

グラウンドが重要な回路のプロトタイピングの方法 (PRBS-16 RTL つき)

著者: 石井 聡

はじめに

とある理由により、図 1 のような基板をつくりました。本来であればプリント基板を起こしたかったのですが、手持ちで無垢基板の古いものがあり、また秋月電子で 0.65mm の TSOP-8、0.8mm の SOT-23 の変換基板が売っていたので、基板を作らずに試作してみようと思いました。

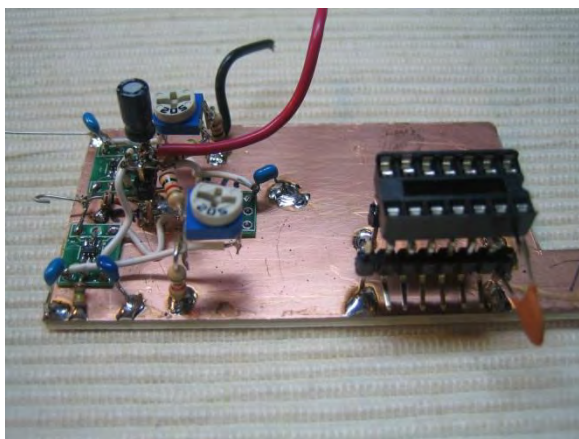


図 1. 作った回路 (製作途中)

アナログ・エンジニアが好んで使う無垢基板

この写真は途中まで作った基板のようすです。無垢の基板をベタ・グラウンドとしています。アナログ回路は「安定したグラウンド」が大切です。ワイヤを使ってグラウンドを手配線すると、どうしてもインダクタンスの問題が生じてしまいますので、このような形が良いといえます。

周りにアナログ回路ご専門のエンジニアの方がいらっしゃると、その人の作業台のうえて、このような基板を見ることもあろうかと思えます。とはいえ、あまりそのような環境ではない方などにとっては、ご参考になろうかと思いましたのでご紹介します。

変換基板で高速コンパレータを実装する

基板上には、高速コンパレータ ADCMP601 を取り付けました。ADCMP601 は 0.65mm ピッチかつ小型パッケージ (SC70) であるため、TSOP-8 の変換基板には実装できず、0.8mm の SOT-23 の変換基板の方にギリギリ乗せて実装しました。

ADCMP601: コンパレータ、超高速、6 ピン SC70 および SOT-23 パッケージ

<http://www.analog.com/jp/adcmp601>

また写真右側に DIP IC ソケットが見えますが、ここにはディレイラインを挿入します。各タップから遅延した出力が得られます。今まであまり良い IC を探すことができませんでしたが、Data Delay Devices 社というものを見つけ、この会社のもの (3D3220 シリーズ) を使うこととしました。

「アナログ回路基板のプロトタイピングのはずでは？」と疑問もお持ちになるかもしれませんが、プロトタイピングの一般論としてご覧ください…。

無垢の銅張基板の加工方法

無垢の銅張基板の加工方法について説明しておきましょう。無垢基板はサンハヤトなどで入手可能です。取引のあるプリント基板メーカーにお問い合わせするのも手かもしれません。

適切な工具を用いて加工する

これを図 2 のようなカッター (これは OLFA の p カッターというもの) の、刃の下側の鋭角部分を基板に当てて、何度かそぎ切るように切り込み (溝) を作ります。ジョリジョリと切り込みが入られます。切り込むようすは OLFA の HP で動画がありますので、参考にしてみてください。

片側を深さ 1/2 程度切り込みを入れて、反対 (裏) 側もちょうと切り込みを入れると、簡単にパキッと折ることができます。



図 2. プリント基板の切断に用いる特殊カッター (OLFA の p カッター)

アナログ・デバイス株式会社は、提供する情報が正確で信頼できるものであることを期していますが、その情報の利用に関して、あるいは利用によって生じる第三者の特許やその他の権利の侵害に関して一切の責任を負いません。また、アナログ・デバイス社の特許または特許の権利の使用を明示的または暗示的に許諾するものでもありません。仕様は、予告なく変更される場合があります。本紙記載の商標および登録商標は、それぞれの所有者の財産です。
 ©2014 Analog Devices, Inc. All rights reserved.

Rev. 0

アナログ電子回路技術ノート

TNJ-011

無垢の銅張基板の加工方法

とはいえ、この無垢の基板は長期間在庫していると化学変化（酸化）して表面の色が変わり、半田付けができなくなってきました。

そこで折り取った基板を以下のようにして対応します。実は今回使用した無垢基板も「ぼろぼろ」になっているモノでした。

- ① クレンザーとスポンジで水を使いながら表面を磨きます
- ② 当然ですが台所や給湯室などのスポンジは使わないください。また実施する場所も考慮してください
- ③ キレイになったところで、エアーガンやドライヤーで乾かします
- ④ 表面にフラックスを塗っておきます。半田付けがしやすくなるとともに、再度の表面酸化から（ある程度の期間ではありますが）保護することができます

このようにして作った基板が図 1 の写真です。「グラウンドを安定にさせるべき」ちょっとしたアナログ回路の試作のときに、ご活用ください。

他の方から頂いた試作方法ノウハウ

他の方からも試作ノウハウ情報をいただきました！

「私は高周波回路を試作するときには、銅箔テープをよく使います。ダンボールを適当に切って銅箔テープを貼り付ければ、全面 GND 基板の出来上がりです。これなら、はさみとカッターナイフだけで作れます」

とのことでした。耐久性は問題ですが、プロトでは充分かと思えます。銅箔テープはホームセンターなどで購入できます。

出来た基板でナニをする？

ここまでの基板も他のデジタル IC も入手でき、図 3 のように、すべてを接続して動作することができました。詳しい説明はしませんが、3.3V ロジックで中間レベルになったところでトリガを出そうという回路です。イメージとして、

ADCMP601 & ADCMP601 ⇒ TC7S00FU ⇒ TC7W74FU

2 つの ADCMP601 がウインドコンパレータになっています。この実装のようすを図 3 に示します。

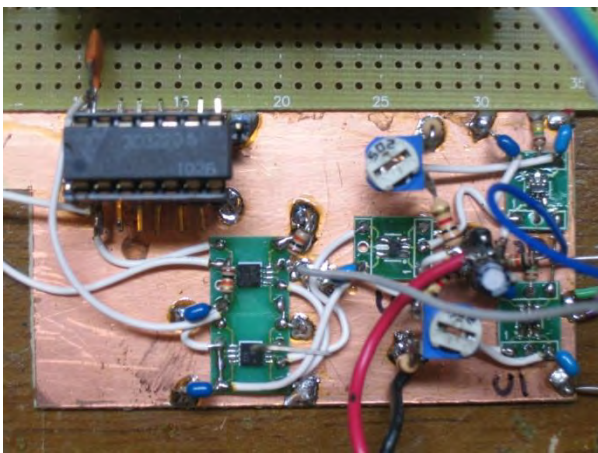


図 3. 作った回路（完成状態）

図 4 の上半分は 3.3V ロジックの中間レベルを生成する回路です。2 出力を抵抗で接続し、PRBS-16（Pseudo Random Binary Sequence-16）の 65535 シーケンスのうち、1 シーケンスで「のみ」論理が H/L になる状態の 2 出力を作り、この出力の間に 2

本の直列を接続します。この 2 抵抗の中間タップから中間電圧レベルを図 5 のように生成します。

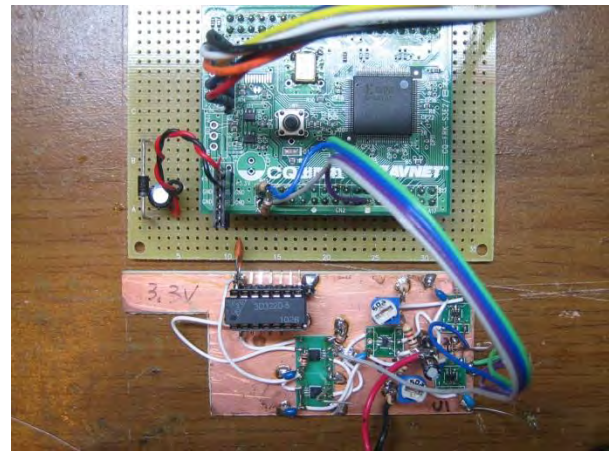


図 4. 入力信号として与える FPGA 基板とともに

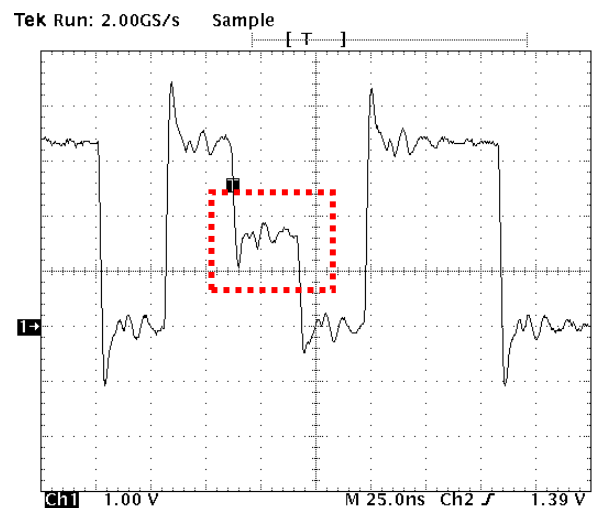


図 5. 入力信号として与えるロジック中間レベルの信号

PRBS-16 には手持ちの FPGA 基板を使用した

PRBS-16 の回路を入れた基板は雑誌付録の FPGA 基板で、ツールは ISE 9.1i です（当時）。「そんな古いバージョン使っていないで、ちゃんと最新のバージョンに上げてくださいよ」…と言っていたのは、弊社代理店様の心やさしいシニア・マネージャの方でした(人)。

作った回路の動作するようす

作った回路は、図 5 のような中間レベルを検出するものです。このような中間レベルは「メタステーブル」とか出力の衝突が発生したときに見られるものです。

この回路図を図 6 に、またこの回路で図 5 の中間レベルの状態を検出した波形を図 7 に示しておきます。

この基板は P 板.com の「パネル de ボード」サービス（アナログ・デバイセズと共同でご提供しているもの）で購入できます。以下の URL をご覧ください。

http://www.p-ban.com/panel_de_board2/

アナログ電子回路技術ノート

TNJ-011

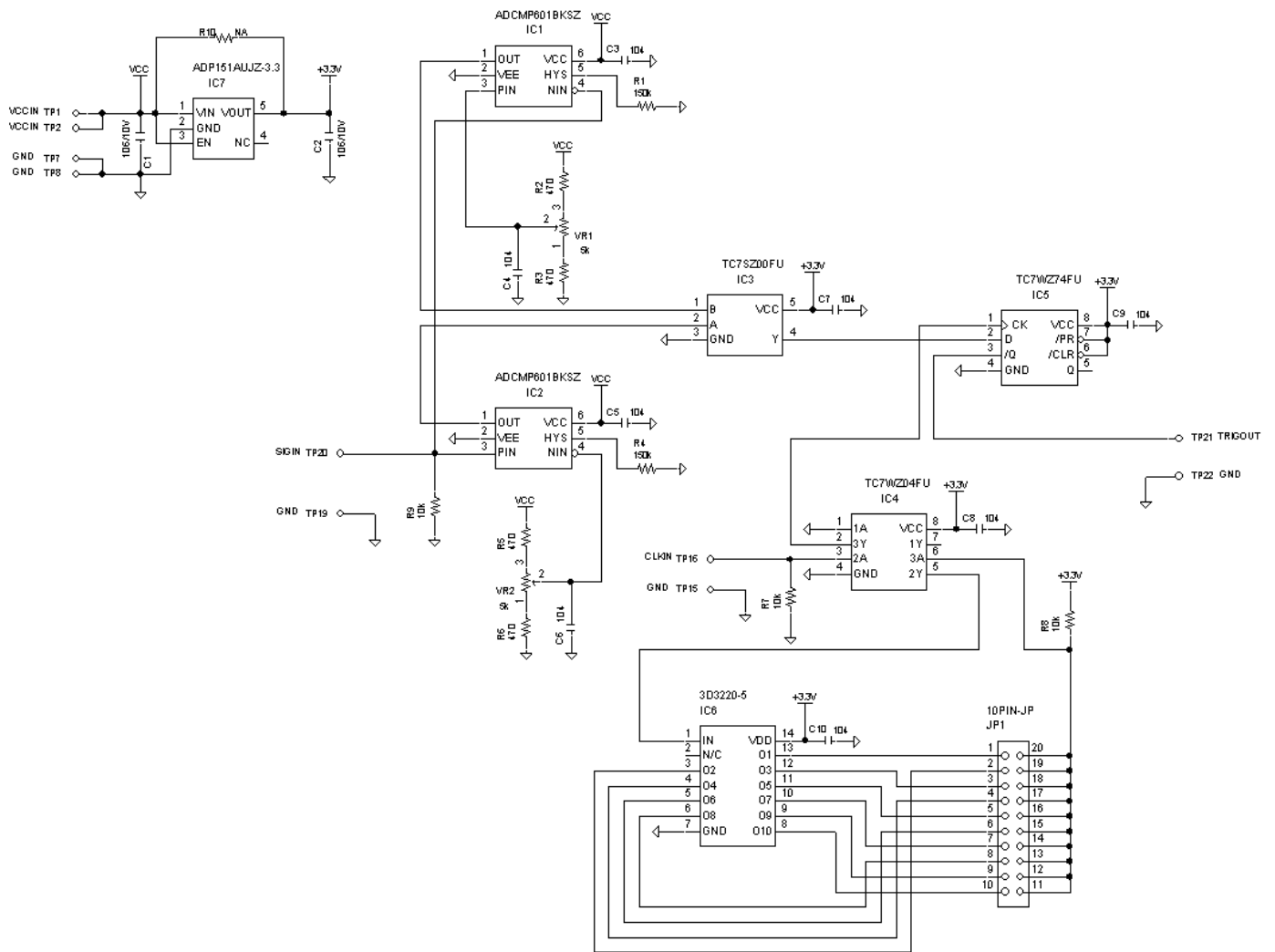


図 6. 作った回路の回路図

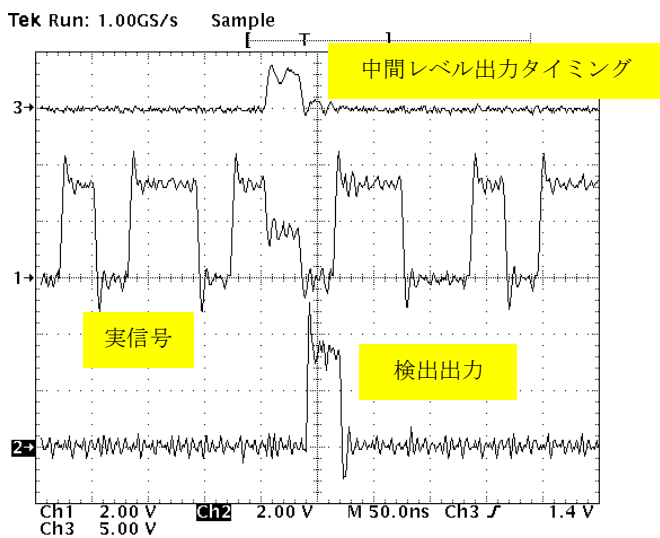


図 7. ロジック中間レベルを検出したようす

アナログ電子回路技術ノート

TNJ-011

FPGA 内部の回路（RTL）のご紹介

アナログ技術ノートの趣旨を逸脱するようなデジタル系の話なのですが、今回作った FPGA から出すランダム・データ PRBS-16 のソースを添付します。entity 以降です。

```
entity PRBS16 is
port (
    CLK, RESET : in std_logic;
    CK_OUT : out std_logic;
    PRBS : out std_logic;
    PRBS2 : out std_logic
);
end PRBS16;

architecture RTL of PRBS16 is
    signal LFSR : std_logic_vector(15 downto 0);

begin
    CK_OUT <= CLK;
    process (CLK, RESET) begin
        if (RESET = '0') then
            PRBS <= '0';
        elsif (CLK'event and CLK='1') then
            PRBS <= LFSR(0);
        end if;
    end process;

    process (CLK, RESET) begin
        if (RESET = '0') then
```

```
            PRBS2 <= '0';
        elsif (CLK'event and CLK='1') then
            if (LFSR = x"ffff") then
                PRBS2 <= not LFSR(0);
            else
                PRBS2 <= LFSR(0);
            end if;
        end if;
    end process;

    process (CLK, RESET) begin
        if (RESET = '0') then
            LFSR <= x"ffff";
        elsif (CLK'event and CLK='1') then
            -- LFSR Shift
            -- Polynomial is  $X(n) = X_{16} + X_{15} + X_9 + X_4$ 
            LFSR(0) <= LFSR(15) xor LFSR(14) xor
                LFSR(8) xor LFSR(3);
            for I in 0 to 14 loop
                LFSR(I+1) <= LFSR(I);
            end loop;
        end if;
    end process;
end RTL;
```