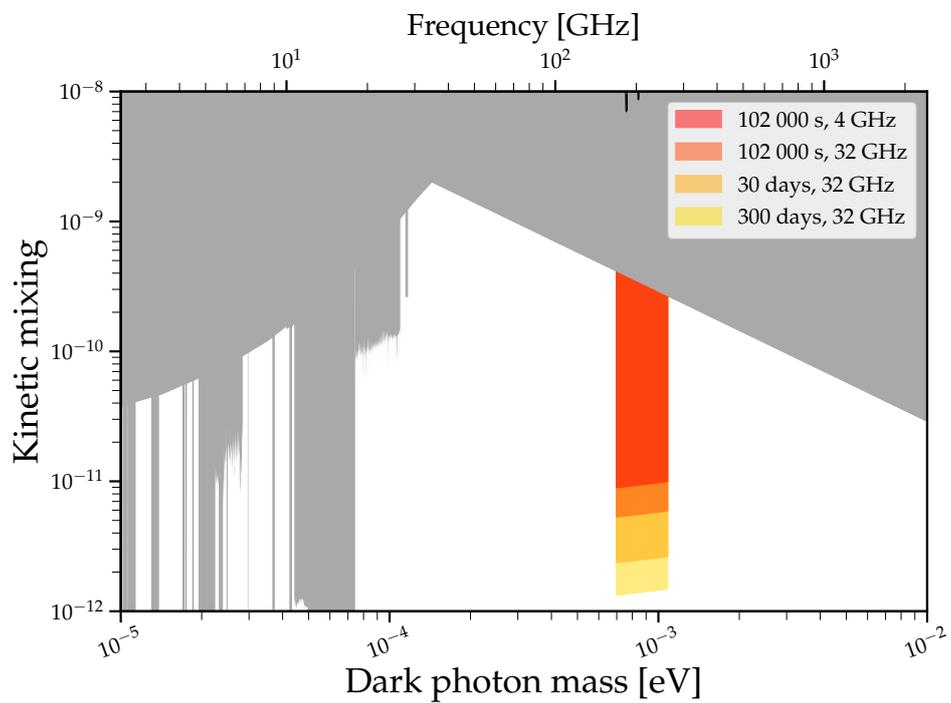


図 6.2: 170–260 GHz 帯探索における  $\chi$  の感度の見込み。各領域はそれぞれ総観測時間を 102 000 s、30 日、300 日としたときの予想感度 ( ) である。また、分光計の並列化の比較として、帯域幅 4 GHz の場合と 32 GHz の場合の二通りを考えた。灰色の領域は既存の制限 [7]。

## 第7章

# 結論

本研究では DP-CDM 探索に向けて、4 GHz の帯域幅をデッドタイムなしに分光することができる広帯域の分光計を開発した。また、典型的に  $\Delta f/f \sim 10^{-6}$  と極めて狭い転換光の周波数ピークを検出するため、16 kHz の周波数分解能を確保した。この分光計により DP-CDM 探索を効率的に行えるようになり、統計量の増加による  $\chi$  の感度の向上が見込まれる。

これらの性能を実現するため、ハードウェアや、FPGA 上の論理回路のようなファームウェア、さらにソフトウェアの設計を行った。まず、RFSoc 内蔵の 4 GSPS の ADC を 2 つと IQ ミキサを使うことで、4 GHz 幅の帯域幅を確保した。また、内蔵 FPGA 上で 16 並列の FFT 回路によって FFT 変換を行い、アキュムレータ回路によってデータ量を圧縮することで、ADC からの信号を余すことなく解析できるようにした。並列化による回路規模の増大を低減するため、省メモリな回転器を開発し、さらに汎用回転器の必要数が少ない Radix-2<sup>4</sup> アーキテクチャを採用した。PS 上のソフトウェア処理では、FPGA から PS へのデータ転送を PS と外部機器の通信が阻害しないようにするため、マルチスレッド処理を行うようにした。これらの設計の工夫により、要求性能通りに分光計を製作することができた。このような広帯域・時間効率と良い周波数分解能を併せ持つ分光計やそれを可能にした技術は、物理学研究だけでなくフーリエ変換を利用する様々な用途に有効であると考えられることから、特許出願も行った。

さらに、製作した分光計について性能評価試験を行い、DP-CDM 探索に有用であることを実証できた。具体的には、単色 RF 源を入力して予想通りのスペクトルが得られていることや、0.5% 未満の精度で線形性を確認した。また、24 時間連続できる分光計の安定性や、分光計が実際にデッドタイムなしで分光できていることを確認した。

この分光計を使うことで DP-CDM 探索の感度向上が見込まれる。例えば、先行実験 [10] から分光計を差し替えるだけで 4 倍以上  $\chi$  の感度を向上できる見込みである。さらに、今後探索を計画している 170–260 GHz 帯においても、既存の実験よりも 2 桁程度良い感度で探索できる見込みである。また、本研究で開発した分光計は DP-CDM の転換光に限らず様々な RF 信号を分光できるため、アクシオンなどの他のダークマター候補粒子の探索や、電波天文学などへの応用も期待できる。



# 謝辞

これまで2年間の研究生生活を支えてくださった皆様、ありがとうございました。

特に、田島さん、鈴木さん、安達さんの3人の指導教員の方々には研究全般にわたってお世話になりました。田島さんには、研究の内容に発表、さらには書類の書き方やスケジュール管理まで、あらゆることをご指導いただきました。田島さんのアドバイスのおかげでなんとかなった場面が数えきれないほどあります。アブストラクトの提出締め切り日の勘違いを指摘していただいていたなければ、修論を提出することすら叶わなかったかもしれません。

鈴木さんには研究テーマの決定や、FPGAでの開発手法、実験の原理といった様々なことで質問させてもらいました。なんでも丁寧に説明して下さったおかげで、安心して研究を進めることができたと思います。FPGA開発の経験が全くなかった私が開発を完遂できたのは鈴木さんのご指導あってこそです。

安達さんはとても気さくな方で、日頃の会話の中でもDOSUE-RRのことはもちろん、他の実験や物理のことについても色々教えてくださいました。特許出願という未知のことを実際にやってみることになったのも、出張中の安達さんとの会話がきっかけだったような気がします。

特許出願では指導教員の方々だけでなく、若林様をはじめとした京大の知財部門の皆様、そして松阪国際特許事務所の松阪弁理士に大変お世話になりました。出願に関わる中で特許の世界の一端を覗くことができましたが、あの複雑怪奇な制度や事情に精通しながら、かなり専門的な出願内容についてもしっかりと把握して手続きを進める姿はとても頼りになりました。

研究室の先輩方にもたくさんお世話になりました。末野さんには毎週のミーティングなどでお世話になりました。しっかり研究を進めていてしかも周囲には優しい姿は、自分もこうありたい、という一つの目標になりました。中田さんは同室だったM1の研究テーマが決まらなかった頃から面倒を見てもらいました。執筆中の差し入れも嬉しかったです。一つ上の先輩の藤中さんや武市さんは修士学生の視点で話をしてくれて、特に研究テーマを考えるときには大きな助けになりました。同期のみんなにも助けられました。それぞれ違う研究テーマでも、共に研究をする仲間でした。また同期会やりましょう。

最後に、研究室の皆様や、家族、自分と関係のあった全ての方々に感謝を捧げます。適当な性格で様々な迷惑をかけていたのではないかと思います。皆様のご辛抱のおかげでここまでやってきました。これからもまだまだご迷惑をおかけしていくとは思いますが、少しずつでも恩返しをして行きたいと思います。



## 付録 A

# 時間効率 $\epsilon$ の推定方法

5.4 節で測定した時間効率  $\epsilon$  は正規分布に従わないため、 $\epsilon$  の推定方法には注意が必要である。本章では  $\epsilon$  の推定方法について説明する。

5.4 節で述べたように熱雑音  $P$  を周波数  $\nu$  から  $\nu + \Delta\nu$  の範囲で時間  $t$  から  $t + \Delta t$  の間測定した時の測定値  $P_{\text{obs}}(\nu, t)$  は、平均値  $P$ 、標準偏差  $\Delta P$  の正規分布に従う。ただし、 $P$ 、 $\Delta P$  は式 (5.1), (5.2) で与えられる。

$P_{\text{obs}}(\nu, t)$  を  $t$  に関して  $n_t$  回、 $\nu$  に関して  $n_\nu$  回測定したとき、 $P(\nu)$  は  $P_{\text{obs}}(\nu, t)$  の (時間) 平均  $\overline{P_{\text{obs}}(\nu)}$ 、 $\Delta P(\nu)^2$  は  $P_{\text{obs}}(\nu, t)$  の  $t$  についての不偏分散  $V_{\text{obs}}(\nu)$  によって推定できる。 $P_{\text{obs}}(\nu, t)$  は分散  $P(\nu)^2/(\Delta\nu\epsilon\Delta t)$  の正規分布に従うので、 $V(P_{\text{obs}})(\nu)^2 \cdot \Delta\nu\epsilon\Delta t/P(\nu)^2$  は自由度  $n_t - 1$  のカイ 2 乗分布に従う。よって、確率変数  $X_{\text{obs}}(\nu)$  を  $X_{\text{obs}}(\nu) = V(P_{\text{obs}})(\nu)^2 \cdot \Delta\nu\Delta t/(P(\nu)^2 \cdot (n_t - 1))$  とおくと、 $X_{\text{obs}}(\nu)$  の期待値  $E$  と分散  $V$  は

$$E(X_{\text{obs}}) = \frac{1}{\epsilon} \quad (\text{A.1})$$

$$V(X_{\text{obs}}) = \frac{2}{(n_t - 1)\epsilon^2} \quad (\text{A.2})$$

となる。つまり、 $X_{\text{obs}}$  の測定から  $1/\epsilon$  を推定できる。

実際には真の平均電力  $P(\nu)$  は分からないので、 $\overline{P_{\text{obs}}(\nu)}$  を使って  $X_{\text{obs}}(\nu)$  を求める必要があるが、実用上  $P(\nu)$  で求めた  $X_{\text{obs}}(\nu)$  の分布とみなして問題ない。例として、 $\Delta\nu = 31.25$  kHz,  $\Delta t = 65.536$  ms,  $\epsilon = 0.5$ ,  $n_t = 10$  としたときの  $X_{\text{obs}}(\nu)$  のモンテカルロ・シミュレーションと、カイ 2 乗分布から求めた  $X_{\text{obs}}(\nu)$  の理論分布を図 A.1 に示す。

$n_\nu$  個の  $X_{\text{obs}}(\nu)$  の平均値から求めた  $1/\epsilon$  の推定値を  $(1/\epsilon)_{\text{obs}} = \overline{X_{\text{obs}}}$  とすると、

$$E[(1/\epsilon)_{\text{obs}}] = \frac{1}{\epsilon} \quad (\text{A.3})$$

$$V[(1/\epsilon)_{\text{obs}}] = \frac{2}{(n_\nu - 1)(n_t - 1)\epsilon^2} \quad (\text{A.4})$$

となる。ただし、 $V[(1/\epsilon)_{\text{obs}}]$  は不偏分散である。また、 $n_\nu, n_t$  が十分に大きければ正規分布と見做せる。 $\epsilon_{\text{obs}} = 1/(1/\epsilon)_{\text{obs}}$  として  $\epsilon_{\text{obs}}$  を求めることで、バイアスなく  $\epsilon$  を推定することができる。

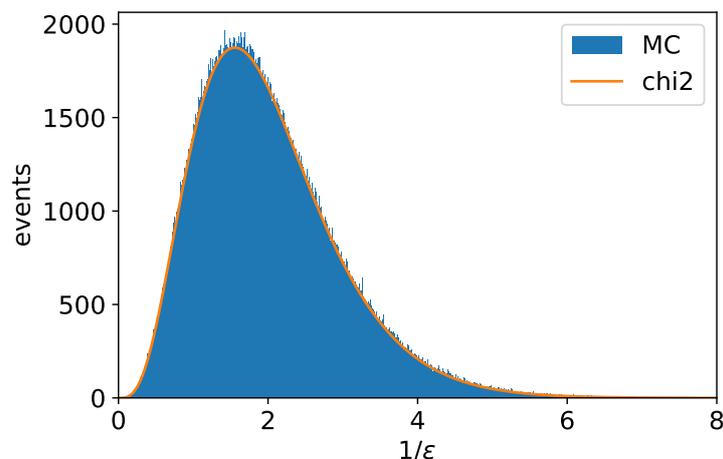


図 A.1:  $\Delta\nu = 31.25$  kHz,  $\Delta t = 65.536$  ms,  $\epsilon = 0.5$ ,  $n_t = 10$  としたときの  $X_{\text{obs}}(\nu)$  のモンテカルロ・シミュレーションと、カイ 2 乗分布から求めた  $X_{\text{obs}}(\nu)$  の理論分布。理論分布はシミュレーションに合わせて規格化してる。

一方、直接  $\epsilon$  を推定、すなわち  $\epsilon'_{\text{obs}} = \overline{1/X}$  として  $\epsilon'_{\text{obs}}$  を求めると推定にバイアスが生じる。例として、図 A.2 に 5.4 節の測定と同じ  $\Delta\nu$ ,  $\Delta t$ ,  $n_\nu$ ,  $n_t$  の値でモンテカルロ・シミュレーションを 100 回行ったときの、 $\epsilon_{\text{obs}}$  と  $\epsilon'_{\text{obs}}$  の分布を示す、ただし、真の  $\epsilon$  は  $\epsilon = 1$  とした。5.4 節の測定の場合、 $\epsilon_{\text{obs}}$  がバイアスなく推定できている一方、 $\epsilon'_{\text{obs}}$  は 0.2% 程度大きく推定してしまっていることが分かる。

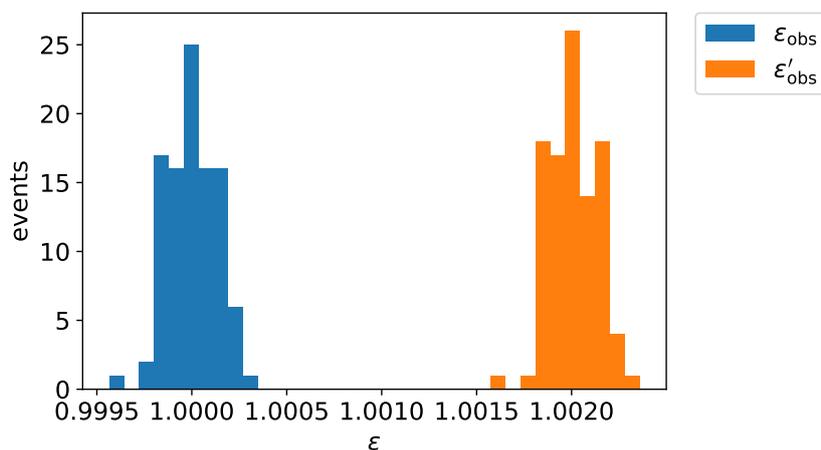


図 A.2:  $(1/\epsilon)_{\text{obs}} = \overline{X_{\text{obs}}}$  から求めた推定値  $\epsilon_{\text{obs}}$  と、 $\epsilon'_{\text{obs}} = \overline{1/X}$  として直接求めた  $\epsilon'_{\text{obs}}$ 。  $\epsilon_{\text{obs}}$  がバイアスなく推定できている一方、 $\epsilon'_{\text{obs}}$  では 0.2% 程度大きく推定してしまっている。

## 参考文献

- [1] N. Aghanim, Y. Akrami, M. Ashdown, J. Aumont, C. Baccigalupi, M. Ballardini, A. J. Banday, R. B. Barreiro, N. Bartolo, S. Basak, R. Battye, K. Benabed, J.-P. Bernard, M. Bersanelli, P. Bielewicz, J. J. Bock, J. R. Bond, J. Borrill, F. R. Bouchet, F. Boulanger, M. Bucher, C. Burigana, R. C. Butler, E. Calabrese, J.-F. Cardoso, J. Carron, A. Challinor, H. C. Chiang, J. Chluba, L. P. L. Colombo, C. Combet, D. Contreras, B. P. Crill, F. Cuttaia, P. d. Bernardis, G. d. Zotti, J. Delabrouille, J.-M. Delouis, E. Di Valentino, J. M. Diego, O. Doré, M. Douspis, A. Ducout, X. Dupac, S. Dusini, G. Efstathiou, F. Elsner, T. A. Enßlin, H. K. Eriksen, Y. Fantaye, M. Farhang, J. Fergusson, R. Fernandez-Cobos, F. Finelli, F. Forastieri, M. Frailis, A. A. Fraisse, E. Franceschi, A. Frolov, S. Galeotta, S. Galli, K. Ganga, R. T. Génova-Santos, M. Gerbino, T. Ghosh, J. González-Nuevo, K. M. Górski, S. Gratton, A. Gruppuso, J. E. Gudmundsson, J. Hamann, W. Handley, F. K. Hansen, D. Herranz, S. R. Hildebrandt, E. Hivon, Z. Huang, A. H. Jaffe, W. C. Jones, A. Karakci, E. Keihänen, R. Keskitalo, K. Kiiveri, J. Kim, T. S. Kisner, L. Knox, N. Krachmalnicoff, M. Kunz, H. Kurki-Suonio, G. Lagache, J.-M. Lamarre, A. Lasenby, M. Lattanzi, C. R. Lawrence, M. Le Jeune, P. Lemos, J. Lesgourgues, F. Levrier, A. Lewis, M. Liguori, P. B. Lilje, M. Lilley, V. Lindholm, M. López-Cañiego, P. M. Lubin, Y.-Z. Ma, J. F. Macías-Pérez, G. Maggio, D. Maino, N. Mandolesi, A. Mangilli, A. Marcos-Caballero, M. Maris, P. G. Martin, M. Martinelli, E. Martínez-González, S. Matarrese, N. Mauri, J. D. McEwen, P. R. Meinhold, A. Melchiorri, A. Mennella, M. Migliaccio, M. Millea, S. Mitra, M.-A. Miville-Deschênes, D. Molinari, L. Montier, G. Morgante, A. Moss, P. Natoli, H. U. Nørgaard-Nielsen, L. Pagano, D. Paoletti, B. Partridge, G. Patanchon, H. V. Peiris, F. Perrotta, V. Pettorino, F. Piacentini, L. Polastri, G. Polenta, J.-L. Puget, J. P. Rachen, M. Reinecke, M. Remazeilles, A. Renzi, G. Rocha, C. Rosset, G. Roudier, J. A. Rubiño-Martín, B. Ruiz-Granados, L. Salvati, M. Sandri, M. Savelainen, D. Scott, E. P. S. Shellard, C. Sirignano, G. Sirri, L. D. Spencer, R. Sunyaev, A.-S. Suur-Uski, J. A. Tauber, D. Tavagnacco, M. Tenti, L. Toffolatti, M. Tomasi, T. Trombetti, L. Valenziano, J. Valiviita, B. Van Tent, L. Vibert, P. Vielva, F. Villa, N. Vittorio, B. D. Wandelt, I. K. Wehus, M. White, S. D. M. White, A. Zacchei, and

- A. Zonca. Planck2018 results: Vi. cosmological parameters. *Astronomy & Astrophysics*, Vol. 641, p. A6, September 2020.
- [2] P. Arias, D. Cadamuro, M. Goodsell, J. Jaeckel, J. Redondo, and A. Ringwald. Wispy cold dark matter. *Journal of Cosmology and Astroparticle Physics*, Vol. 2012, No. 06, p. 013 – 013, June 2012.
- [3] P. W. Graham, J. Mardon, and S. Rajendran. Vector dark matter from inflationary fluctuations. *Physical Review D*, Vol. 93, No. 10, May 2016.
- [4] D. Horns, J. Jaeckel, A. Lindner, A. Lobanov, J. Redondo, and A. Ringwald. Searching for wispy cold dark matter with a dish antenna. *Journal of Cosmology and Astroparticle Physics*, Vol. 2013, No. 04, p. 016 – 016, April 2013.
- [5] J. Jaeckel and J. Redondo. An antenna for directional detection of wispy dark matter. *Journal of Cosmology and Astroparticle Physics*, Vol. 2013, No. 11, p. 016 – 016, November 2013.
- [6] R. Catena and P. Ullio. A novel determination of the local dark matter density. *Journal of Cosmology and Astroparticle Physics*, Vol. 2010, No. 08, p. 004 – 004, August 2010.
- [7] C. O’Hare. cajohare/axionlimits: Axionlimits. <https://cajohare.github.io/AxionLimits/>, July 2020.
- [8] J. Suzuki, Y. Inoue, T. Horie, and M. Minowa. Hidden photon cdm search at tokyo. *Suzuki*, Vol. Jun’ya ”Hidden Photon CDM Search at Tokyo” in Proceedings, pp. 22 Jun 2015–26 Jun 2015 ; DESY, 2015.
- [9] F. Bajjali, S. Dornbusch, M. Ekmedžić, D. Horns, C. Kasemann, A. Lobanov, A. Mkrtchyan, L. H. Nguyen, M. Tluczykont, G. Tuccari, J. Ulrichs, G. Wieching, and A. Zensus. First results from brass-p broadband searches for hidden photon dark matter. *Journal of Cosmology and Astroparticle Physics*, Vol. 2023, No. 08, p. 077, August 2023.
- [10] S. Kotaka, S. Adachi, R. Fujinaka, S. Honda, H. Nakata, Y. Seino, Y. Sueno, T. Sumida, J. Suzuki, O. Tajima, and S. Takeichi. Search for dark photon dark matter in the mass range 74–110  $\mu$  eV with a cryogenic millimeter-wave receiver. *Physical Review Letters*, Vol. 130, No. 7, February 2023.
- [11] S. Adachi, R. Fujinaka, Y. Muto, H. Nakata, Y. Sueno, T. Sumida, J. Suzuki, O. Tajima, H. Takeuchi, and S. Honda. Search for dark photon dark matter in the mass range 41–74  $\mu$ eV using millimeter-wave receiver and radioshielding box. *Physical Review D*, Vol. 109, No. 1, January 2024.
- [12] Mini-Circuits. Vlf-1800+. <https://www.minicircuits.com/WebStore/dashboard.html?model=VLFG-1800%2B>, 2024.
- [13] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, Vol. 19, No. 90, p. 297 – 301, 1965.
- [14] G. O’ Leary. Nonrecursive digital filtering using cascade fast fourier transformers. *IEEE*

- 
- Transactions on Audio and Electroacoustics*, Vol. 18, No. 2, p. 177 – 183, June 1970.
- [15] H. Groginsky and G. Works. A pipeline fast fourier transform. *IEEE Transactions on Computers*, Vol. C – 19, No. 11, p. 1015 – 1019, November 1970.
- [16] M. Garrido. A survey on pipelined fft hardware architectures. *Journal of Signal Processing Systems*, Vol. 94, No. 11, p. 1345 – 1364, July 2021.
- [17] J. E. Volder. The cordic trigonometric computing technique. *IRE Transactions on Electronic Computers*, Vol. EC-8, No. 3, p. 330 – 334, September 1959.
- [18] R. Hartley. Optimization of canonic signed digit multipliers for filter design. In *1991 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 1991.
- [19] Despain. Very fast fourier transform algorithms hardware for implementation. *IEEE Transactions on Computers*, Vol. C – 28, No. 5, p. 333 – 341, May 1979.
- [20] S. He and M. Torkelson. Design and implementation of a 1024-point pipeline fft processor. In *Proceedings of the IEEE 1998 Custom Integrated Circuits Conference (Cat. No.98CH36143)*, CICC-98. IEEE.
- [21] J.-Y. OH. New radix-2 to the 4th power pipeline fft processor. *IEICE Transactions on Electronics*, Vol. E88-C, No. 8, p. 1740 – 1746, August 2005.
- [22] M. Garrido, J. Grajal, M. A. Sanchez, and O. Gustafsson. Pipelined radix- $2^k$  feedforward fft architectures. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 21, No. 1, p. 23 – 32, January 2013.
- [23] 坪井昌人. 宇宙の観測 II: 電波天文学. July 2020.
- [24] E. Aprile, K. Abe, F. Agostini, S. Ahmed Maouloud, M. Alfonsi, L. Althueser, E. Angelino, J. R. Angevaere, V. C. Antochi, D. Antón Martin, F. Arneodo, L. Baudis, A. L. Baxter, L. Bellagamba, A. Bernard, R. Biondi, A. Bismark, A. Brown, S. Bruenner, G. Bruno, R. Budnik, C. Capelli, J. M. R. Cardoso, D. Cichon, B. Cimmino, M. Clark, A. P. Colijn, J. Conrad, J. J. Cuenca-García, J. P. Cussonneau, V. D’ Andrea, M. P. Decowski, P. Di Gangi, S. Di Pede, A. Di Giovanni, R. Di Stefano, S. Diglio, A. Elykov, S. Farrell, A. D. Ferella, H. Fischer, W. Fulgione, P. Gaemers, R. Gaior, M. Galloway, F. Gao, R. Glade-Beucke, L. Grandi, J. Grigat, A. Higuera, C. Hils, L. Hoetzsch, J. Howlett, M. Iacovacci, Y. Itow, J. Jakob, F. Joerg, A. Joy, N. Kato, P. Kavargin, S. Kazama, M. Kobayashi, G. Koltman, A. Kopec, H. Landsman, R. F. Lang, L. Levinson, I. Li, S. Li, S. Liang, S. Lindemann, M. Lindner, K. Liu, F. Lombardi, J. Long, J. A. M. Lopes, Y. Ma, C. Macolino, J. Mahlstedt, A. Mancuso, L. Manenti, A. Manfredini, F. Marignetti, T. Marrodán Undagoitia, K. Martens, J. Masbou, D. Masson, E. Masson, S. Mastroianni, M. Messina, K. Miuchi, K. Mizukoshi, A. Molinario, S. Moriyama, K. Morã, Y. Mosbacher, M. Murra, J. Müller, K. Ni, U. Oberlack, B. Paetsch, J. Palacio, R. Peres, J. Pienaar, M. Pierre, V. Pizzella, G. Plante, J. Qi, J. Qin, D. Ramírez García, S. Reichard, A. Rocchetti, N. Rupp, L. Sanchez, J. M. F. d. Santos, I. Sarnoff, G. Sar-

torelli, J. Schreiner, D. Schulte, H. Schulze Eißing, M. Schumann, L. Scotto Lavina, M. Selvi, F. Semeria, P. Shagin, S. Shi, E. Shockley, M. Silva, H. Simgen, A. Takeda, P.-L. Tan, A. Terliuk, D. Thers, F. Toschi, G. Trincherro, C. Tunnell, F. Tönnies, K. Valerius, G. Volta, Y. Wei, C. Weinheimer, M. Weiss, D. Wenz, C. Wittweg, T. Wolf, Z. Xu, M. Yamashita, L. Yang, J. Ye, L. Yuan, G. Zavattini, Y. Zhang, M. Zhong, T. Zhu, and J. P. Zopounidis. Emission of single and few electrons in xenon1t and limits on light dark matter. *Physical Review D*, Vol. 106, No. 2, July 2022.