

超伝導検出器MKIDsを多重読み出しする FPGA ロジックの開発

計測システム研究会2012@東京大学
～開発の現状を知り将来について議論する～

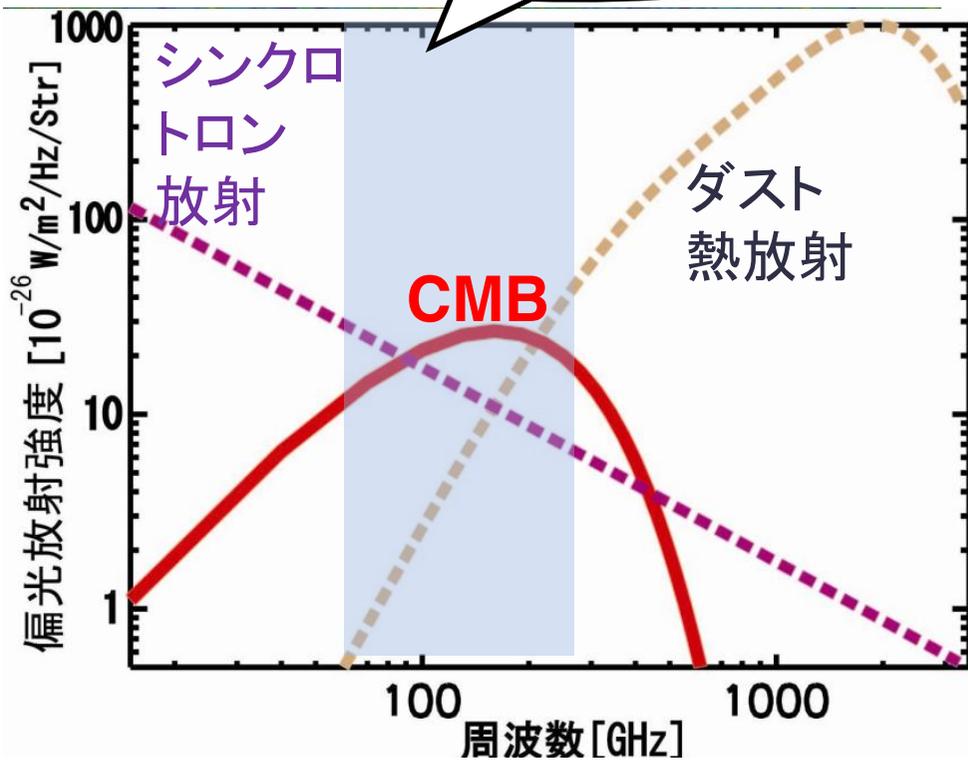
岐部 佳朗
岡山大学

目次

- Introduction
 - 宇宙マイクロ波背景放射と偏光成分
 - MKIDs
- 開発中の読み出し系
- まとめ

宇宙マイクロ波背景放射と偏光成分

周波数依存性で前景放射と分離
→ 複数の周波数帯域

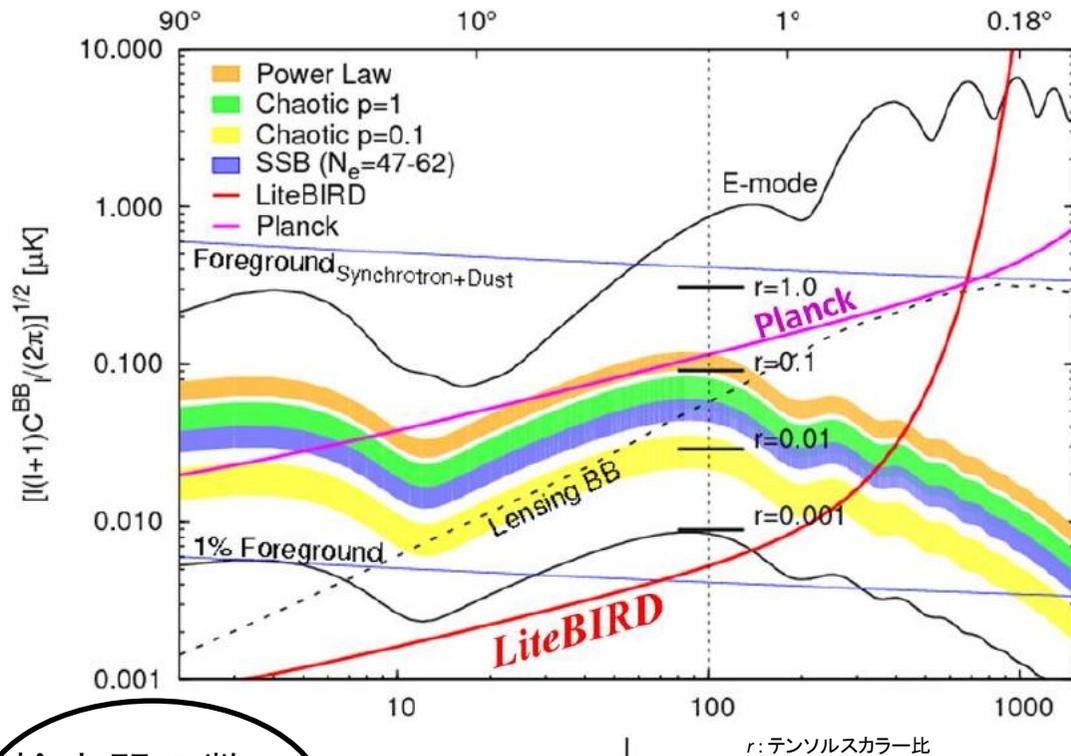


インフレーションモデルの検証
特徴的な偏光成分 (Bモード) の精密測定



高感度の検出器が多数必要！

理論モデルとの比較



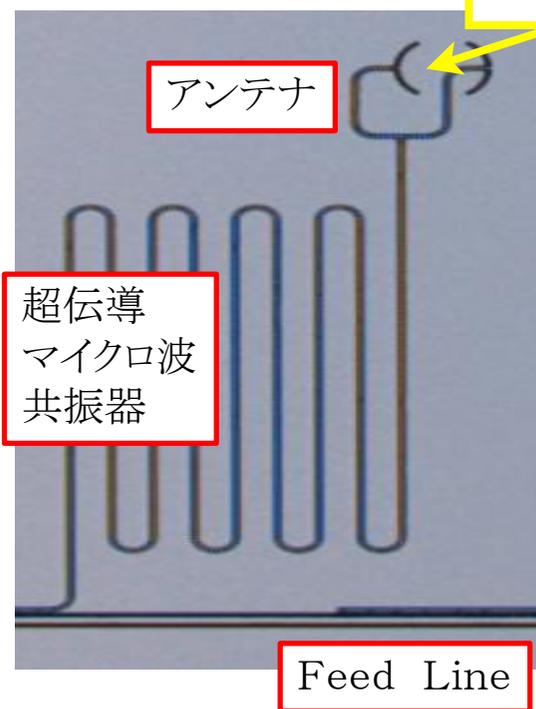
検出器の数で
統計を稼ぐ

帯域 = 50 ~ 270 GHz
検出器数 = 2×10^3 pixels
感度: NEP = 10^{-18} W/ $\sqrt{\text{Hz}}$

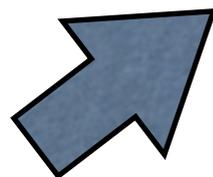
高感度の超伝導検出器

→ 検出器の候補の一つ

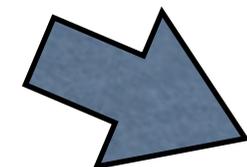
CMB



振幅の変化

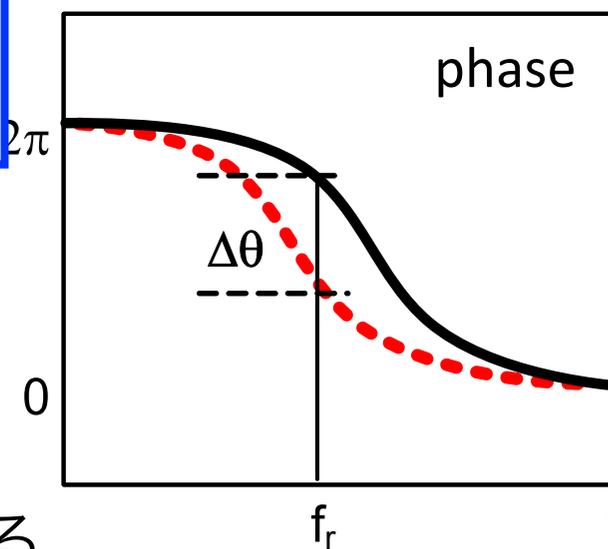
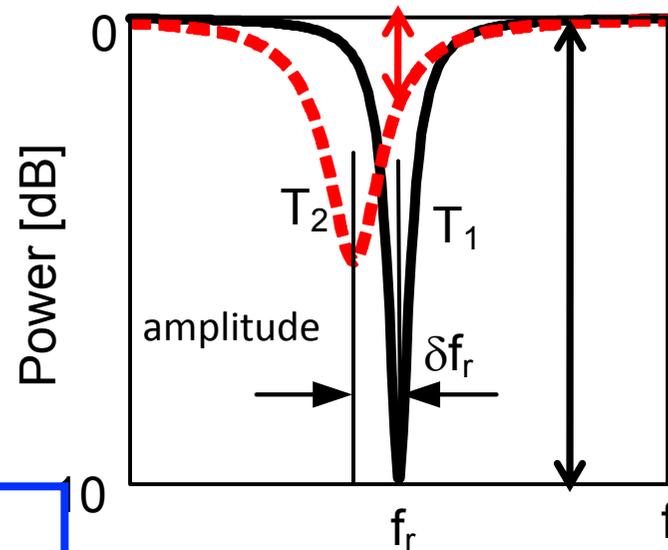


共振器のKinetic Inductance が変化する
→ 共振周波数が変化



位相の変化

共振周波数は共振器の長さなどで決まる

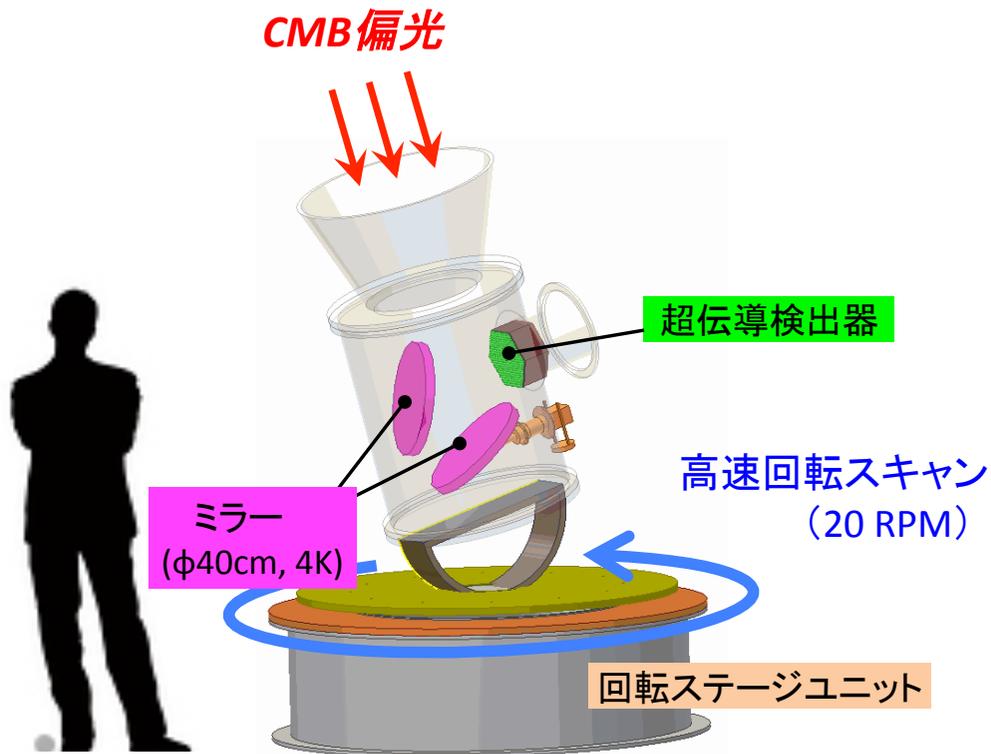


条件の異なる共振器を feed line にそって並べる

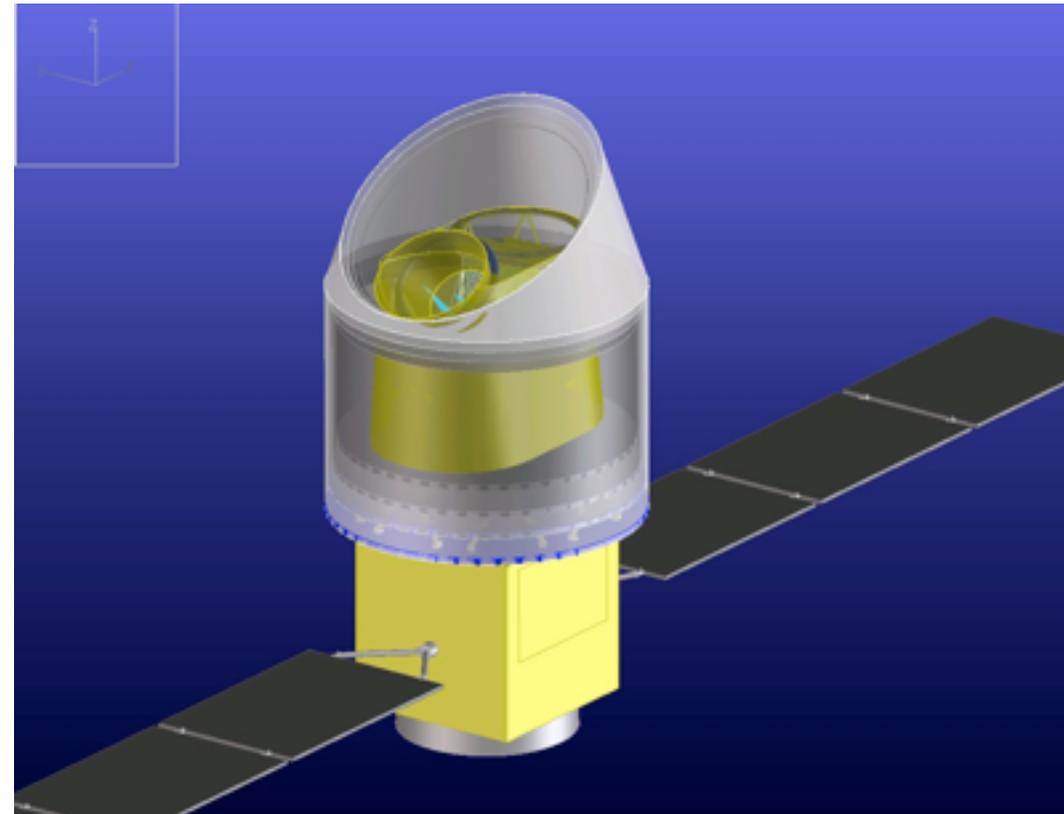
→ 一本のケーブルで多素子の読み出しが可能！

搭載予定の検出器

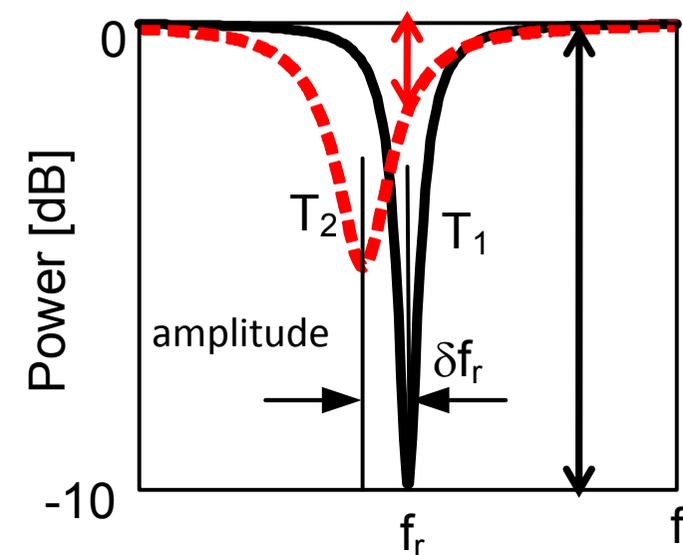
地上観測：GroundBIRD



宇宙での観測：LiteBIRD



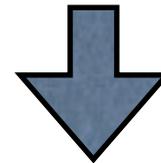
どういった読み出し系を目指すのか



専用のハードウェアで最大限のノイズ低減

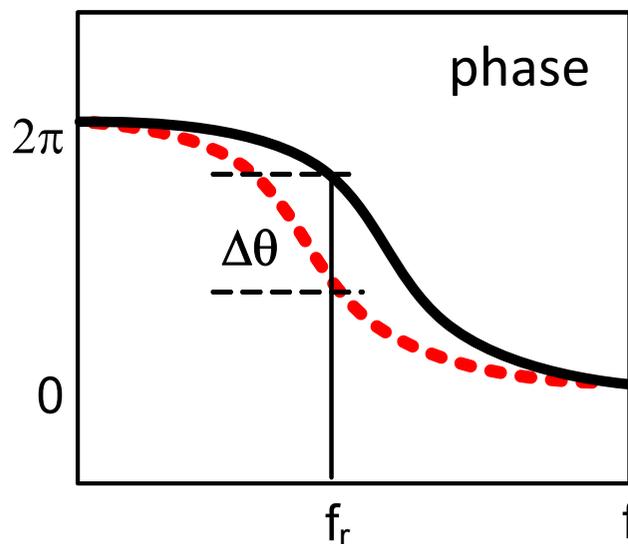
ダイナミックレンジが広い

→大きな信号にも対応可能



FFT を使った読み出しではなく.....

→共振周波数を追いかける (Tracking) !



振幅、位相をモニター

→位相変化を共振周波数の変動に変換

共振周波数の追従は加速器制御では

よく使われる手法

Tracking を実装

FPGA :Vertex-4 (XC4VSX35-10FF668)

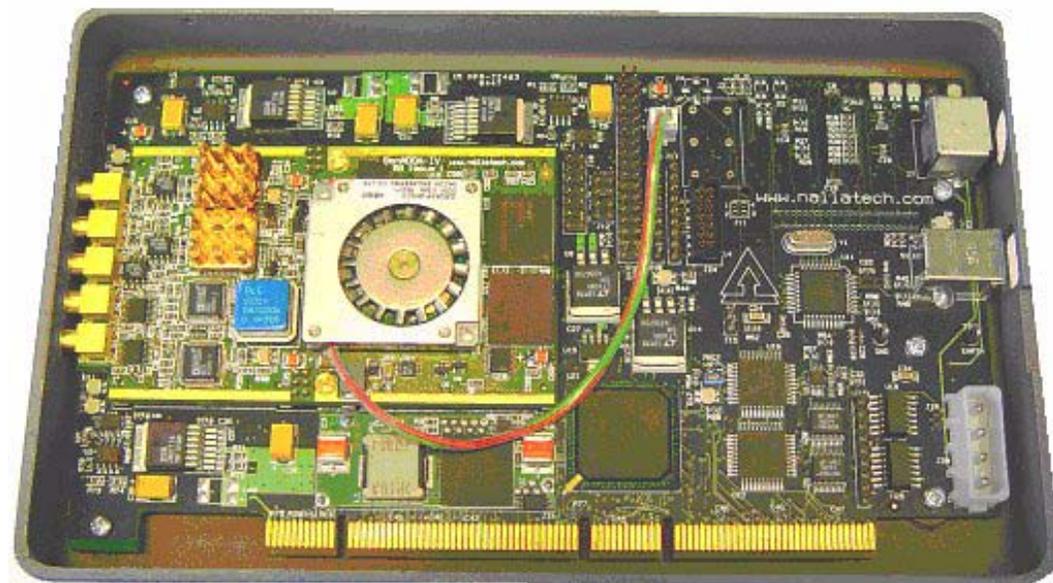
Clock: 105 MHz (on board oscillator)

DAC: 2ch, 14-bit, 160 Msps

ADC: 2ch, 14-bit, 105 Msps

Memory: 2 SRAM(512K×32-bit)

Interface: 3.3V/5V PCI 32-bit/33-MHz,
USB v1.1



Xstream DSP Development Kit-IV
(Xilinx FPGA Vertex-4)

現状、データや命令のやり取りはUSB経由で行っている

読み出し方は二種類：

DMA 読み出し（速い）、Register 読み出し（遅い）

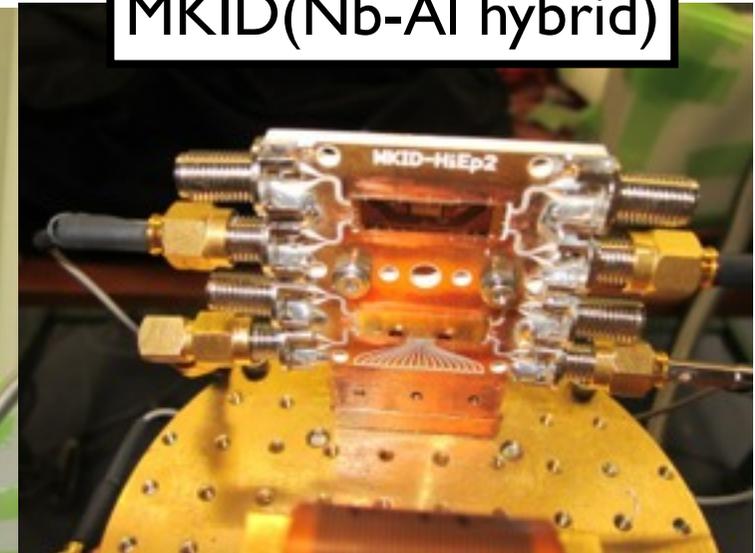
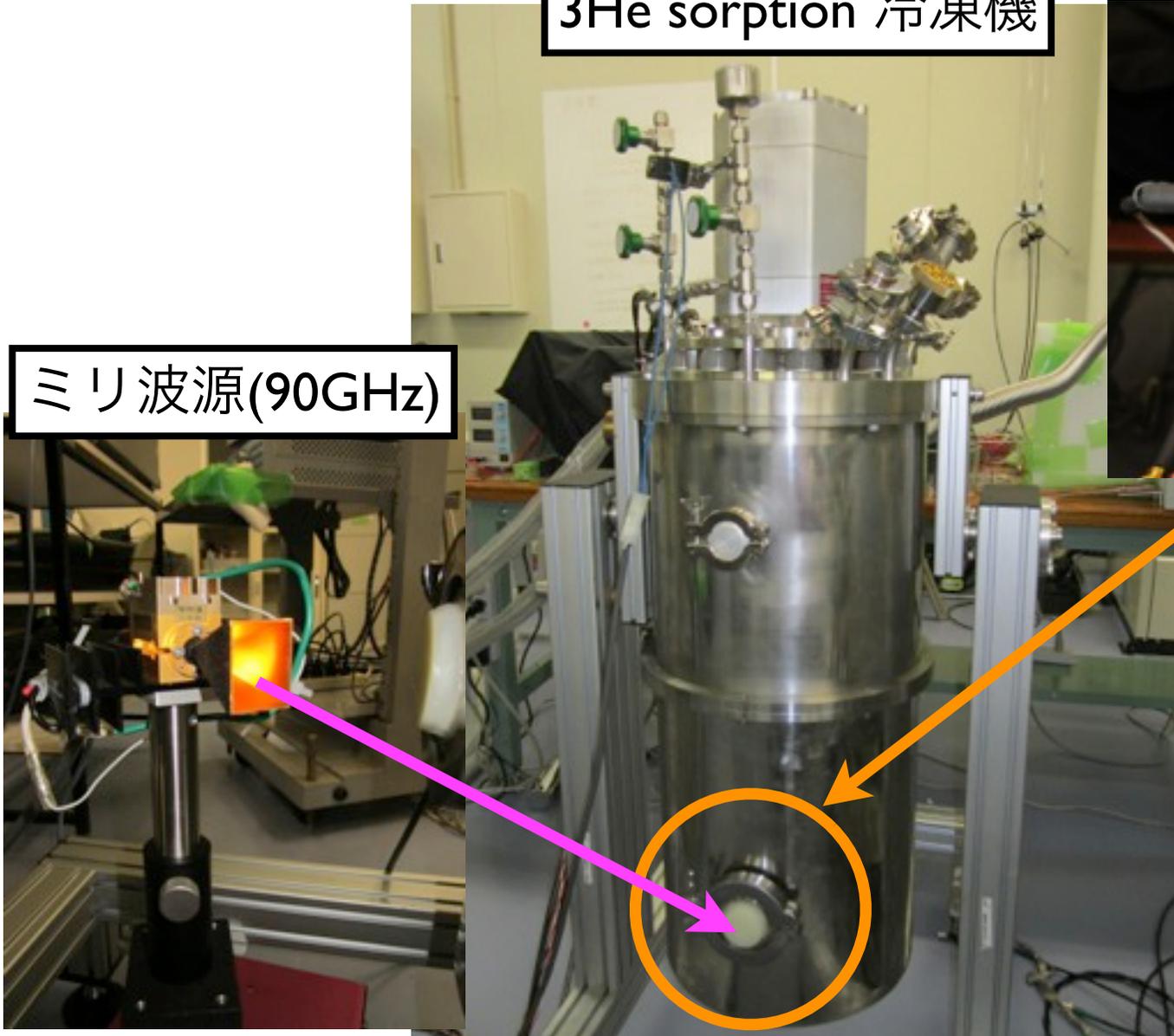
Tracking を実装

^3He sorption 冷凍機@KEK でミリ波を当てて実験

3He sorption 冷凍機

MKID(Nb-Al hybrid)

ミリ波源(90GHz)

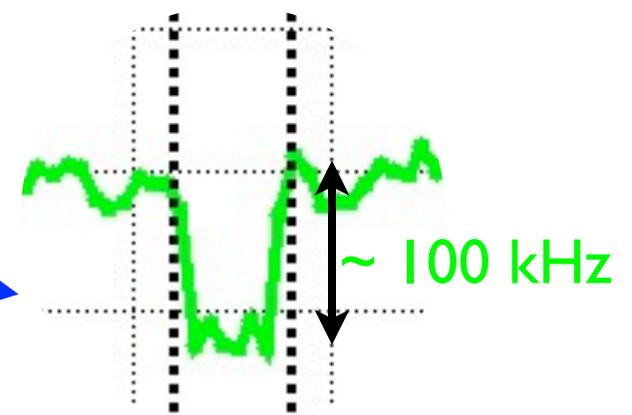
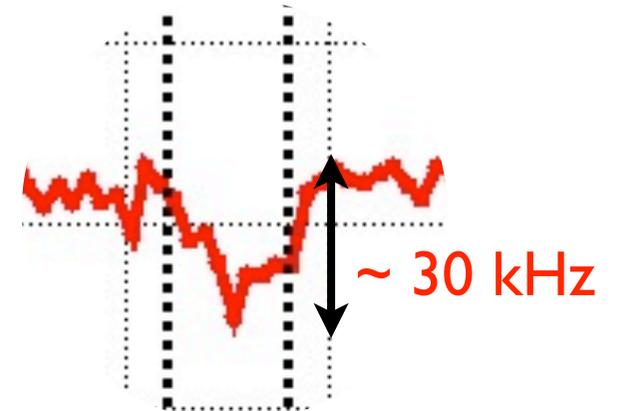
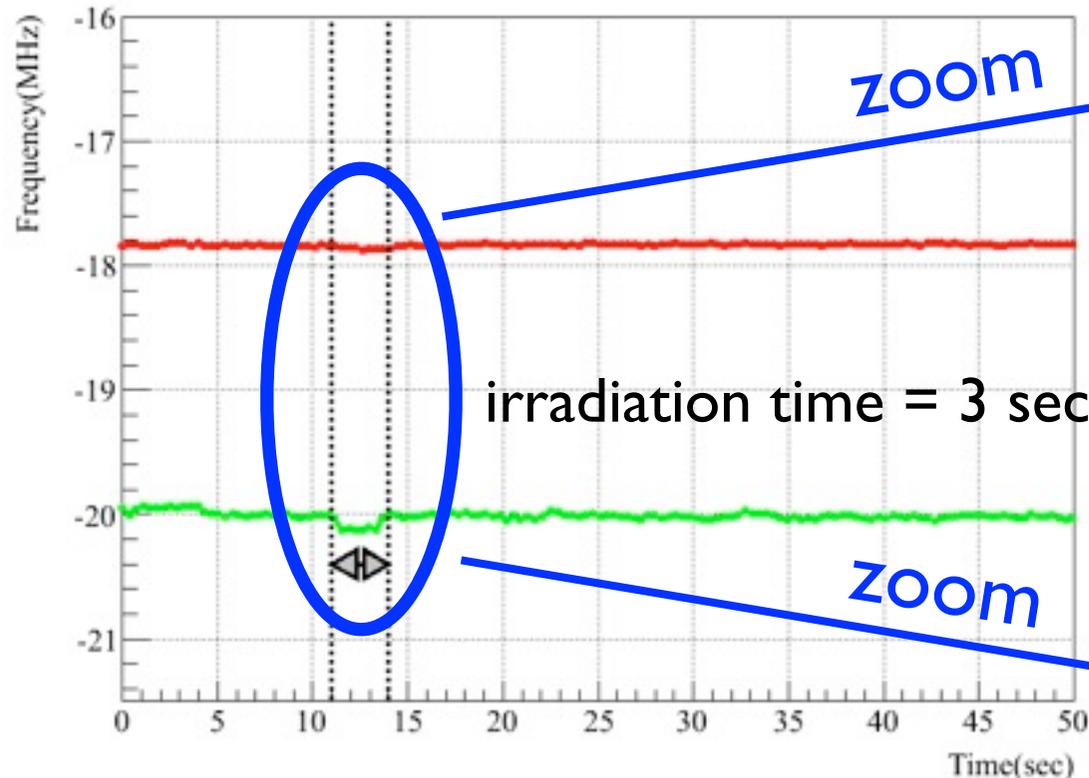


MKID を ~ 0.3 Kまで
冷やして、ミリ波を
当てる

Tracking を実装

^3He sorption 冷凍機@KEK でミリ波を当てて実験
readout rate(PCへの読み出し速度) ~ 17 Hz

共振周波数の時間変化



2 ch での Tracking は実装できた

→ MKID の基礎特性 (ノイズなど) も測ってみよう

Tracking を実装.....したけど

本来の目的：

HEMT の $1/f$ noise ~ 1 kHzを測定
readout rate w/ FIFO ~ 6.4 kHz (1.28 sec.)

Spike noise が見える.....

→今までこのような信号は見えていない

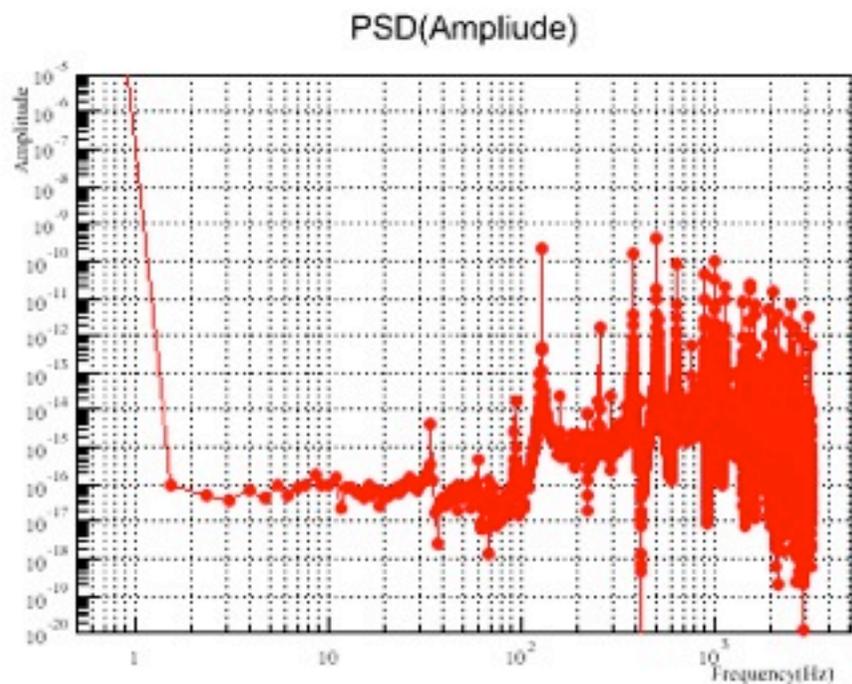
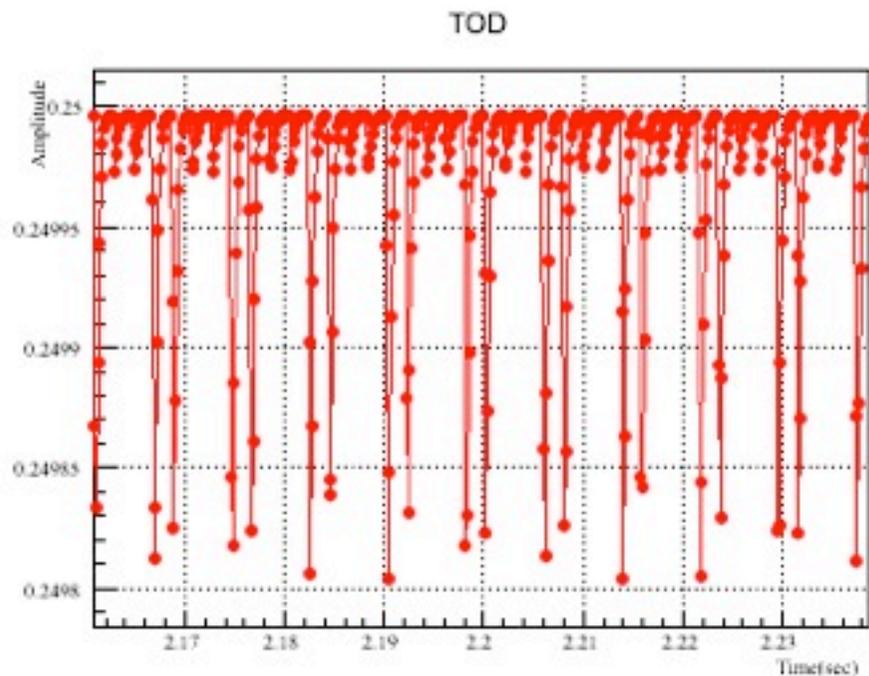


Tracking の時の読み出し速度では見えな
かったものが見えている

原因は??

→FPGA 内での信号処理のやり方が
原因らしいが.....

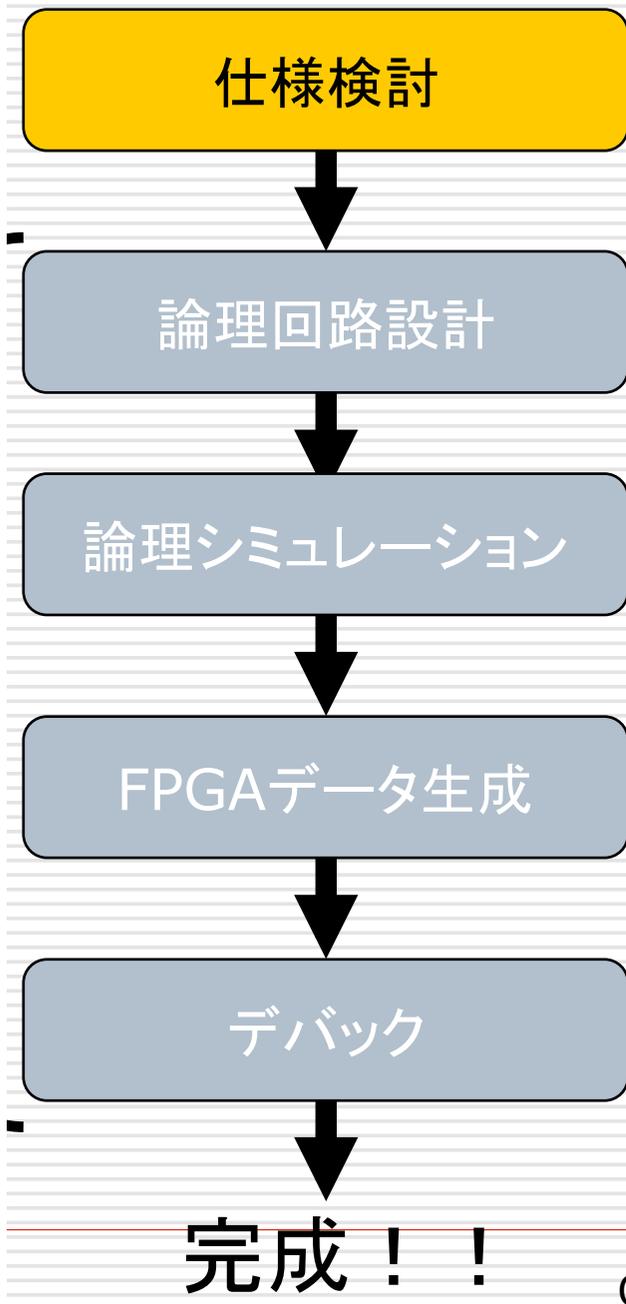
→その部分だけを修正しても上手く
いかない



何がまずかったのか

(from Open-It トレーニングコース)

色々（経験不足など）あるけど.....



要求仕様を確定する

- 処理性能
- 使い勝手
- 形状
- I/F ...

この部分がかなり曖昧だった
→読み出し速度はどれくらい？

→ADC/DAC Sampling rate はどの程度必要？

.....

論理シミュレーションも含めて、FPGA 内の
信号処理を数値計算でシミュレーションした
りする基本的な部分が抜け落ちてる

何がまずかったのか

Tracking には支障がなかったので放置していた

例えば Timing score が酷く悪い

→そもそも Timing constraint で failing が出ている.....

(9.54 nsec の要求が 64.482 nsec しか出てない)

The screenshot shows the ISE Project Navigator interface. The 'ScanTrackMKID Project Status' window is open, displaying the following information:

ScanTrackMKID Project Status (11/23/2012 - 20:58:41)			
Project File:	ScanTrackMKID_2ch_LargeFIFO_ver3.xise	Parser Errors:	No Errors
Module Name:	ScanTrackMKID	Implementation State:	Programming File Generated
Target Device:	xc4vsx35-10ff668	Errors:	No Errors
Product Version:	ISE 13.1	Warnings:	224 Warnings (15 new)
Design Goal:	Balanced	Routing Results:	All Signals Successfully Routed
Design Strategy:	Xilinx Default (unlocked)	Timing Constraints:	X 1 Failing Constraint
Environment:	System Settings	Final Timing Score:	64.482 (Timing Report)

The 'Device Utilization Summary' window is also open, showing the following data:

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	10,725	30,720	34%	
Number of 4 input LUTs	13,215	30,720	43%	
Number of occupied Slices	8,442	15,360	54%	
Number of Slices containing only related logic	8,442	8,442	100%	
Number of Slices containing unrelated logic	0	8,442	0%	
Total Number of 4 input LUTs	14,863	30,720	48%	
Number used as logic	12,903			
Number used as a route-thru	848			
Number used as 16x1 RAMs	16			
Number used for Dual Port RAMs	128			
Number used as Shift registers	168			

The 'Warnings' window at the bottom shows a warning message: "WARNING:Par:468 - Your design did not meet timing. The following are some suggestions to assist you to meet timing in your design."

Open-It

とにかく開発の経験値が乏しいので.....

→ Open-It に参加させていただくことになりました

μ-TPC readout system for SMILE	京都大学	産	京都大, IPNS	気球実験用システム、低消費電力	FPGA, PCB
IACT-PMT readout	CTA (京都大学、山形大学)	産	京都大、IPNS	高速エレクトロニクス、SiTCPの普及	PCB, FPGA, soft
PMT高圧制御	京都大学	産	京都大、IPNS	気球実験用システム	PCB, FPGA
Muon g-2/EDM SiTA 評価ボード	理研、KEK	上野	IPNS	高集積化	PCB
SVX4読み出しボード	大阪大学	花塚	IPNS	小型テストシステム	PCB, FPGA
XFEL用同期データ配信システム	Spring-8	阿部	IPNS	同期信号配信システム	PCB, FPGA
LAr用波形サンプリングADC	早稲田大学	寄田	IPNS	小型テストシステム	PCB, FPGA
SiTCPおよびDAQMW導入	IMSS(PF) 深さ方向分解 XAFS	雨宮	IMSS, IPNS	システム統合	PCB, FPGA, soft
SiTCP学習		内田	IPNS	SiTCPの使い方の学習	FPGA
γ-SiTCP	ICRR, 茨城大学、北里大学	榎本	IPNS	小型ガンマ線カメラ	FPGA
汎用MPPC読み出しモジュール	大阪大学、東北大学、KEK	吉村	IPNS	新しい共用資産	PCB, FPGA
CMB偏光観測のための読み出しシステム (ページ作成中)	岡山大学	岐部	IPNS	デジタル信号処理システム	FPGA

内田さん (E-sys) の suggestion を受けて、仕様検討から組み立て直し

読み出し系に要求される性能

1), ケーブル一本で多チャンネル読み出しが可能

→ MUX in Frequency Domain

メリット：極低温で動作させるので、一本の配線で複数チャンネル読めると、熱流入を防げる

2), Sampling Rate が検出器応答速度より十分速い

→ $\gg 1/\tau_{qp} \sim 10 \text{ kHz}$

3), 観測時間のロスがない **dead time less** での読み出し

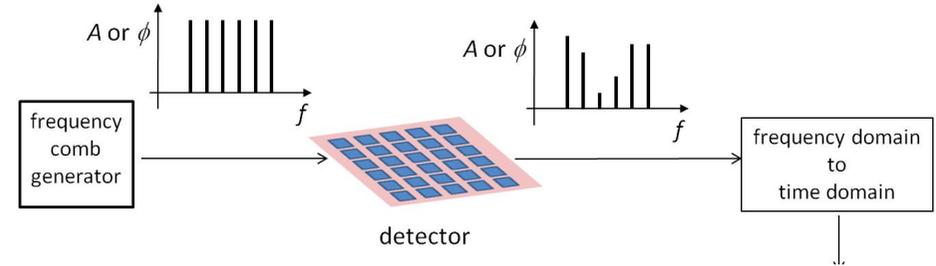
*ただし down-sampling はしてもいい

4), readout rate (offline に読み出す速度) $> 10 \text{ kHz}$

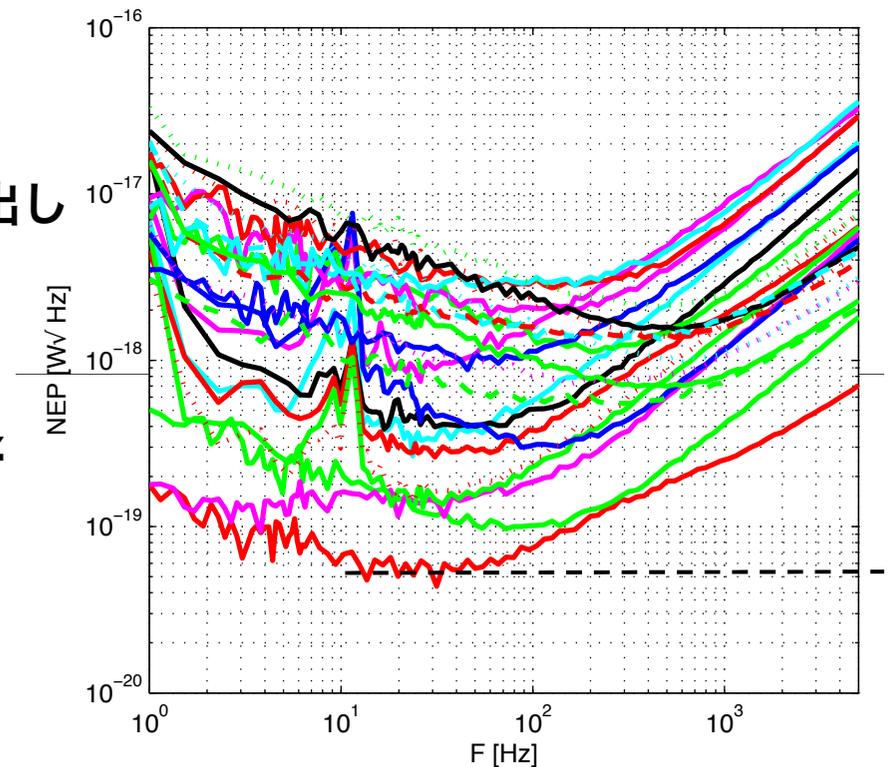
→ Common-mode Suppression でノイズを低減

→ typical f_{knee} of $1/f$ noise $\sim 1 \text{ kHz}??$

LiteBIRD 読み出し系の概念図



MKIDs のノイズスペクトル



Vertex-4 での達成目標

1), ADC/DAC 105 Msps (FPGA clock 105 MHz)

← 超伝導体の準粒子寿命($1/\tau_{qp} \sim 10\text{kHz}$)より十分短い間隔

2), readout rate: $> 10\text{kHz}$

← common mode (1/f noise knee $\sim 1\text{kHz}$??)より速いこと

3), ノイズcharacterizationができること

→ フーリエ変換できること

→ dead time lessで一定期間のdataを読めること

4), common mode noise suppressionの確認

→ # of MUX = 2ch

論理回路設計

FPGA 内での処理
を数式で表現

$$V_{DAC} = \begin{cases} \sum_i \cos \omega_i t \\ \sum_i \sin \omega_i t \end{cases}$$

$$V_{ADC} = \begin{cases} \sum_i \cos (\omega_i t + \phi_i) \\ \sum_i \sin (\omega_i t + \phi_i) \end{cases}$$

検出器が信号を受ける
→位相項が追加

受け取った ADC からある周波数 ω_k について I,Q を求める

$$\begin{aligned} I(t, \omega_k) &= \sum_i \cos (\omega_i t + \phi_i) \cos \omega_k t + \sum_i \sin (\omega_i t + \phi_i) \sin \omega_k t & Q(t, \omega_k) &= \sum_i \sin (\omega_i t + \phi_i) \cos \omega_k t - \sum_i \cos (\omega_i t + \phi_i) \sin \omega_k t \\ &= \sum_i \cos ((\omega_i - \omega_k) t + \phi_i) & &= \sum_i \sin ((\omega_i - \omega_k) t + \phi_i) \\ &= \cos \phi_k + \sum_{i \neq k} \cos ((\omega_i - \omega_k) t + \phi_i) & &= \sin \phi_k + \sum_{i \neq k} \sin ((\omega_i - \omega_k) t + \phi_i) \end{aligned}$$

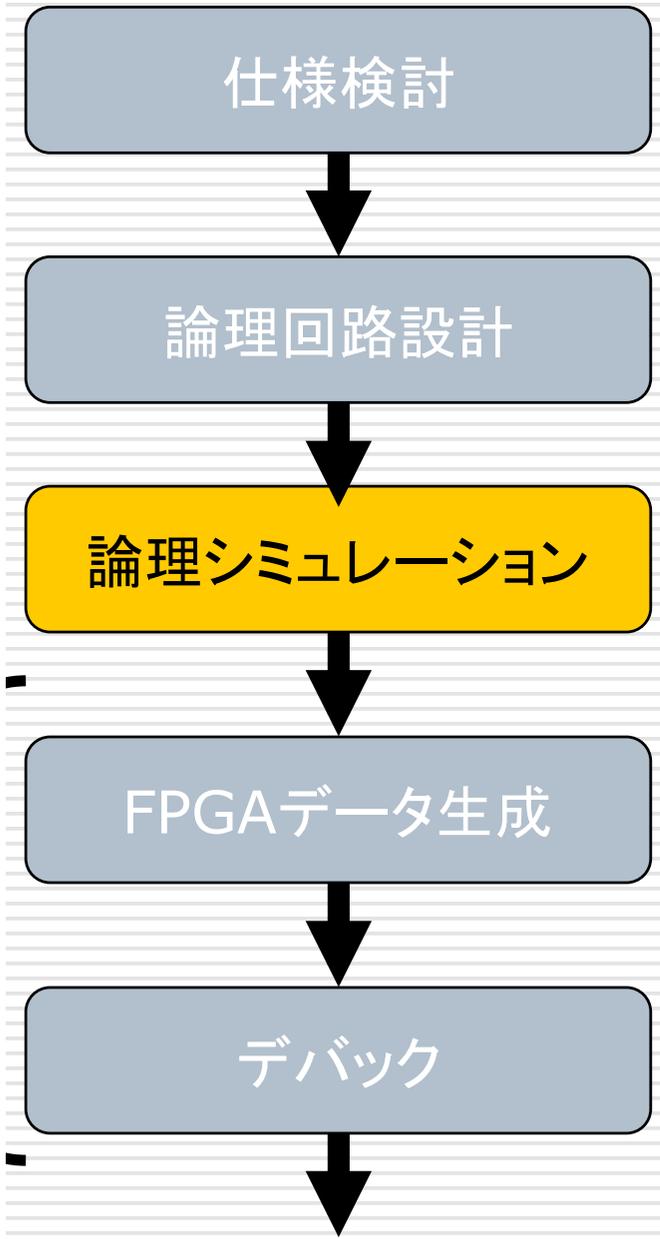
単純平均を取って down-sampling

$$\begin{cases} \frac{1}{T} \int_T I(t, \omega_k) dt \sim \frac{1}{T} \sum_{j=0}^N I(t_j, \omega_k) \Delta t = \cos \phi_k \\ \frac{1}{T} \int_T Q(t, \omega_k) dt \sim \frac{1}{T} \sum_{j=0}^N Q(t_j, \omega_k) \Delta t = \sin \phi_k \end{cases}$$

$$(T = N \Delta t)$$

デバッグ

(from Open-It トレーニングコース)



この部分はすっぽり抜け落ちてる
→予想通りの結果が得られるかどうかを

確認したい



例えば.....

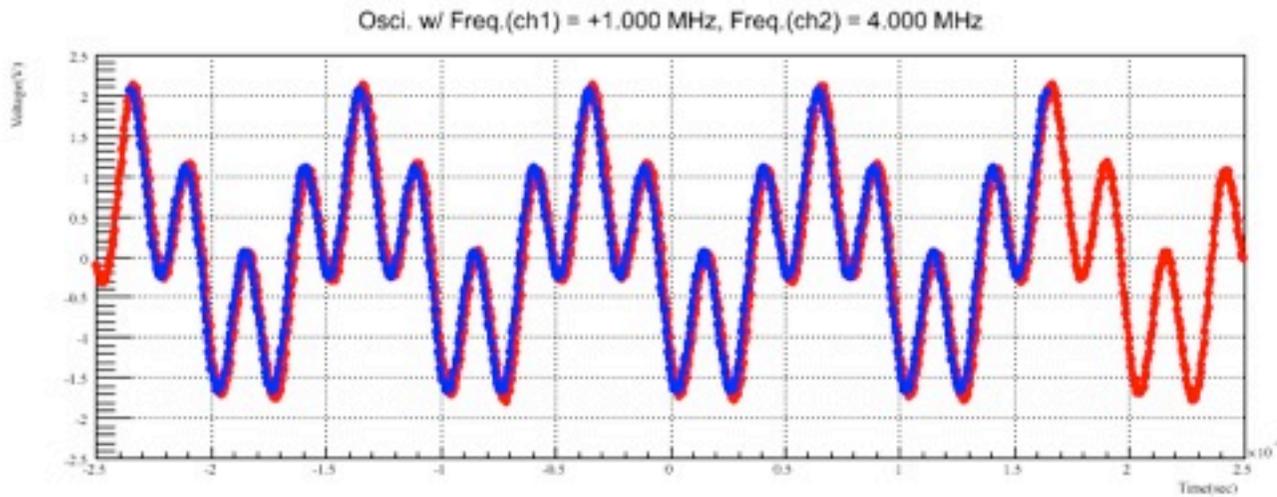
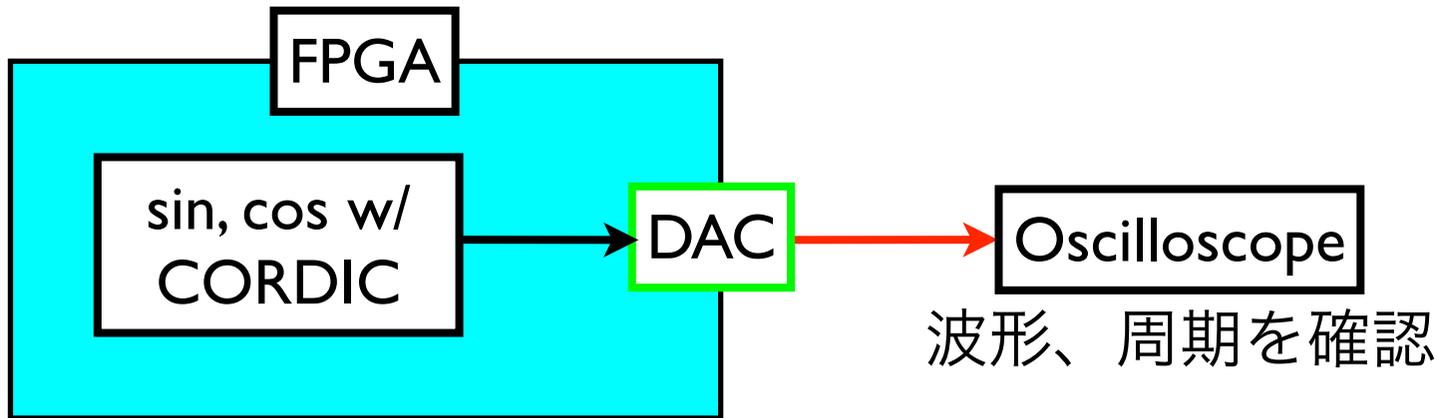
→ 10,000 samples の足し算はちゃんと動いているのか

(単純に足し算しても大丈夫?)

こういったものを積み重ねて、実際に FPGA にダウンロードして再度デバッグ

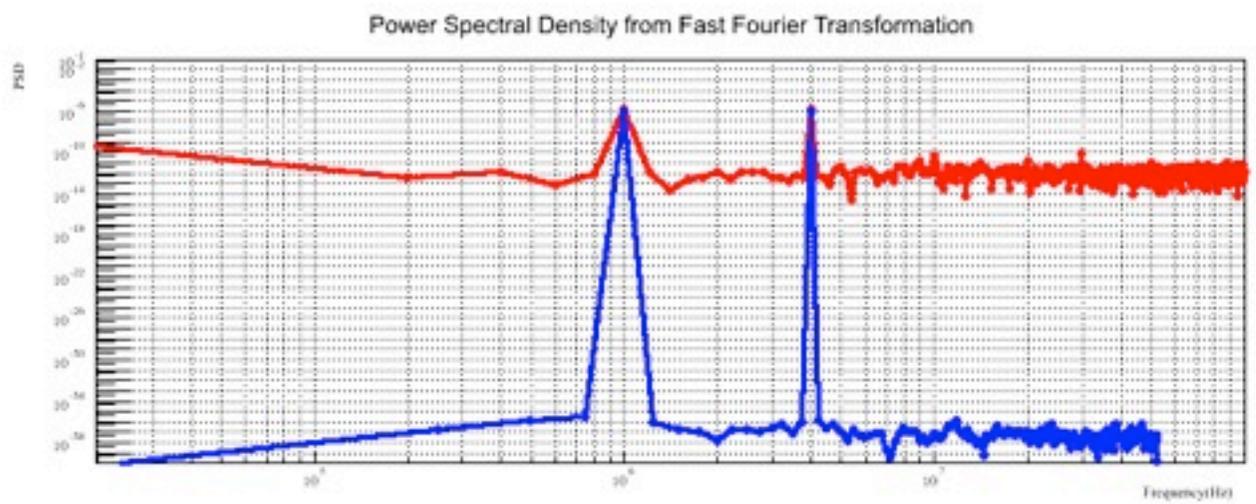
現在の開発状況

ADC/DAC のデバックまで終了



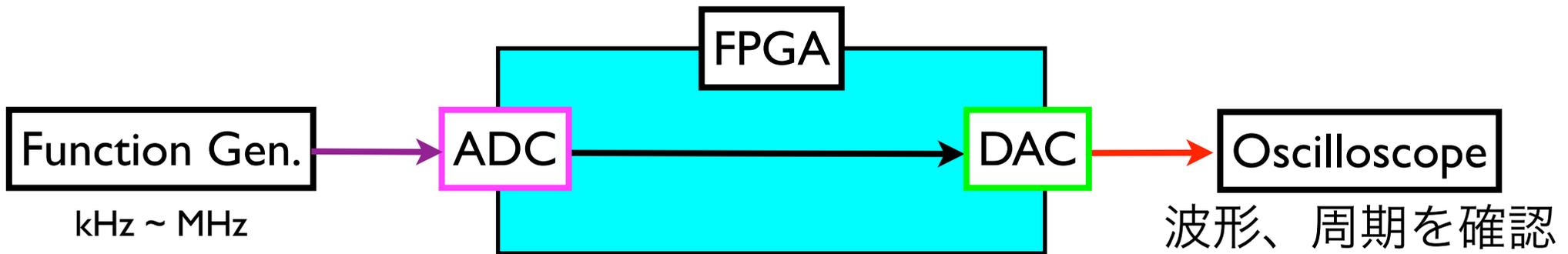
DAC output
MC

f1 = +1.0 MHz
f2 = +4.0 MHz

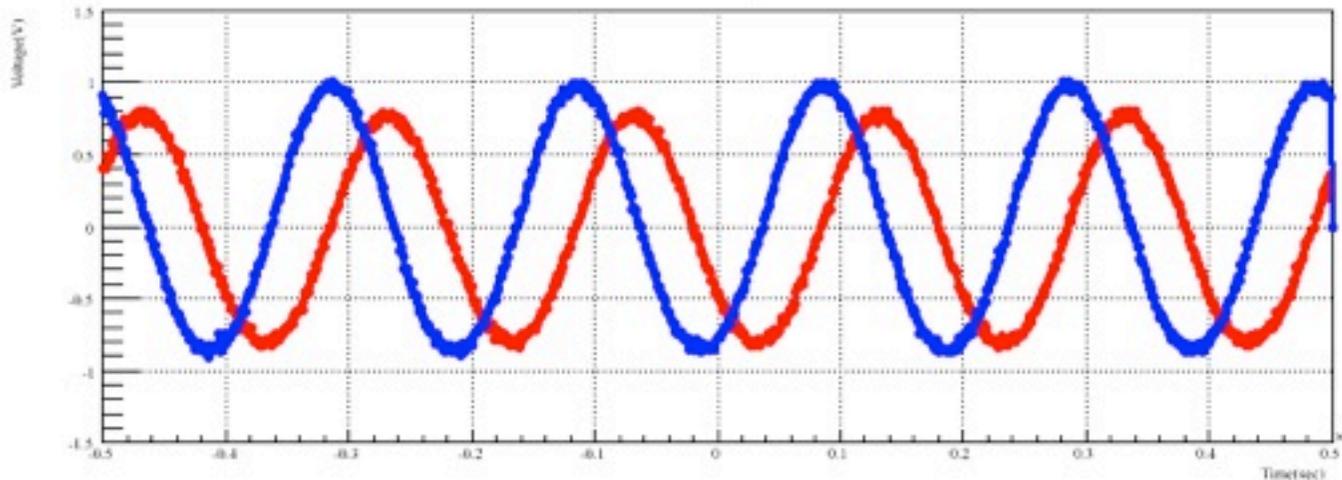


現在の開発状況

ADC/DAC のデバックまで終了



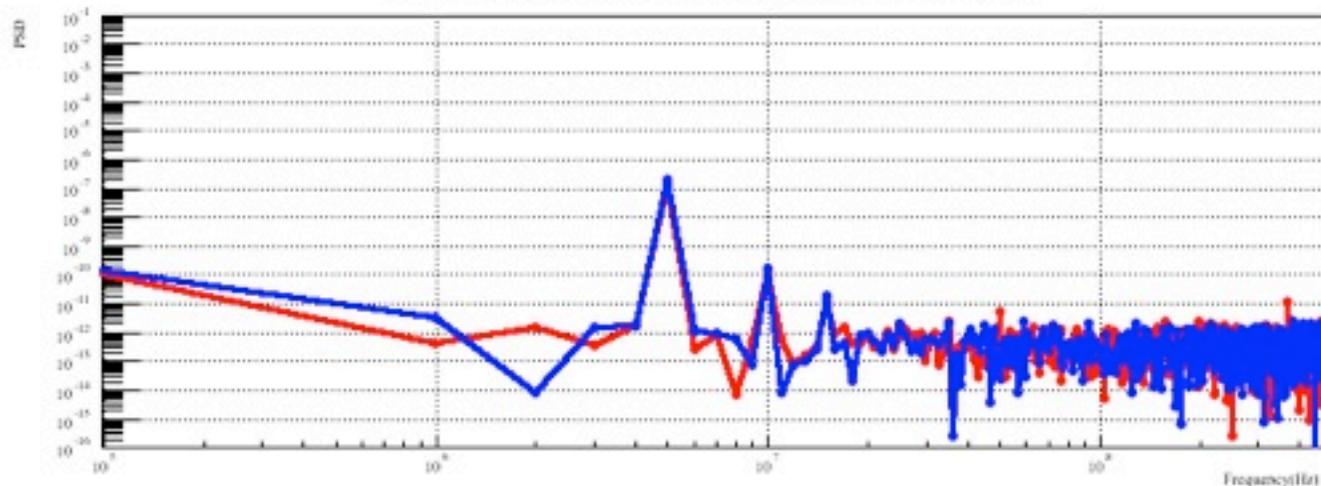
Fuction Generator: Freq. = +5.0000 MHz



DAC output
Function Gen.

$f = +5.0 \text{ MHz}$
(Function Gen. の読み)

Power Spectral Density from Fast Fourier Transformation



今後について

まずは本年度中に 2ch で正常に（仕様通りに）動くものを作る



100 ch 以上の同時読み出しを実装 + Tracking

ただ Vertex-4 では ~8 ch が限界

→ ボードをグレードアップ！

Vertex-4 → Kintex-7 へ

FPGA: Kintex-7 (XC7VLS325T
-2FFG900C)

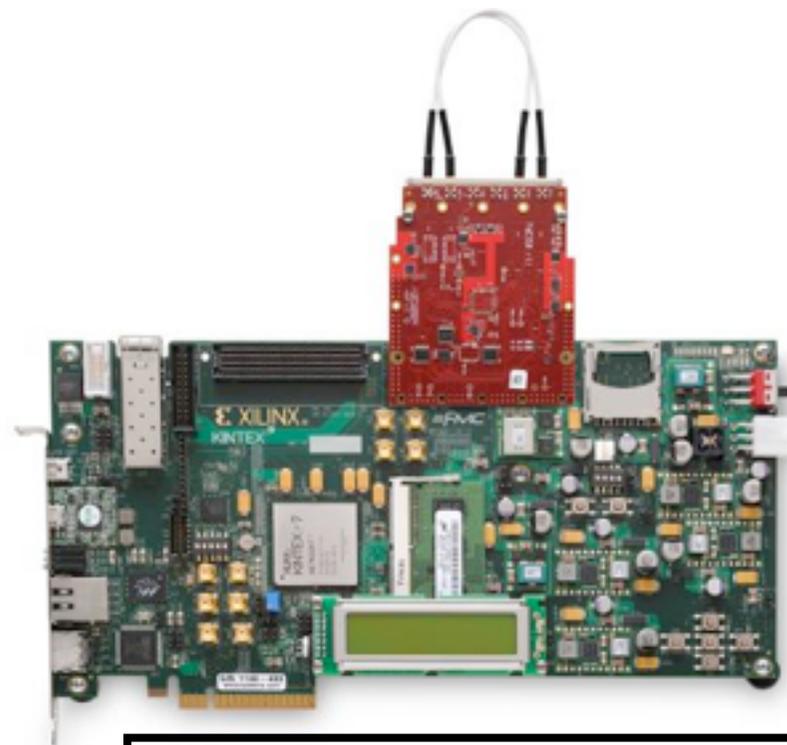
Clock: 200 MHz

DAC: 2ch, 16-bit, 800 Msps

ADC: 2ch, 14-bit, 250 Msps

Interface: PCI, UART-USB, Ethernet

LAN もあるので SiTCP の選択も



Xilinx Kintex-7 DSP 評価ボード

今後について

まずは本年度中に 2ch で正常に（仕様通りに）動くものを作る



100 ch 以上の同時読み出しを実装 + Tracking

単純に2ch → 100ch に拡張するだけでは難しい

例えば.....

- DAC からの出力は 100 ch 分の cos, sin 波になるが、100 ch 分の足し算で表現できるのか。もっといい方法はないのか
- 限りのあるリソースでなるべくコンパクトな設計にする必要がある。2ch × 50 ではリソースが足りない
- SiTCP にするのはいいが、今までやったことがない
(Vertex-4 には LAN がない)
- などなど.....

本年度（二月まで）の目標

2ch 読み出しで正しく動作するものを作る

→ 問題の精査、100 ch へのスムーズな移行

→ Common-mode suppression の実証

→ MKID がCMB観測に対して有用であることの証明

下記の仕様を完全に満たすものを製作

1), ADC/DAC 105 Msps (FPGA clock 105 MHz)

← 超伝導体の準粒子寿命($1/\tau_{qp} \sim 10\text{kHz}$)より十分短い間隔

2), readout rate: $> 10\text{kHz}$

← common mode (1/f noise knee $\sim 1\text{kHz}$??)より速いこと

3), ノイズcharacterizationができること

→ フーリエ変換できること

→ dead time lessで一定期間のdataを読めること

4), common mode noise suppressionの確認

→ # of MUX = 2ch

APPENDIX

論理回路設計

CLK = 105 MHz

数式→モジュール群に焼き直す (1chについて)

