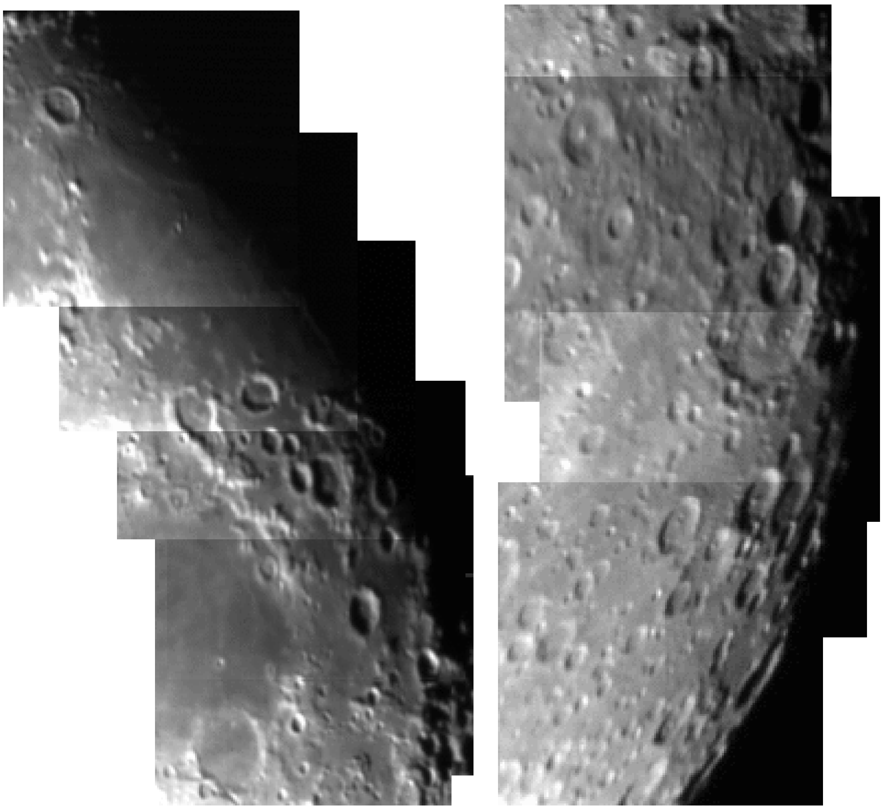
基盤Ｓカメラ検出器システムの説明書



2008年3月13日

内一・勝野　由夏

基盤 S カメラ（ANIR）は、TAO１m望遠鏡に搭載される予定の近赤外線カメラである。ANIR の検出器コントロールシステムについて、構成と製作状況を報告する。

1. 検出器

1.1. HAWAII-2 検出器

検出器としてはRockwell Scientific Company(RSC: 現　Teledyne)の HAWAII-2 アレイを使用する。HAWAII-2 は 2048×2048 の画素数を持つ HgCdTe 検出器である。HAWAII-2 は４つの独立した象限を持ち、読み出しは各象限で独立に行う。表 1 に HAWAII-2 の性能を示す。図 1 はダークカレントのヒストグラムであり、低いダークが実現していることを示す。

表 1　HAWAII-2 の性能（Rockwell ホームページより）

Parameter	Measured Performance	Units
Detector Interface Circuit	SFD	
Cell Pitch	18	µm
Die Size	1600	mm2
Integration Capacity	1.0x105	Carriers
Integration Capacitance	18 - 35	fF
Signal Conversion Gain	3.0 - 6.0	µV/e-
Output Signal Excursion	0.4 - 1.0	V
Maximum Data Rate	>1	MHz
Maximum Slew Rate	400	nsec
Minimum Read Noise (CDS)	<10	e-
Minimum Read Noise (Fowler Sampling)	<3	e-
Quantum Efficiency	>60	%
Spectral Response	0.85 - 2.5	µm
Dark Current @78K	<0.03	<10e-/sec
Power Dissipation	<2	mW

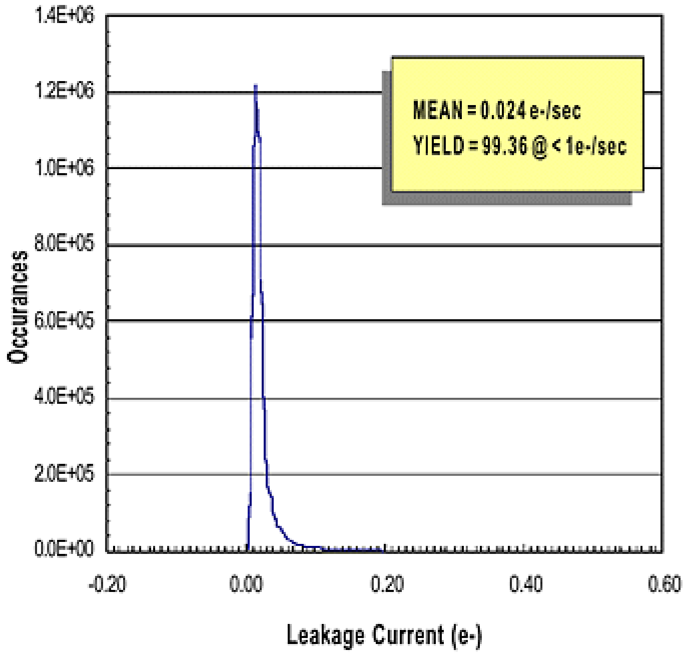


図 1　ダークカレントのヒストグラム（Rockwell ホームページより）

1.2. HAWAII-2 検出器の制御

Rockwell ホームページに掲載されていた HAWAII-2 に供給する電源と信号を、表 2、表 3、表 4、表 5、表 6 に示す。図 2 にパッドリスト、図 3 にクロックのタイミングを示す。

Power Supplies (7):

表 2　供給電圧（Rockwell ホームページより）

VDD	Digital high
VSS	Digital low
MUXSUB	Multiplexer substrate
VDDA	Analog high
DSUB	Detector substrate
DRAIN	Amp drain voltage, pulled up to VDDA internally
CELLDRAIN	Analog low in the unit cell

Control Signals (13):

表 3　クロック（Rockwell ホームページより）

CLK1	Clock for horizontal register (Pixel)

CLKB1	Clock for horizontal register (Pixel)
CLK2	Clock for horizontal register (Pixel)
CLKB2	Clock for horizontal register (Pixel)
VCLK	Master clock for the vertical register
LSYNC	External line sync
FSYNC	External frame sync
RESET	Control signal for reseting all the pixels
READ	Control signal for the readout
O1	Control signal for the output option
O2	Control signal for the output option
LRST	Reset for horizontal shift register
RESETEN	Optional control signal to reset the column bus to celldrain while no readout

Biases (3):

表 4　バイアス電圧（Rockwell ホームページより）

BIASPWR	Source voltage of bias P-fet
BIASGATE	Gate voltage of bias P-fet
VRESET	Zero signal reference in the cell

Outputs (9):

表 5　出力信号（Rockwell ホームページより）

OUTPUT[1:8]	The final outputs for 8 buses
OUTPUT9	Output of reference signal

Test Pads (3):

表 6　チェック端子（Rockwell ホームページより）

LINECHK	Horizontal testing signal output when TESTEN = high
FRAMECHK	Vertical testing signal having output when TESTEN = high

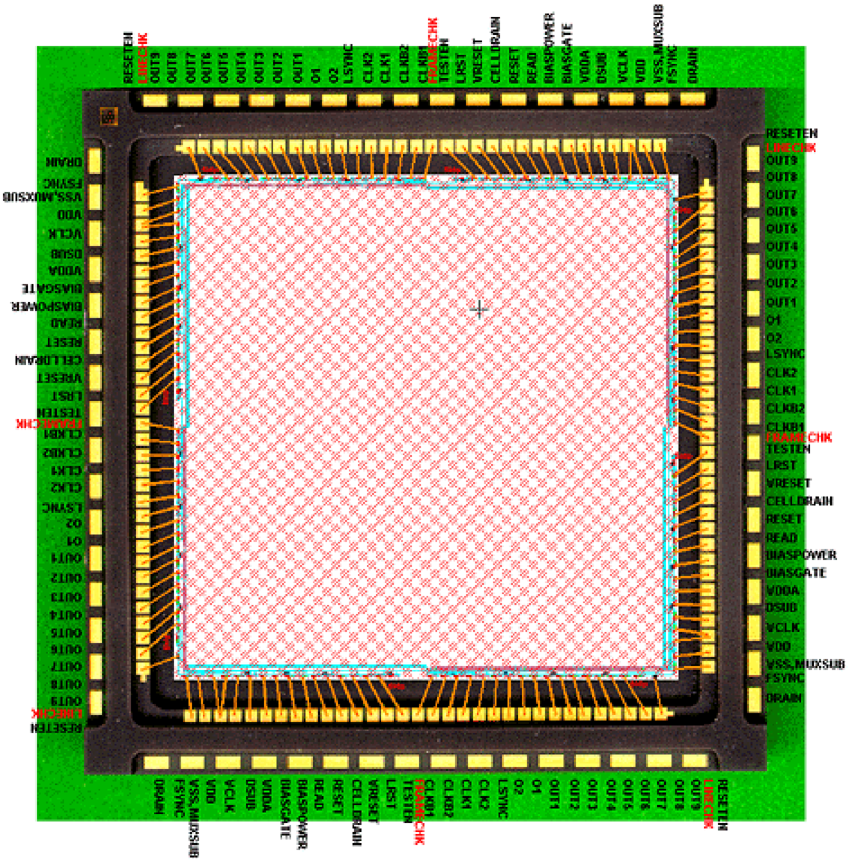


図 2　HAWAII-2 パッドリスト（Rockwell ホームページより）

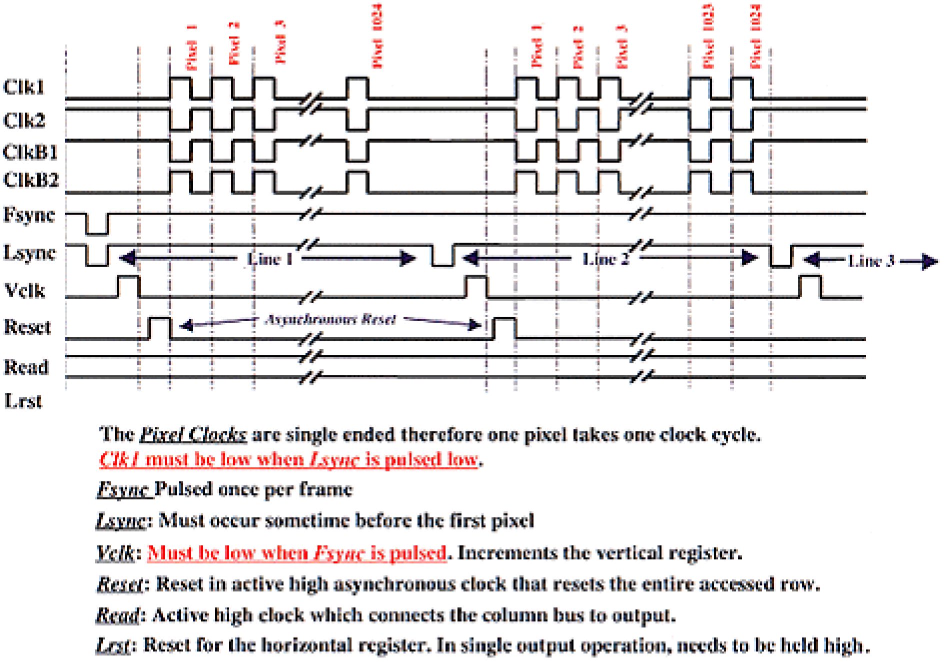


図 3　クロックのタイミング（Rockwell ホームページより）

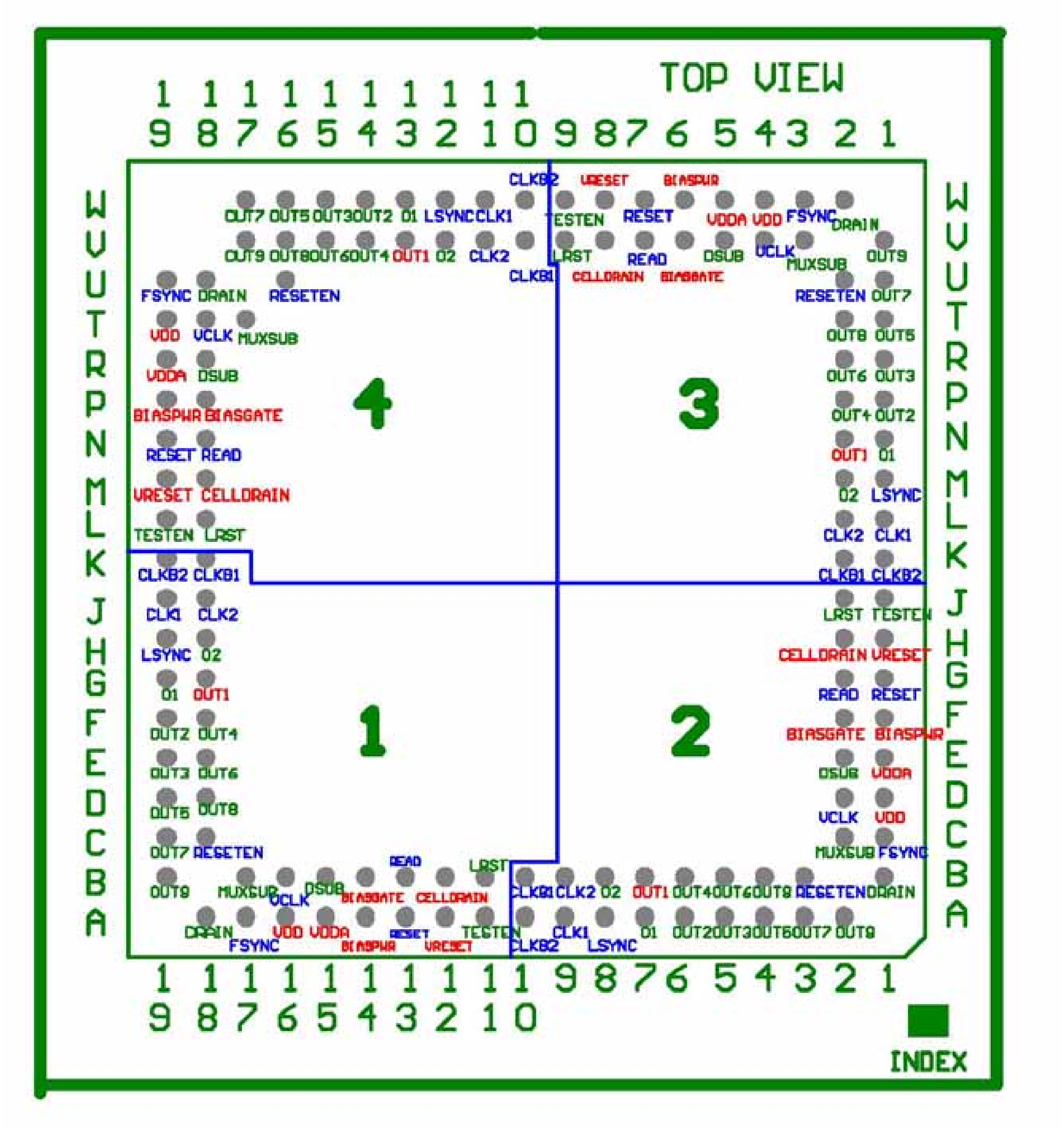


図 4　HAWAII-2 ピン配置　（市川隆氏作成）

検出器には、現在国立天文台ハワイ観測所にある HAWAII-2 エンジニアリングチップを借り、搭載することを予定している。

2. 検出器制御システムの設計と構成

2.1. フロントエンド回路

検出器は LINUX 計算機、PCI ボード、フロントエンド回路、ファンアウトボードによって制御、駆動を行う。

フロントエンド回路には酒向氏作成の中間赤外用フロントエンド回路を応用する。下図にシステム概要の流れ図を示す。ディジタル部にはインターフェイス社 PCI ボード PCI2772C （Interface 社 DMA 転送 20 MHz 駆動 32 点入出力）を用いる。

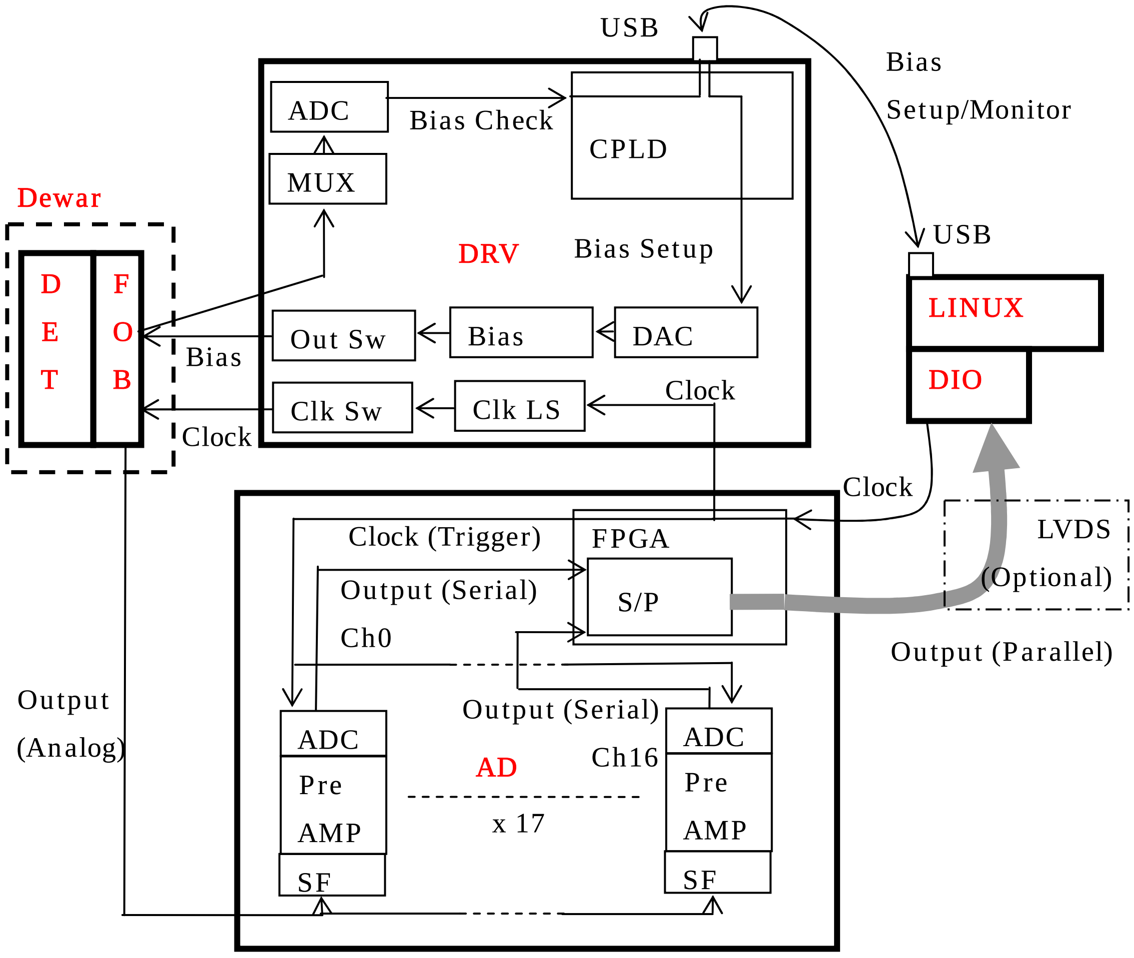


図 5 **酒向システムの概要**

フロントエンド回路は、DRV ボードと ADC ボードで構成される。DRV ボードは、バイアス生成とクロック生成を担う。DRV ボードにはバイアスが 12 ラインあり、ANIR ではそのうち 7 ラインを使う。中間赤外検出器は多様なバイアスが必要なので、オフセットを

Low, Mid, High の３種類設定(各 4 本)し、さらに DAC を通じて計算機からバイアス値の設定を行い、バイアス電圧を生成する設計になっている。HAWAII-2 のバイアスは 5 V が 6 ラインと 0.5 V が 1 ラインの計 7 本であるので、Mid と High は同じオフセットにする。バイアス電圧値は DRV ボード上の ADC を介して計算機から読むことができる。

ADC ボードは、プリアンプ、AD コンバータ （＋クロック受け取りと DRV ボードへの送信）を搭載している。17 チャンネルの出力を扱うことができるが、ANIR では 1 チャンネルのみを使用する。

2.2. ファンアウトボード

ファンアウトボードは ANIR の仕様に合わせて設計・製作した。デュワーが小さく、HAWAII-2 検出器を搭載する ZIF ソケットの大きさに対し、ボードの大きさが小さいため、HAWAII-2 の４象限のうち、１つの象限しか用いない。読み出し方法は 1 チャンネル出力/象限とする。1 ピクセル読み出すのに 5μs～10μs（200 kHz～100 kHz）かかるため、1k×1k の読み出しに 5~10 秒かかる。 1 フレームのデータサイズは 2 MB (1024×1024×16 bit) となる。ファンアウトボードは東北大 MOIRCS システムと本原氏の CISCO システムを参考にして製作した。

2.3. 検出器制御計算機

検出器制御には Linux 計算機を用いる。ADC ボード制御のためにはリアルタイム化が必要である。

(1) 実験用計算機

現在検出器テストのために使用している計算機の仕様は以下のとおりである。

Linux 計算機 Vine Linux 4.1
システムワークス社　Pentium 4 ~3 GHz, 1G メモリ
2bit PCI バスのスロット: 3 個
メモリ: 1GB
Parallel ATA

チリ・アタカマ山頂ではハードディスクが動かない可能性があるので、将来的に compact flash を使用してディスクレスシステムを構築する可能性があるため PATA にしたが、後に SSD 化(Solid State Drive)することになった。SSD は記憶媒体としてフラッシュメモリを用いるドライブ装置で、ディスク駆動部分が存在しない。SATA インターフェイスで動く。

実験室で使用されている実験用マシン名は ika.mtk.ioa.s.u-tokyo.ac.jp である。

2.4. 検出器カセットの構造

(1) 設計

検出器カセットは三谷夏子氏が設計した。入るものは、検出器、ZIF ソケット、ボード、ヒートシンク、ケーブル、である。

図 6 に HAWAII-2 パッケージの寸法､図 7､図 8 に検出器カセットの概念設計を示す。図 7 に入れてある寸法はおよその見積もりであり、正確でないことに注意されたい。

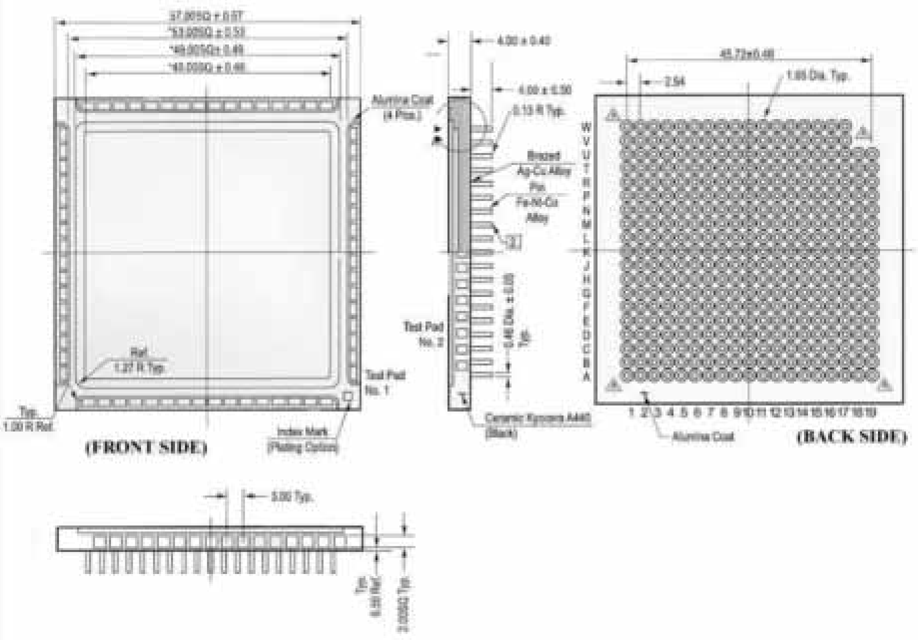
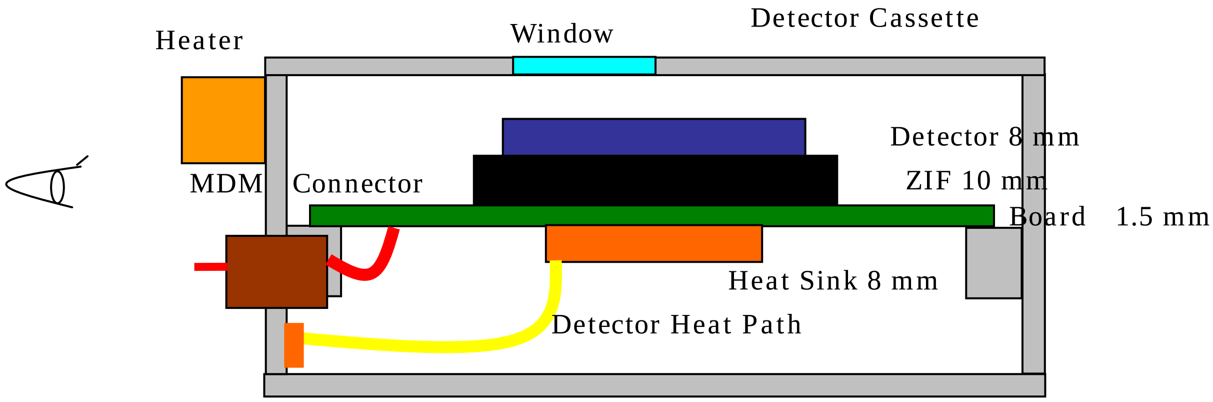


図 6　HAWAII-2 パッケージの詳細寸法（NIFS のページより。元は Rockwell）



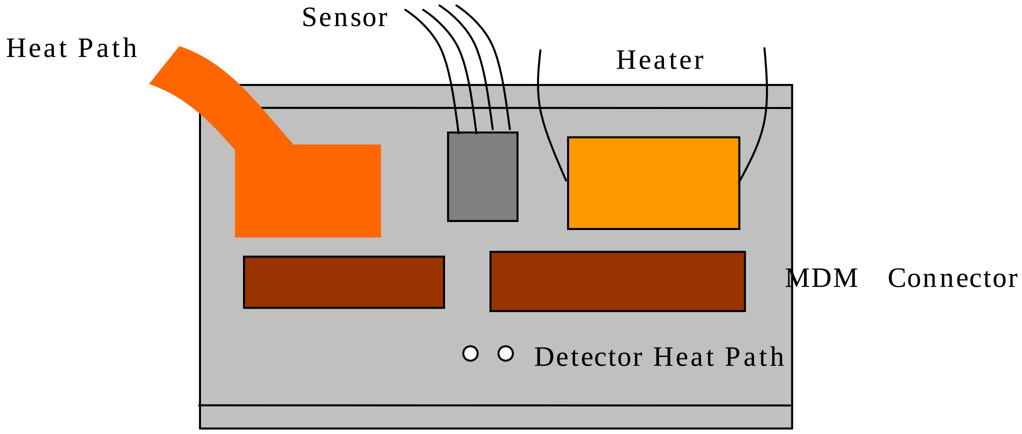


図 7　概念図：　検出器カセット断面図（上）と MDM コネクタ面（下）

図 8　概念図：　検出器カセット本体部　（ウインドウ方向から見た図）

2.5. 配線

ANIR の配線に関係する箇所の名前の定義と、デュワー内配線の大雑把なイメージを以下に示す。実際の部品の位置とは違うものもあるので注意すること。図中のマゼンタ色の数字は、配線の本数を示す。

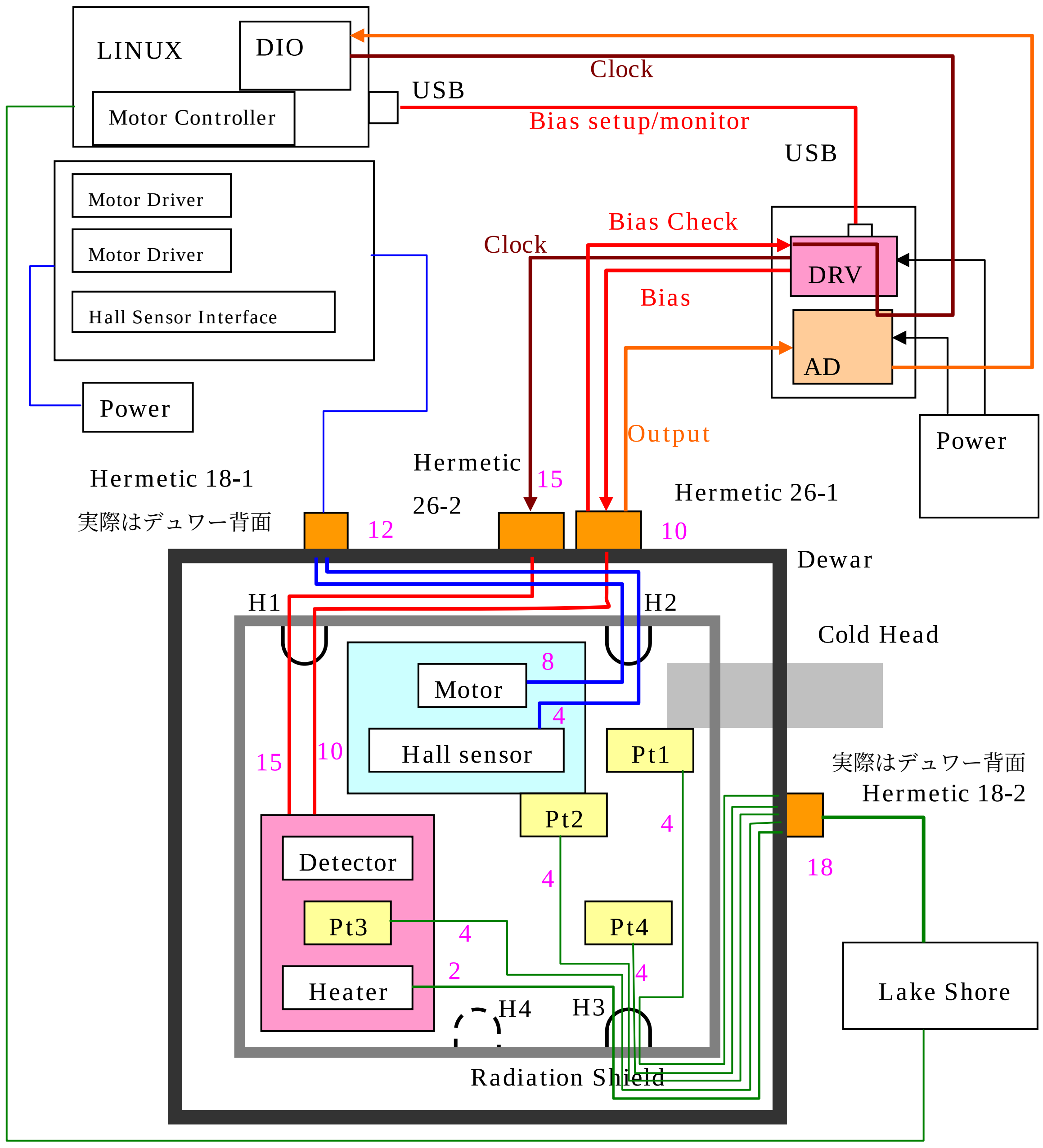


図 9　ANIR 配線図

ANIR 検出器配線ピン配置表

2007/11/08 Uchimoto
2008/2/16 Motohara

名前	種類	線色	電圧値[V]	FrontEnd	DIN96(DRV)	DIN96(ADC)	Hermetic16-23	Hermetic16-26	MDM25	MDM31
AGND	GND	黒	0	AGND	B?	–	A	–	1	–
DRAIN	BIAS	紫	5	D BIAS00	A4	–	B	–	2	–
VDDA	BIAS	赤	5	D BIAS01	C4	–	C	–	3	–
VDD	BIAS	赤	5	D BIAS02	A5	–	D	–	4	–
VRESET	BIAS	青	0.5	D BIAS05	C6	–	E	–	5	–
FETBIAS	BIAS	灰	5	D BIAS06	A7	–	F	–	6	–
FETGATE	BIAS	灰	5	D BIAS07	C7	–	G	–	7	–
VFET	BIAS	緑	5	D BIAS08	A8	–	H	–	8	–
OUTSIG1	OUT	黄	2.5	VOUT0	–	C17	R	–	14	–
OUTREF1	REF	橙	2.5	VGND0	–	B17	S	–	15	–
OUTSIG2	OUT	黄	2.5	VOUT1	–	C16	T	–	16	–
OUTREF2	REF	橙	2.5	VGND1	–	B16	U	–	17	–
OUTSIG3	OUT	黄	2.5	VOUT2	–	C15	V	–	18	–
OUTREF3	REF	橙	2.5	VGND2	–	B15	W	–	19	–
OUTSIG4	OUT	黄	2.5	VOUT3	–	C14	X	–	20	–
OUTREF4	REF	橙	2.5	VGND3	–	B14	Y	–	21	–
RESET	CLOCK	青	0/5	D CLK0	A31	–	–	A	–	1
READ	CLOCK	赤	0/5	D CLK1	C30	–	–	B	–	2
FSYNC	CLOCK	茶	0/5	D CLK2	A30	–	–	C	–	3
LSYNC	CLOCK	緑	0/5	D CLK3	C30	–	–	D	–	4
VCLK	CLOCK	黄	0/5	D CLK4	A29	–	–	E	–	5
CLK1	CLOCK	紫	0/5	D CLK5	C29	–	–	F	–	6
CLK2	CLOCK	紫	0/5	D CLK6	A28	–	–	G	–	7
CLKB1	CLOCK	灰	0/5	D CLK7	C28	–	–	H	–	8
CLKB2	CLOCK	灰	0/5	D CLK8	A27	–	–	J	–	9
LRST	CLOCK	橙	0/5	D CLK9	C27	–	–	K	–	10
RESETEN	CLOCK	青	0/5	D CLK10	A26	–	–	L	–	11
O1	CLOCK	赤	0/5	D CLK11	C26	–	–	M	–	12
DGND	GND	黒(太)	0	DGND	B?	–	–	N	–	20,21
PTDETV+	PT	橙	–	–	–	–	–	–	–	28
PTDETI+	PT	橙	–	–	–	–	–	–	–	29
PTDETI-	PT	青	–	–	–	–	–	–	–	30
PTDETV-	PT	青	–	–	–	–	–	–	–	31

右表へ

図 10　検出器配線ピン配置表

2.6. 周辺機器

電源　：　TEXIO（旧 KENWOOD)　PWR1.8Q。１台。日本電計より購入

ラック：　BRAIN 製。4 レーン。共立トレーディングより購入

2.7. 検出器制御ソフトウエア

酒向重行氏作成の中間赤外検出器制御用ソフトウエアを近赤外用として応用する。

(1) ハードウエア構成

ハードウエアは、Linux パソコンとインターフェイス社 PCI ボード PCI2772C（Interface 社 DMA 転送 20 MHz 駆動 32 点入出力）で構成される。Linux は RTAI 化する。RTAI はフリーのものを使う。インストール作業は ANIR ウェブの制御用 PC のメモに詳しい。現在の OS の構成は以下の通りである。

Vine Linux 4.1　kernel 2.6.17 + RTAI 3.5

DSP ボードではなく、リアルタイム OS を使用することが特徴と言える。

(2) 検出器制御ソフトウエア

現在インストールされている検出器制御ソフトウエアは以下の通りである。

TAO array controller (TAC) Ver.5.0.0 2008/2

酒向氏作成 developer's manual に、ソフトウエアの詳細、設定とインストール、オペレーションについて詳しく記載されているので、参照すること。

(3) データベース

検出器制御ソフトウエアは MySQL のデータベースを参照して動く。データベースの MySQL サーバは uni が担っており、実験室での構成では、ika が uni のデータベースを参照する。観測では uni がデータベースも検出器制御も行う。

3. 検出器制御システムの製作

3.1. ファンアウトボード

（1）仕様

HAWAII-2 の制御に必要なラインは以下の通りである。

バイアス：　DRAIN, VDDA, VDD, VRESET（４本）
VFET, FETBIAS, FETGATE（3 本）　（計７本）

クロック：RESET, READ, FSYNC, LSYNC, VCLK, CLK1, CLK2, CLKB1, CLKB2,
LRST, RESETEN, O1　（計 12 本、TESETEN と O2 は GND）

出力：　ANIR の場合は出力１本とリファレンス１本（計 2 本）

（2）設計

以上を踏まえてファンアウトボードの設計を行った。設計ソフトウエアには Eagle 4.1（CadSoft Computer 社、LINUX 版）を用いた。図 11 に ANIR 用ファンアウトボード設計図を示す。設計の際に検討した点を以下に示す。

- MOIRCS と CISCO のファンアウトボードの設計を参考にした。
- HAWAII-2 のオンチップアンプは使わない。オフチップアンプとして J270 を用いる。
- HAWAII-2 の 4 チャンネルのうち 1 チャンネルのみを使用する。使用しない象限のピンはグラウンドに落とす。この使い方は、西村さんから Rockwell(Teledyne)の技術者に聞いていただき、問題ないとのことである。
- １象限について１読み出しである。これは、8 読み出しも検討したが、デュワーの大きさから、ファンアウトボードの大きさに制限があり、最小限の部品しか載せられないことが分かったためである。

デュワーの設計から検討した結果、ファンアウトボードに許される大きさは 119 mm ×71 mm であった。

図 12 にボード設計図を示す。赤が Top 面、青が Bottom 面、マゼンタが PWR 面、シアンが GND 面を示す。部品配置の上で考慮した点は以下の通りである。

- グラウンドを広く取るようにした。また、出力のグラウンドがクロックのグラウンドと電気的に遠くなるようにした。
- Top 面にクロック、Bottom 面に出力、中間層（PWR 面）にバイアスを配置した。バイアスも広く取るように設計した。

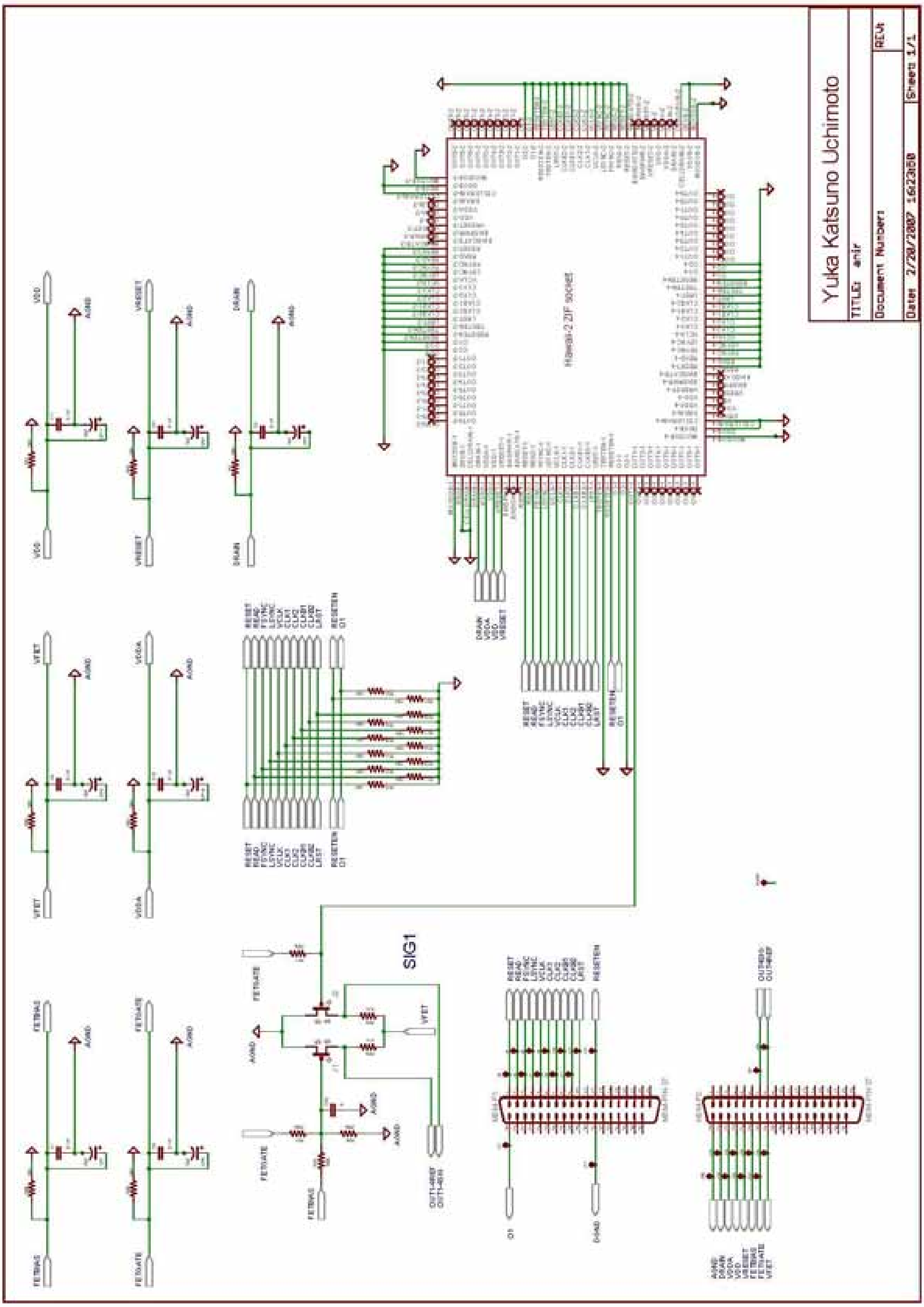


図 11　ANIR ファンアウトボード設計図：2007 年 2 月 22 日発注版

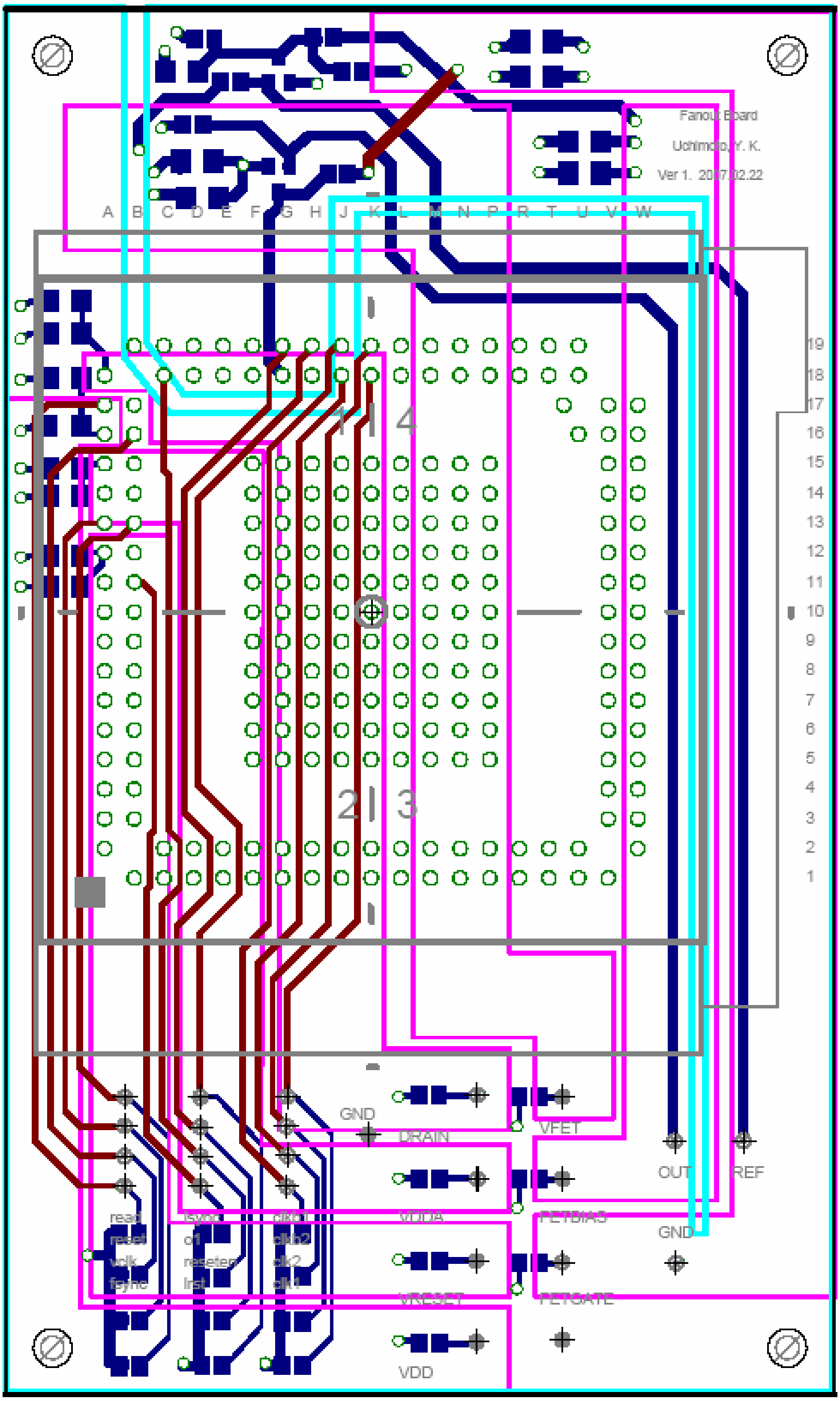


図 12　ANIR ファンアウトボードのボード設計図（119 × 71 mm）

（3）製作

設計したファイルをガーバーデータに変換し、P-ban.com へ発注を行った。図 13 に製作したファンアウトボードを示す。ZIF ソケットは HAWAII-2 制御に必要のない内側のピンを抜いた。中央に残したピンはヒートシンクに用いる。

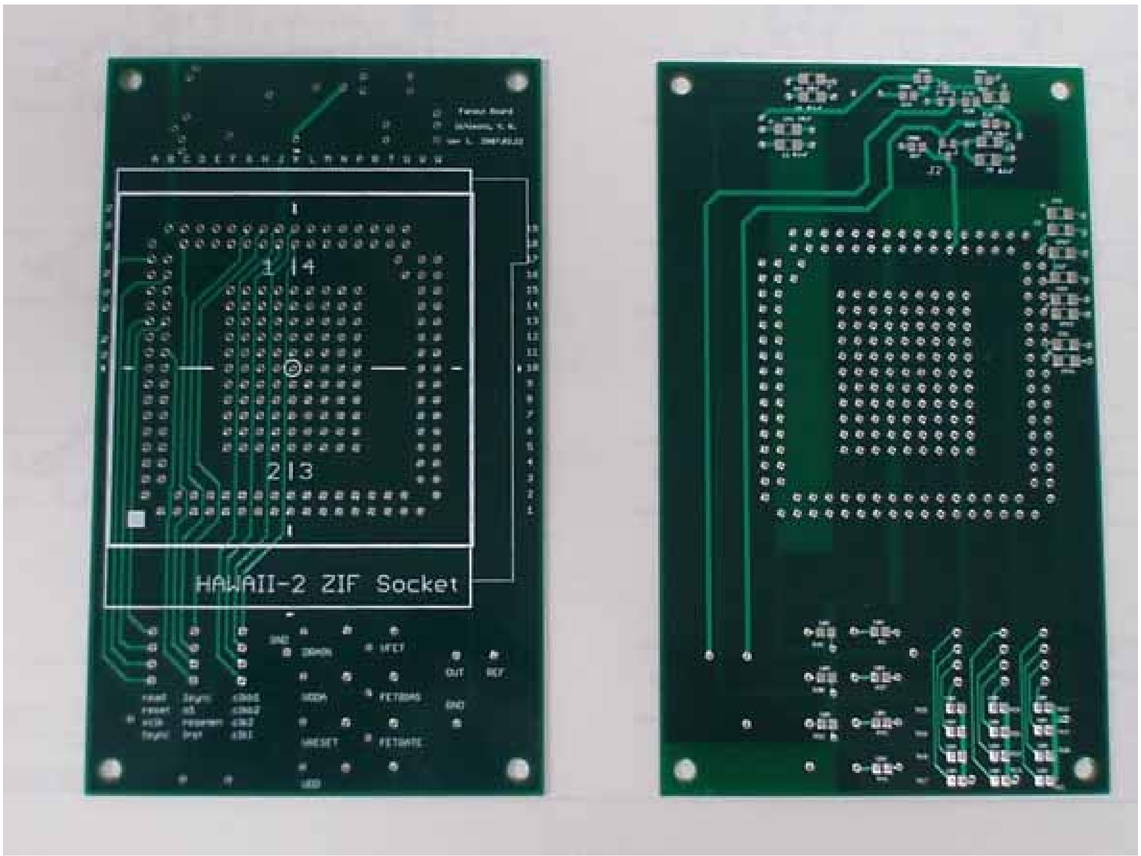


図 13　製作したファンアウトボード

(4) 実装

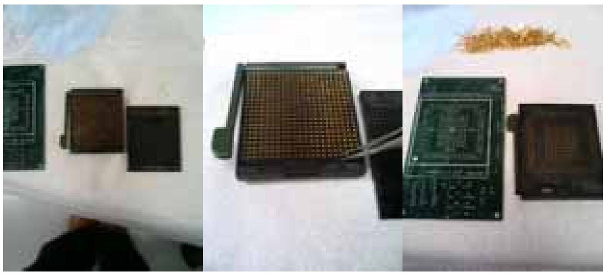


図 14　ファンアウトボードへの ZIF ソケットの実装

(4−1) ZIF ソケットの実装

ZIF ソケットを解体して、必要なピンだけ残し、いらないものをピンセットで抜いた。このとき、レバーは上げないように細心の注意をする。レバーが上がると、全ピンが浮いてしまって、元に戻りにくい。中央部分は読み出しピンではないが、ヒートシンクとして使用する。

ファンアウトボードの設計を間違えて、一つだけピンがずれていた。MUXSUB（読み出しをしない象限）なので、検出器内部で GND になっているはずであり、そのままにしても良い。ピンが柔らかいので、ずれたピンを曲げられるかと思ったが、曲げるとすぐ折れてしまった。ずれていたピンは外して、ZIF ソケットをファンアウトボードに差した。

(4−2) ヒートシンク取り付け・配線

ヒートシンク取り付けと配線は本原氏が行った。図 15　ヒートシンクと検出器ボックス図 15 にヒートシンクと配線・検出器ボックスを組み上げた様子を示す。

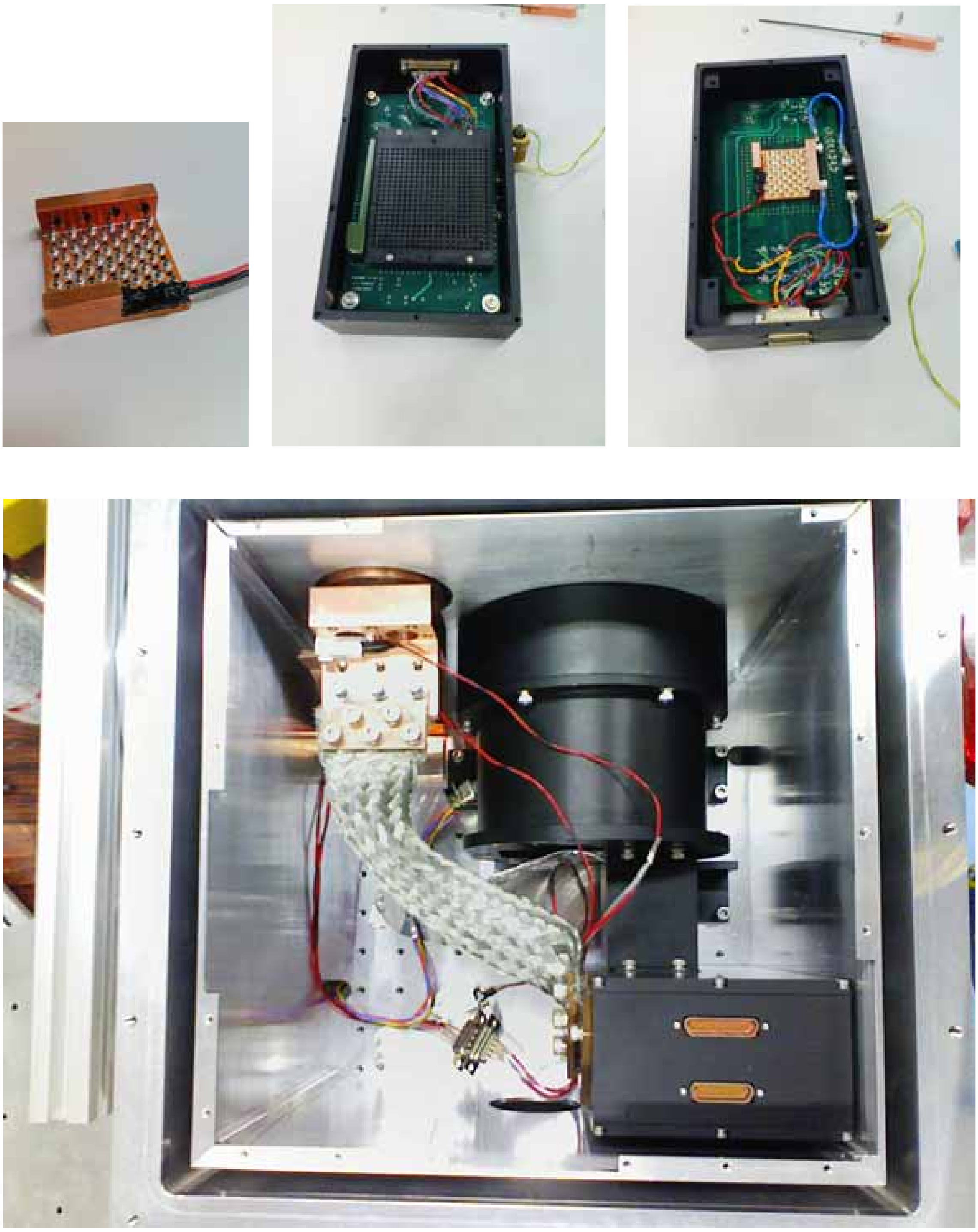
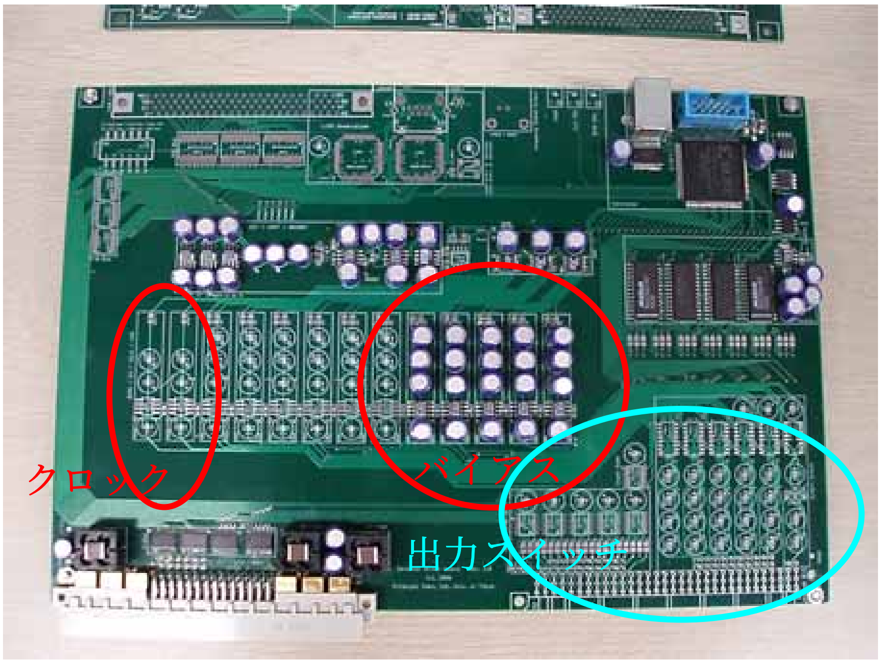


図 15　ヒートシンクと検出器ボックス

3.2. DRV ボード

（1）実装

信号の流れについては 2.1 章　図 5 を参考にされたい。使用するボードを**図** 16 に示す。HAWAII-2 に必要なバイアス値が出るように部品を選択し、半田づけによる実装を行った。



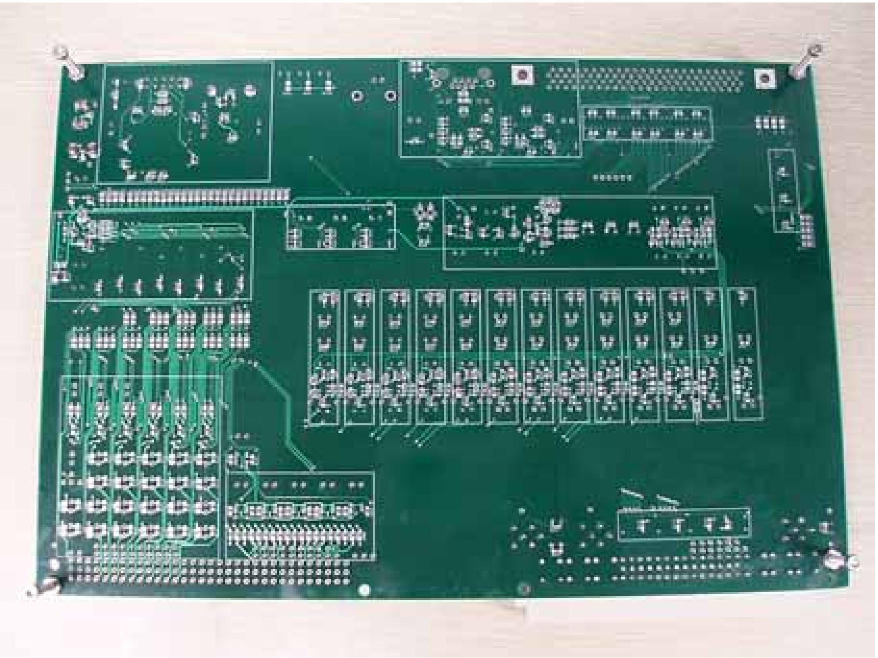


図 16　DRV ボード（上：表面、下：裏面）

[バイアスの種類と定義]
DRV ボード上のバイアスの種類と定義は以下の通りとする。

```
DRV Board    HAWAII-2
---------------------------
High Voltage
  DAC00:  DRAIN    (5V)
  DAC01:  VDDA     (5V)
  DAC02:  VDD      (5V)
Med Voltage
  DAC05:  VFET     (5V)
  DAC06:  FETBIAS  (5V)
  DAC07:  FETGATE  (5V)
Low Voltage
  DAC08:  VRESET   (0.5V)
```

バイアス供給部分については、HAWAII-2 に必要なバイアス電圧が出るように素子の定数を決め、半田づけを行った。**回路から設定したオフセット電圧は以下の通りである。**

<u>VOFFA = -2.47 V, VOFFB = -2.47 V, VOFFC = -0.112 V</u>

中間赤外検出器はバイアス電圧が仕様から外れると壊れる可能性があるため、アウトプットスイッチ（OSW）をもうけ、回路内で正しいバイアスが出ていることを確認してからバイアスを検出器に送る設計になっている。ANIR も同様に、まずバイアスを回路内で設定し、モニタしてから、OSW を ON にし、検出器にバイアスを送る。

（2）DRV ボードへのソフトウエアの書き込み

図 17　DRV ボードへのソフトウエア書き込み作業

ドライバボードの電源供給部分と、デジタル部分の半田づけを終えたところで、酒向氏に指示をもらいながら、2 つの IC （USB 用 FT232：バイアス設定、Xilinx 用 ds013：ディジタル制御）にコマンドを書き込む作業を行った。作業は Windows のソフトウエア Xilinx ISE7.1i を使った。書き込みの際には、1 つは USB 経由、もう一つは Xilinx 専用のケーブル(写真:カラーのもの)を用いた。書き込みをした後に、LINUX マシンに USB でボードをつないだところ、無事ボードを認識した。半田づけ、書き込みはうまくいったと思われる。図 17 に作業の様子を示す。

（3）DRV ボードの駆動

パソコンからバイアス電圧を設定し、想定とおりのバイアスが出ていることを確認した。バイアス制御ソフトウエアは、酒向氏製作の Max38 用ソフトウエアをコピーし、ANIR 用に変更した。

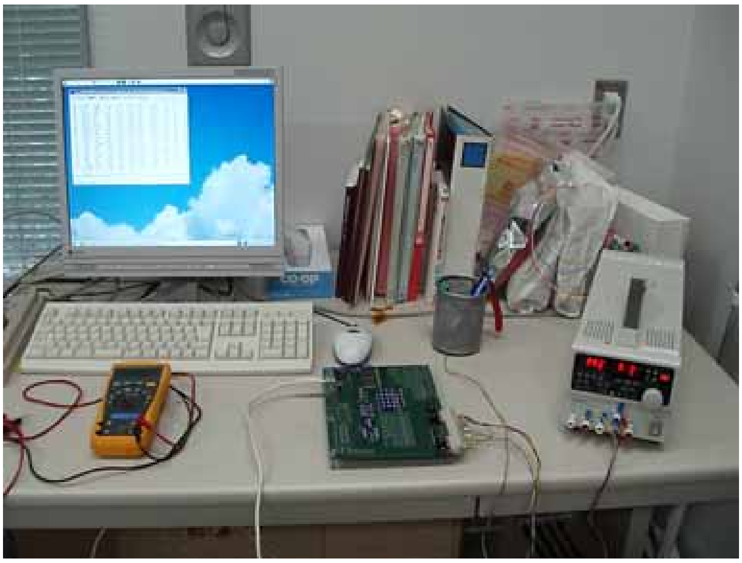


図 18　DRV ボード駆動の様子

テスト用バイアス設定　 2007.07.18　Uchimoto

(1) テスト用にプログラムを編集する

a)　/home/wasabi/abin/bis/src/bias.c

データベースを読みにいく設計になっているため、テストのために write_status というところはすべてコメントアウトする。

read_status というところはコメントアウトし、read_dac = 128 ;
と書き加える。
例)

```
/*
sprintf(key,"DRV_R_OHM");
sprintf(line,"%d", (int)DRV_R_OHM );
if(write_status(key,line)<0) error_at_rt(key) ; */

/* DAC に前回設定した DAC カウントをセットする */
for(i=0;i<TBL_NUM;i++){

   /* ステータスファイルから前回設定した DAC カウントを読み込む
   sprintf(key,"DRV_%s_DAC_SETTING",btbl[i].name);
   if(read_status(key,line)!=0) error_at_rt(key) ;
   sscanf(line,"%d",&read_dac);*/
   read_dac = 128 ;
```

b) /home/wasabi/abin/bis/src/bias.h

tbl_data[i].ddac (3000, 3050, 0 …) を変更することによって
バイアス (DAC の出力値) が変わる。
先ほどの bias.c の書き換えで 指定値=val=128 にしている。(read_dac)

DRV_SET_DAC_MAX/2 = 128
SET_DAC_GAIN = 5

より、{ (128-128) * 5 + 3000 } * 3.0 V / 4096 = DACOUT となる。

```
/****************************************************************************/
/*                              データテーブル                                */

/* DAC に設定されるカウント値は、((指定値=val) - DRV_SET_DAC_MAX/2)*SET_DAC_GAIN + tbl_data[i].ddac */
/* val はコンソールからの指定値。                                              */
/* val の指定範囲は SET_DAC_MIN < val < SET_DAC_MAX。                           */
/* tbl_data[i].ddac から大きく外れた電圧値を設定してしまわないように、             */
/* 調整範囲を(指定値=val)*SET_DAC_GAIN に狭めている。                            */
```

```
/*  内部バイアス電圧(=内部生成電圧) */
/*    TYPE, NAME, MUXCH, DACH, OSWCH, V, I, DDAC, OSWORD, ALIAS */
static struct struct_bias btbl_data[TBL_NUM] = {
   { "BIAS", "BIAS0" ,14, 0 , 0 , 7.7,       0, 3000,    2, "S_RON" },
   { "BIAS", "BIAS1" ,0 , 1 , 1 , 7.7,       0, 3050,    1, "VNWELL"},
   { "BIAS", "BIAS2" ,9 , 2 , 2 , 0.0,       0,       0,    0, "--"       },
   { "BIAS", "BIAS3" ,15, 3 , 3 , 0.0,       0,       0,    0, "--"       },
   { "BIAS", "BIAS4" ,1 , 4 , 4 , 5.5,       0,    350,    6, "AON"      },
   { "BIAS", "BIAS5" ,7 , 5 , 5 , 5.0,       0, 3490, 10, "VRST"    },
   { "BIAS", "BIAS6" ,6 , 6 , 6 , 3.0,       0,    700, 13, "VLG"      },
   { "BIAS", "BIAS7" ,8 , 7 , 7 , 4.0,       0, 2150, 11, "VDI"      },
   { "BIAS", "BIAS8" ,30, 8 , 8 , 2.0,       0,    720, 12, "VDSUB" },
   { "BIAS", "BIAS9" ,24, 9 , 9 , 2.0,       0,    700, 17, "VREF"    },
---
```

バイアス出力値は以下のようになる。

```
-----------------------------------------------------------------
ID        OFFSET tbl_data[i].ddac          DACOUT        -VOFF           BIAS
-----------------------------------------------------------------
BIAS0    VOFFA    3000 * 3.0 V / 4096 = 2.20 V         2.47 V          4.67 V
BIAS1    VOFFA    3050                               2.23 V        2.47 V          4.70 V
BIAS2    VOFFA          0                               0.00 V        2.47 V          2.47 V
BIAS3    VOFFA          0                               0.00 V        2.47 V          2.47 V
BIAS4    VOFFB      350                               0.26 V        2.47 V          2.73 V
BIAS5    VOFFB    3490                               2.56 V        2.47 V          5.03 V
BIAS6    VOFFB      700                               0.51 V        2.47 V          2.98 V
BIAS7    VOFFB    2150                               1.57 V        2.47 V          4.04 V
BIAS8    VOFFC      720                               0.53 V        0.11 V          0.64 V
BIAS9    VOFFC      700                               0.51 V        0.11 V          0.62 V
-----------------------------------------------------------------
```

(2) プログラムをコンパイルする。

```
[/home/wasabi/abin/bis]% su
パスワード(P):
[root@ika bis]# pwd
/home/wasabi/abin/bis
[root@ika bis]# make clean
rm ./src/*.o ;
[root@ika bis]# make
gcc -I../olib/src -I../olib/src -I./src    -Wall -O -c -o src/bias.o ./src/bias.c
src/bias.c: 関数 `main' 内:
src/bias.c:87: 警告: unused variable `key'
src/bias.c:87: 警告: unused variable `line'
gcc -I../olib/src -I../olib/src -I./src    -Wall -O -o bias ./src/bias.o ../olib/src/usblib.o -lm
```

(3) バイアスの設定

```
[/home/wasabi/abin/bis]% su
パスワード(P):
[root@ika bis]# pwd
/home/wasabi/abin/bis
[root@ika bis]# ./bias

 +---------   Driver board controller   --------+

     Usage: ./bias DAC default
            ./bias DAC ch(0..23) val(0-255)
            ./bias OSW on/off
            ./bias OSW ch(0..23) on/off
            ./bias MON all/ALL
            ./bias MON ch(0..23)


[root@ika bis]# ./bias DAC default
```

これで (1),(2) で設定したバイアス値になる。

(4) バイアスのモニタ

以下のコマンドにより、バイアス値をモニタすることができる。

```
[root@ika bis]# ./bias MON 0
   [root@ika bis]# ./bias MON all
```

ANIR 用バイアス設定値　2007.12.18　Uchimoto

1. バイアス設定値の変更

```
[root@ika anir]# cd bis/
[root@ika bis]# pwd
/home/wasabi/anir/bis
[root@ika bis]# cd src/
[root@ika src]# less bias.h
[root@ika src]# xemacs bias.h &
[1] 8997
******* bias.h を編集する。
[root@ika src]# cd ../
[root@ika bis]# ls
BiasMon*   Makefile   bias*   error070619.txt   src/
[root@ika bis]# pwd
/home/wasabi/anir/bis
[root@ika bis]# make clean
rm ./src/*.o ;
[root@ika bis]# make
gcc -I../olib/src -I../olib/src -I./src   -Wall -O -c -o src/bias.o ./src/bias.c
src/bias.c: 警告`main' ニ･
src/bias.c:87: 警告 unused variable `key'
src/bias.c:87: 警告 unused variable `line'
gcc -I../olib/src -I../olib/src -I./src   -Wall -O -o bias ./src/bias.o ../olib/src/usblib.o -lm
```

[root@ika bis]# pwd
/home/wasabi/anir/bis
[root@ika bis]#

2.バイアスの出力とモニタ

(1) バイアスをデフォルト値にする。

[root@ika bis]# ./bias DAC default

(2) バイアスをモニタする。

[root@ika bis]# ./bias MON all

ch	Name	Alias	Vset (V)	DAC	Vint (V)	OSW	Vout (V)	Iset (uA)	Idet (uA)
00	BIAS0	DRAIN	5.00	128 V	5.02 V	ON	5.02 V	0	-21
01	BIAS1	VDDA	5.00	128 V	5.02 V	OFF	0.17 F	0	0
02	BIAS2	VDD	5.00	128 V	5.04 V	OFF	0.17 F	0	0
05	BIAS5	VFET	5.00	128 V	5.03 V	OFF	0.15 F	0	0
06	BIAS6	FETBIAS	5.00	128 V	5.01 V	OFF	0.16 F	0	0
07	BIAS7	FETGATE	5.00	128 V	5.02 V	OFF	0.17 F	0	0
08	BIAS8	VRESET	0.50	128 V	0.51 V	OFF	0.17 F	0	0
09	BIAS9	VREF	2.50	128 V	2.50 V	OFF	0.17 F	0	0

(3) アウトプットスイッチを ON にする。

[root@ika bis]# ./bias OSW on
[root@ika bis]# ./bias MON all

ch	Name	Alias	Vset (V)	DAC	Vint (V)	OSW	Vout (V)	Iset (uA)	Idet (uA)
00	BIAS0	DRAIN	5.00	128 V	5.02 V	ON	5.02 V	0	0
01	BIAS1	VDDA	5.00	128 V	5.02 V	ON	5.01 V	0	21
02	BIAS2	VDD	5.00	128 V	5.04 V	ON	5.04 V	0	0
05	BIAS5	VFET	5.00	128 V	5.03 V	ON	5.03 V	0	-21
06	BIAS6	FETBIAS	5.00	128 V	5.01 V	ON	5.01 V	0	0
07	BIAS7	FETGATE	5.00	128 V	5.02 V	ON	5.02 V	0	0

```
08    BIAS8    VRESET   0.50    128 V   0.51 V   ON      0.51 V    0    21
09    BIAS9    VREF     2.50    128 V   2.50 V   ON      2.50 V    0   -21
```

(参照) 使わないチャンネルもモニタする。

```
[root@ika bis]# ./bias   MON ALL

ch    Name     Alias    Vset    DAC     Vint     OSW     Vout     Iset  Idet
                        (V)             (V)              (V)      (uA)  (uA)
--------------------------------------------------------------------------
00    BIAS0    DRAIN    5.00    128 V   5.02 V   ON      5.02 V    0   -21
01    BIAS1    VDDA     5.00    128 V   5.01 V   ON      5.01 V    0     0
02    BIAS2    VDD      5.00    128 V   5.04 V   ON      5.04 V    0     0
03    BIAS3    --       0.00    128 V   2.47 V   OFF     0.17 F    0     0
04    BIAS4    --       0.00    128 V   2.47 V   OFF     0.16 F    0     0
05    BIAS5    VFET     5.00    128 V   5.03 V   ON      5.03 V    0   -21
06    BIAS6    FETBIAS 5.00     128 V   5.01 V   ON      5.01 V    0     0
07    BIAS7    FETGATE 5.00     128 V   5.02 V   ON      5.02 V    0     0
08    BIAS8    VRESET   0.50    128 V   0.51 V   ON      0.51 V    0    21
09    BIAS9    VREF     2.50    128 V   2.50 V   ON      2.50 V    0   -21
10    BIAS10   --       0.00    128 V   2.73 V   OFF     0.11 F    0     0
11    BIAS11   --       0.00    128 V   0.80 V   OFF     0.11 F    0     0
12    CSA0     --       0.00    128 V   0.96 V   OFF     0.10 F    0     0
13    CSA1     --       0.00    128 V   1.14 V   OFF     0.10 F    0     0
14    CSA2     --       0.00    128 V   1.13 V   OFF     0.10 F    0     0
15    CSA3     --       0.00    128 V   1.11 V   OFF     0.10 F    0     0
16    CSB0     --       0.00    128 V   0.76 V   OFF     0.10 F    0     0
17    CSB1     --       0.00    128 V   0.99 V   OFF     0.10 F    0     0
18    CSB2     --       0.00    128 V   0.92 V   OFF     0.09 F    0     0
19    CSB3     --       0.00    128 V   0.94 V   OFF     0.10 F    0     0
20    VCLKNO   --       0.00    128 V   1.36 V   OFF     1.36 F
21    VCLKNC   --       0.00    128 V   2.59 V   OFF     2.59 F
22    VSSA     --       0.00    128 V   2.53 V   OFF     0.09 F    0     0
23    VSSB     --       0.00    128 V   0.99 V   OFF     0.11 F    0     0

[root@ika bis]#
```

3.3. ADC ボード

流れ図は図 5 を参考にされたい。ANIR の仕様に従って実装を行う。図 19 に使用するボードを示す。

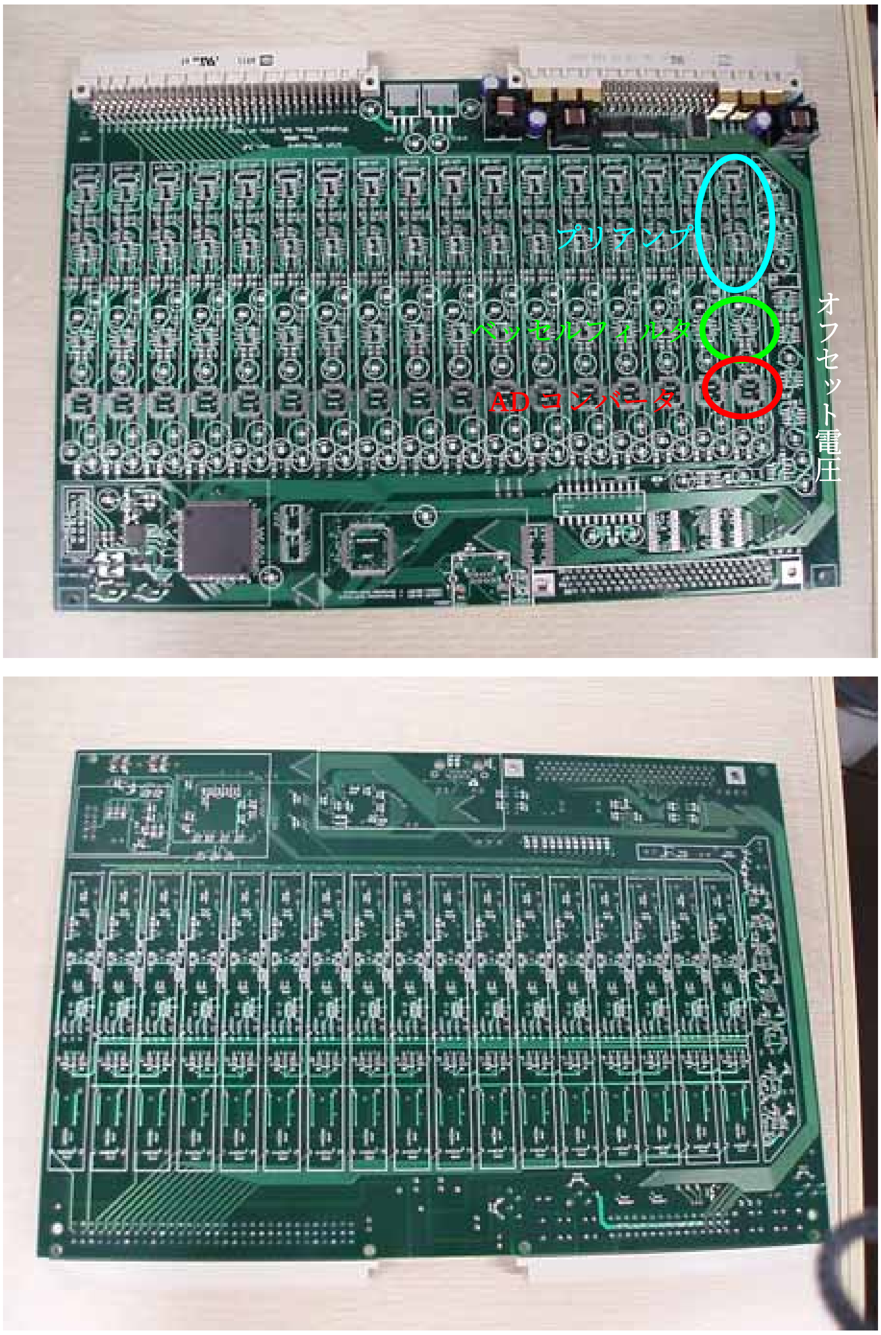


図 19　AD ボード（上：表面、下：裏面）

(1）回路図の説明

ADC ボード・プリアンプ部の解説と実際に実装した部品定数を示す。

a. バッファアンプ（レベルシフト）部の入力と出力の関係

$$A' = \left(1 + \frac{R2}{R1}\right) Vout - \frac{R2}{R1} \cdot Voff$$

$$B' = \left(1 + \frac{R2}{R1}\right) Vref$$

b. 差動アンプの入力と出力の関係

$$A'' = \frac{1}{2}\left(2OCM - (A' - B')\frac{R_B}{R_A}\right)$$

$$B'' = \frac{1}{2}\left(2OCM + (A' - B')\frac{R_B}{R_A}\right)$$

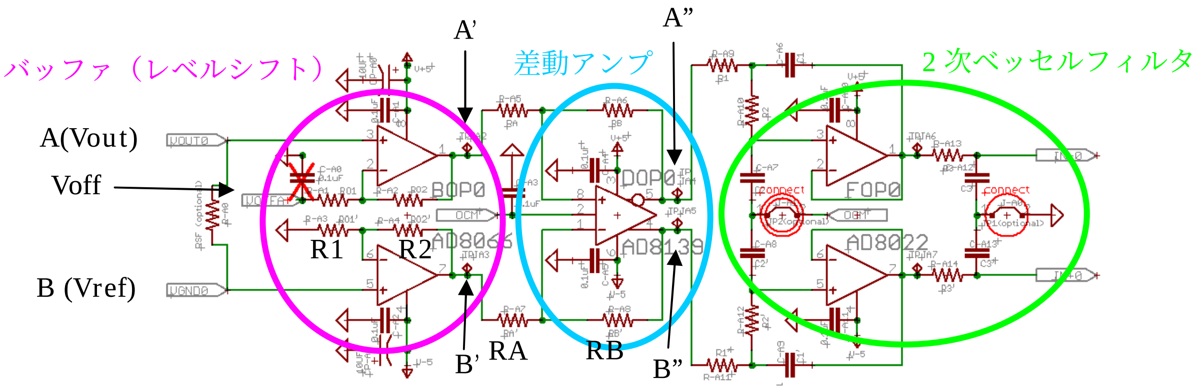


図 20　プリアンプ部の回路図

b に a を代入する。

$$A'' = -\frac{1}{2}\frac{R_B}{R_A}\left\{\left(1 + \frac{R_2}{R_1}\right)(V_{out} - V_{ref}) - \frac{R_2}{R_1}V_{off}\right\} + OCM$$

$$B'' = +\frac{1}{2}\frac{R_B}{R_A}\left\{\left(1 + \frac{R_2}{R_1}\right)(V_{out} - V_{ref}) - \frac{R_2}{R_1}V_{off}\right\} + OCM$$

$$B'' - A'' = V_{ADC} = \frac{R_B}{R_A}\left\{\left(1 + \frac{R_2}{R_1}\right)(V_{out} - V_{ref}) - \frac{R_2}{R_1}V_{off}\right\} \qquad (1)$$

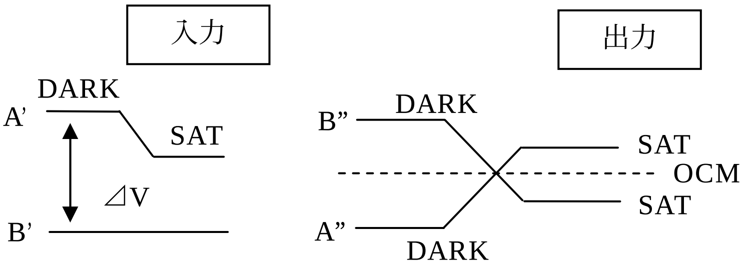
[差動アンプについて]

第 2 段の差動アンプの性質を以下に示す。

$$\left(A''+B''\right)/2 = OCM$$

$$B''-A''=\left(A'-B'\right)\frac{R_B}{R_A}$$

差動アンプ



ADC の制約として、A”を IN-、B”を IN+とすると以下のようになる。

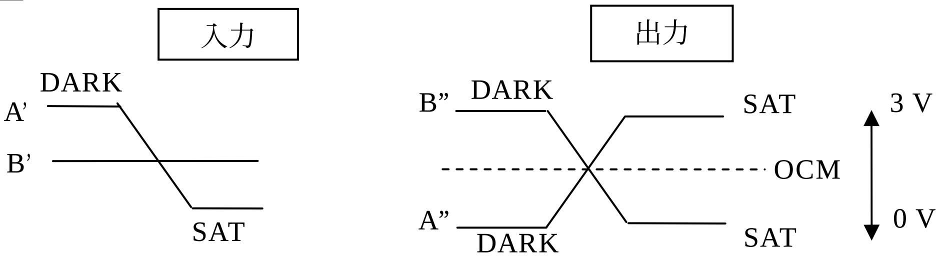
$$-3 < \left(IN_{+} - IN_{-}\right) < +3$$

$$0 < IN_{+} < 3$$

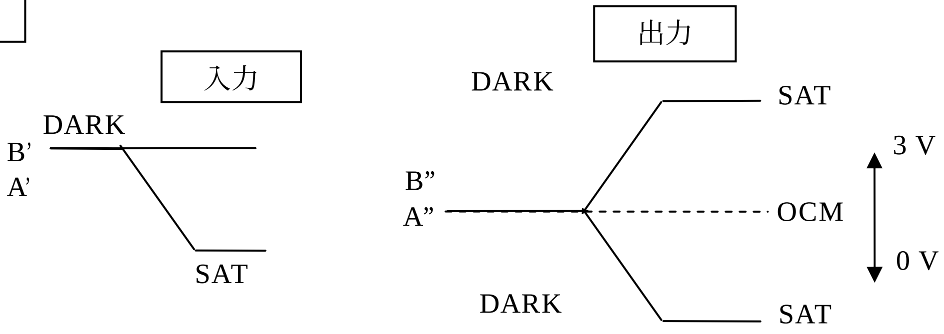
$$0 < IN_{-} < 3$$

また、ADC のレンジを最大に取るには、必ず OCM = 1/2×3 V = 1.5 V になる。

良い例



悪い例



最大±3V の範囲で出力電圧を設定できる。OCM は必ず 1.5 V である。余裕を見て、±2.1 V (70%) を使用する。HAWAII-2 アレイの場合、Vout = 2.5 V @dark, Vout = 2.0 V @ full, Vref = 2.5 V とする。出力の最小は Vout – Vref = －0.5 V のときであり、最大は Vout －Vref = 0.0 V のときである。式(1) にこれを代入する。

$$\frac{R_B}{R_A} = \frac{1}{1+\frac{R_2}{R_1}} \times 8.4$$

ゲインは 8.4 倍を取れる。

$$V_{off} = -0.25 \cdot \frac{R_2}{R_1} \cdot \left(1+\frac{R_2}{R_1}\right)$$

IC の性質から、B' < 4V の範囲で使用しなければならない。Vref = 2.5 V なので、1＋R2/R1 < 1.6 でなければならない。各 IC の入出力電圧範囲を考えて、1＋R2/R1 = 1.3 と設定する。これより、Voff＝－1.083 V であれば良い。この設定により増幅率は $R_B/R_A = 6.46$ となる。

上記の通りに部品定数を取れば、各部電圧は以下の通りが想定される。

リファレンス電圧　B' = 3.25 V

ダークのとき　A' = 3.575 V　　　飽和のとき　A' = 2.925 V

ダークのとき　B" = 2.55 V　　　飽和のとき　B" = 0.45 V

2 次ベッセルフィルタは f0 = 500 kHz に設定する。

システムゲインは、検出器のウェル 10^5 e- = 0.5 V, ADC のサンプリングが 16 ビット, ADC 出力電圧範囲を 6 V とすると、 2.18 [e- / ADU] となる。

(2) 動作実験　（2007 年 9 月 5 日）

ADC ボードに半田付けと配線を行い、パルスジェネレータで入力信号を入れ、出力をオシロスコープで見た。

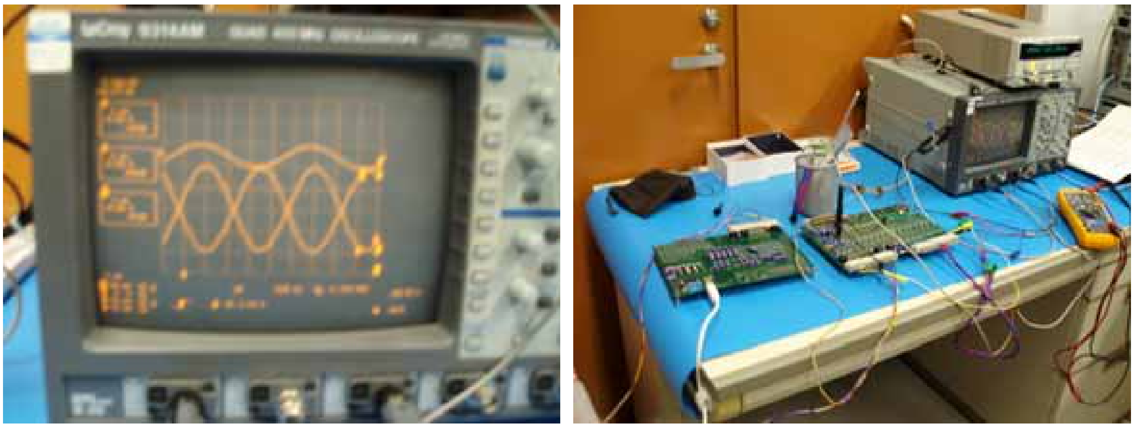


図 21　ADC 動作実験の様子

[入力]
Vout : 2.0 ~ 2.5 V の sin 波（200 kHz）
Vref : 2.5 V（バイアスボードで生成）
Voff : －1.3 V

[出力結果]
A' : 2.93 ~ 3.58 V の sin 波
B' : 3.25 V

A” : 0.45 ~ 2.5 V の sin 波
B” : A” の位相が反転した波形

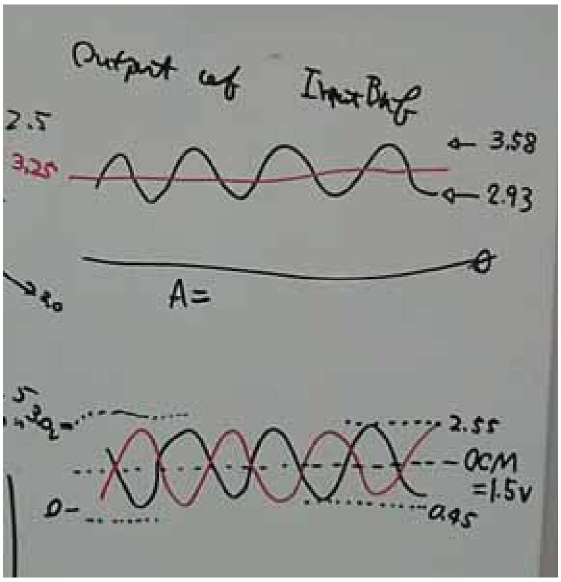


図 22　波形の模式図　（酒向氏画）

3.4. バックプレーン

本システムはバックプレーンにも配線を行う。フロント部はパソコンとの接続であるのに対し、バックプレーンは電源、ADC-DRV ボード間のデジタルクロックの接続、出力信号線が配線される。回路図に従い、これらの配線を製作した。

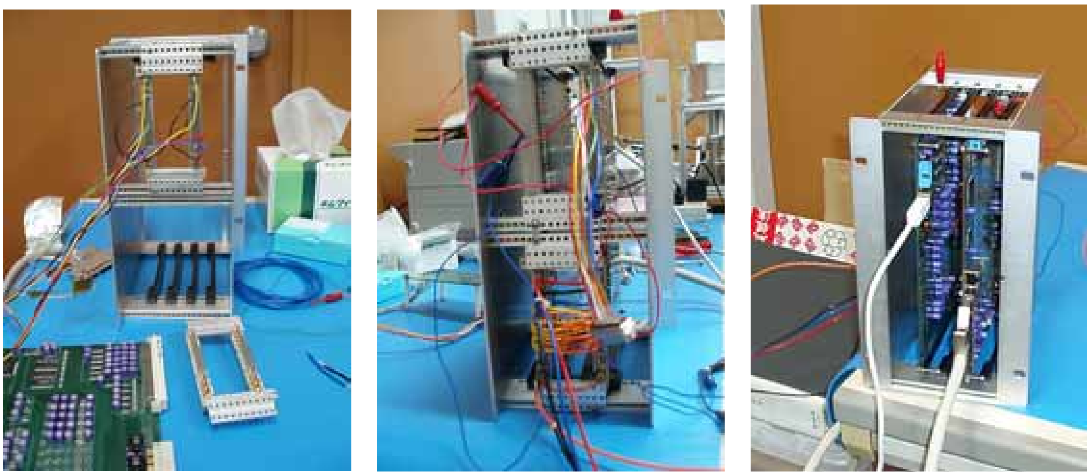


図 23　バックプレーンと配線の様子（左：コネクタ製作、中：バックプレーン、右：表面）

3.5. クロック

流れ図は図 5 を参考にされたい。デジタルクロックは、パソコン・PCI ボードから、ADC ボードを通過し、DRV ボードへ送られる。DRV ボードでアナログクロックが生成され、検出器へと送られる。HAWAII-2 の場合、クロックは 12 本すべて HIGH = 5V, LOW = 0V である。

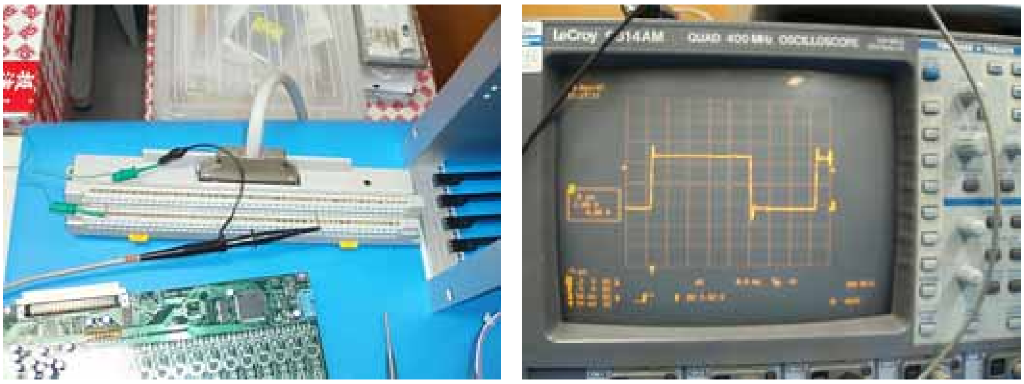


図 24　デジタルクロックの生成テスト：パソコンから生成したクロックの波形

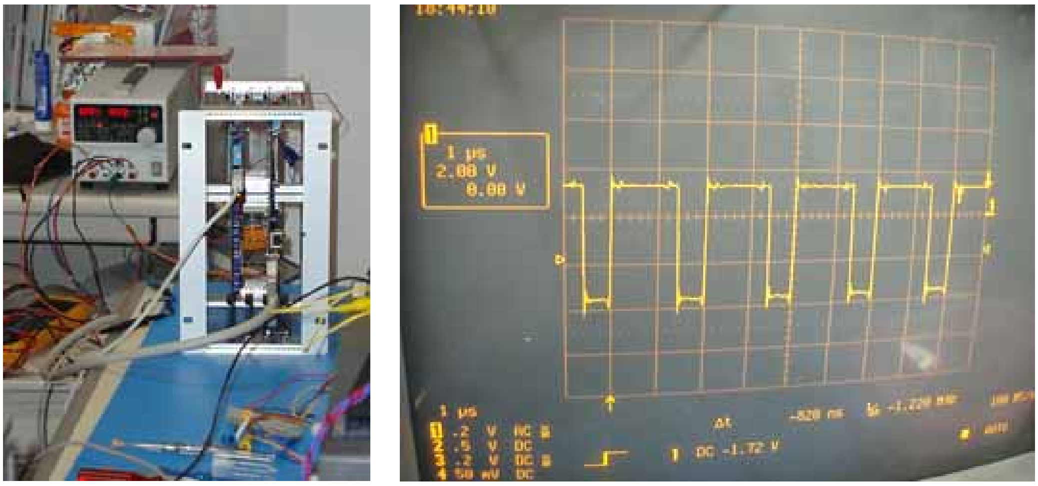


図 25　アナログクロックの生成テスト：DRV ボードから検出器へ送るクロックの波形

HAWAII-2 のクロックは Horizontal 方向に1ピクセル進めるのに必要なクロックが4つあり(CLK1, CLK2, CLKB1, CLKB2)、CLK1 と CLK2 は位相を 25 ns 以内で反転させ、CLK1=CLKB2, CLK2=CLKB1 としなければならない(HAWAII 以前に問題になっていたグロウを減少させるため)。

酒向回路では、パソコンから ADC ボードへ転送するところと、ADC ボードから DRV ボードへ転送するところの2箇所でシュミットトリガを入れており、それぞれで 10 ns ずつ遅れが生じるため、25ns 以内での反転が実現しない。このため、2箇所のシュミットトリガを取り除き、短絡することでクロックを実現する。

また、PCI ケーブルによってパソコンから ADC ボードへクロック転送するとそこでもクロックに遅れが生じるため、LVDS による転送が必須である。

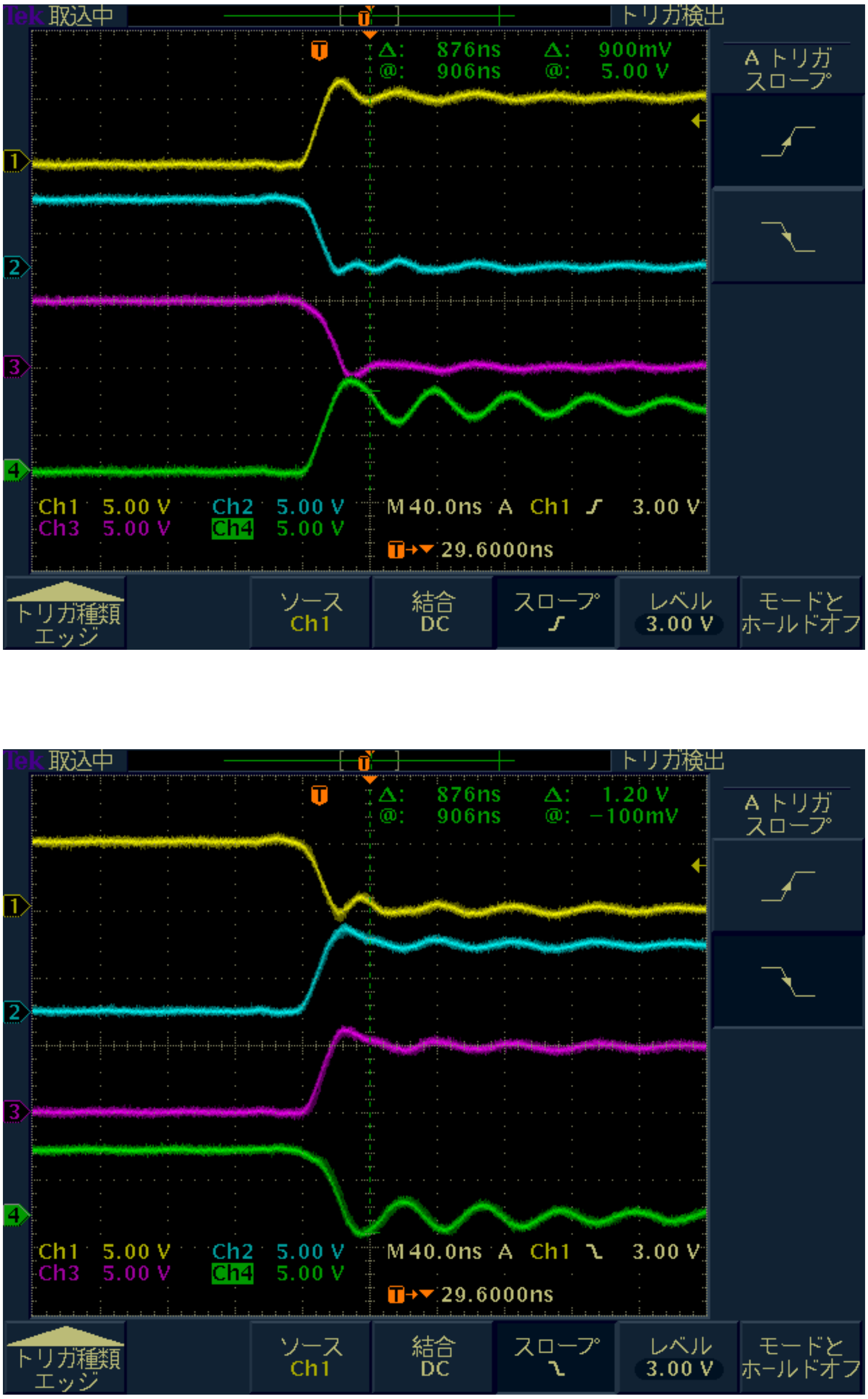


図 26　CLK1, CLK2, CLKB1, CLKB2