



電気通信大学 工学研究部 部報

2015年 調布祭号

電気通信大学 工学研究部 部報

目次

イライラ棒	2
	坂西 潤哉
中井的PCセットアップ	4
	中井 森希
工研ラジオ製作記	8
	永田 門
艦これコス用の装備	13
	林 唯奈
ペンたて製作	18
	眞下 結衣
パーカッションデバイス 「BEATTECH」	20
	Team. にくろむちゃん
さけるチーズ	22
	三村 京太郎
エレキギター製作記	25
	ジジイ
8bitAVRマイコンによるデジタルエフェクタ	27
	あわざる
GPIB-USB変換器の制作	30
	ふな
工学研究部部報用TEXテンプレート整備計画	32
	横田 嶺

イライラ棒

I科1年 坂西潤哉

1. おことわり

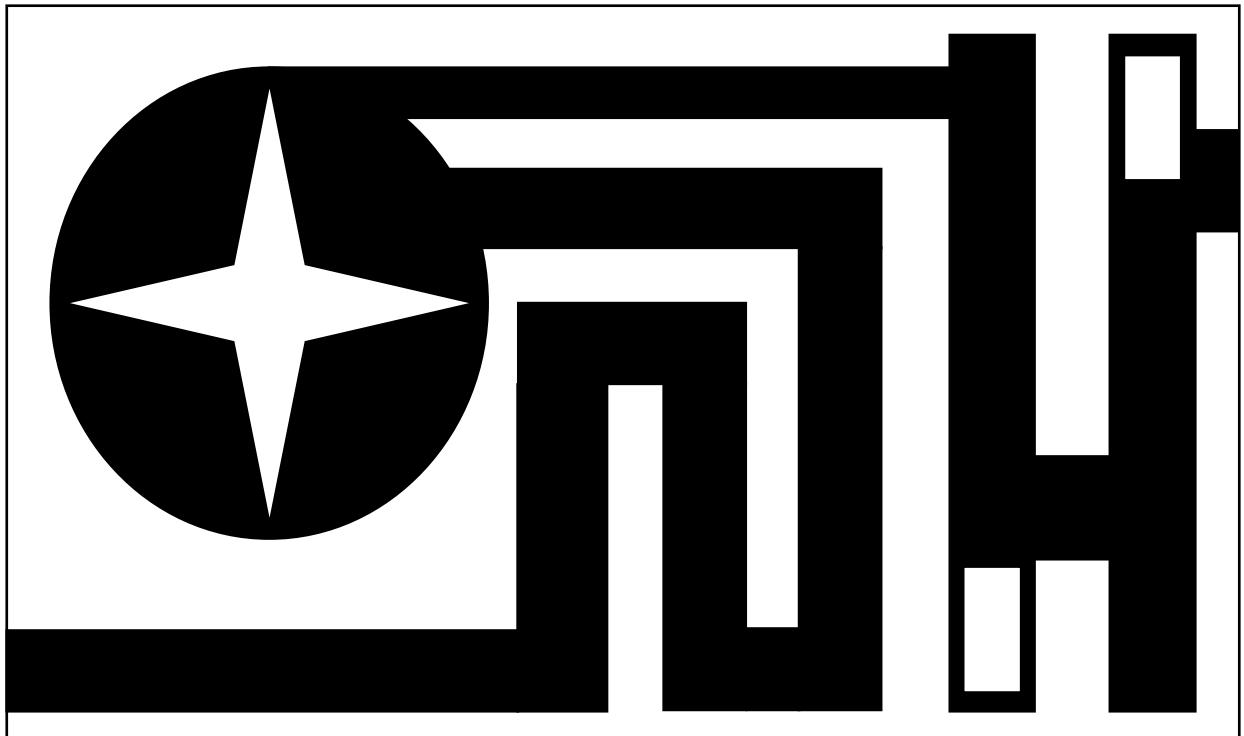
実際僕は工学研究部に入部して少しつが、未だに作品制作に取り掛かっていない。制作の基礎を学ぶため、3D CAD 講習や C 言語講習に参加してきた。しかし、どんな部品を使えばいいか分からず、また夏休み中忙しかったため制作に取り掛かることができなかった。おそらく今から制作しても間に合わない可能性が高いのでここでは「制作予定物」とした。

2. 作るきっかけ

僕はインターネットでイライラ棒のゲームをよくやっていた。そして小学校の頃はイライラ棒を針金で作ったことがあった。しかし、この時のイライラ棒は針金で複雑な道を作った程度であったが、インターネットで見るイライラ棒は風車など動くギミックがある。

そこで、僕は小学校の時のそれとは違うこのような動くギミックを取り入れたイライラ棒を自作してみたいと思った。

3. 作成図



作成図の説明

- ・左がスタート、右がゴールの構成。
- ・動くギミックの1つ目は右回転の風車、図では突起が4つであるが数は変わるべき可能性がある。
- ・2つ目は通路途中の動くブロック。ブロックを上下に動かす。

4. 作品の材料

- ・電源 (ONとOFF切り替えてギミックが動いたりできるようにする。)
- ・針金 (コースを作成する)
- ・枠 (コースを決めてから大きさに合わせて 3DCAD で作る予定)
- ・そのほか電子部品

5. まとめ

「制作予定」になっているうえ、電子部品の知識がほぼないためしっかりととした部法になっていないことは申し訳ない。

しかし、いつかは作成できるよう取り組みたい。

中井的 PC セットアップ

情報・通信工学科 1 年 中井 榛希

1. 怪談

その日はふと郵便局に行こうと思ったんです。ところが、ATM の前ですうっと意識がなくなっていくんです。気づくと、口座から 10 万が消えている代わりに、いつのまにか手には 10 万円を握っていたのです。こわいなあ、こわいなあと思った私は、誰かに追われているような気がして、とりあえず大学生協に駆け込んだんです。しかし残念ながらそれは間違いました。再び気が遠くなっています。

「「「やばい！」」」

そう思った頃には既に手遅れでした。すでに持っていたはずの 10 万円は消え、こんどは右手に新品の Let's note を持っているではありませんか！

2. はじめにはじめに

しょうもない茶番へお付き合いいただきありがとうございました。念のためですが、前章の内容は大部分がフィクションです。あしからず。

さて、この前 念願のノート PC を手に入れました¹。自分はいわゆる宅通²生であり、通学に片道 1 時間 30 分³もかかるので、電車に乗っているあいだの時間を活用できるかは個人的に結構重要だつたりします。小さくて軽いにもかかわらずバッテリーがよく持っているので、よい買い物だったと確信しています。

明らかに 4 月に買うよりも安い割によりスペックのものを手に入れられるので、新入生の皆さん(といってもこれを読んでいる時点ではまだ受験生かもしれません)には、入学時に大学生協が大々的に宣伝する 20 万はする豪華キャンパスモデル⁴のノート PC ではなく、10,11 月頃に在庫処分のためにひっそりと出される 少型落ちとはいえない安くなっている「未開梱品」モデルをおすすめします。安くなっているにもかかわらず生協による 4 年保証もちゃんとついてくるので安心です。

今回買った Let's note CF-SX3 シリーズ キャンパスモデルのスペックを表 1 に示します。

表 1: 初期状態のスペック

OS	Windows 7 Pro 64bit(日本語)
CPU	Intel Core™ i7-4510U
RAM(メモリ)	4GB DDR3L SDRAM
HDD	750GB SATA

- 1 実は今まで持っていないわけではなかったものの、処理が遅い・デカイ・重い・バッテリーが消耗しきっているなどのため持ち運び用途としては使い物にならない状態だった。
- 2 自宅通学の略か。一人暮らしではなく、実家から大学に通うこと。
- 3 往復 3 時間以上、一日の 1/8 に相当する時間を使っていることになる(!)。もっとも、大学の近くで一人暮らしをはじめるとしても家事等にこれ相当の時間は消えそうだ。
- 4 例えばタッチパネル付きである。もちろん人によるが、情報系の学生なら基本キーボード作業なのでタッチパネルは不要なはず...?。

光学ドライブ	DVD スーパーマルチドライブ
LCD	12.1 型 16:9 TFT 液晶 1600x900
無線 LAN	IEEE 802.11a/b/g/n/ac (2.4+5GHz)
Bluetooth	Bluetooth v4.0
端子	SD カード x1, USB2.0x1, USB3.0x1, LAN, HDMI, RGB ミニ D-sub15 ピン, マイク端子, イヤホン端子
カメラ・マイク	1280x960, モノラルマイク
質量	1.40kg(バッテリパック L 装着時) 1.19kg(バッテリパック S 装着時)
付属ソフト	Office2013 Pro Premium (Word, Excel, Outlook, PowerPoint, OneNote, Publisher, Access)

3. 実際の手順

この章では、PC セットアップ時に実際に導入したソフトなどを紹介していきます。

あ、遅れましたが、この文章は脚注⁵マシマシ⁶でお送りします。読みにくいかもしれませんがご容赦ください。

3.1 Windows 7 編

初回起動時は、アカウント名やパスワードなどを設定し、セットアップと最新の Windows Update の適用が終わるまで数時間待ちました。

プリインストール⁷の不要なソフトをいくつかアンインストールすれば、いよいよ自分流カスタマイズの時間です。

自分の場合は、自宅にデスクトップ PC があるので、それに今までに入れたソフトを参考にしながら一通りを入れました。それらの一部を表 2 に示します。

それから、学内 LAN の UEC_Wireless⁸を使うために MAC アドレス⁹を登録しました。

Win+R でコマンドプロンプトを開き、/ipconfig /all

で Wireless LAN adapter ワイヤレス接続 を探し、16 進数 2x6=12 衡の物理アドレス(=MAC アドレス)を調べます。

これを情報基盤センターの PC から登録すれば、統合認証の ID とパスワードで UECWireless を使うことができるようになります。

5 文章中に書くとメインの文章の流れが阻害されるときなどに、このように枠外に補足を書くこと。とはいえる普通はレポートや論文にこのようなムダ話をつらつら書いていると確実に怒られる。

6 増し増し。ラーメン二郎における注文時に唱える”呪文”において、トッピングを增量するときの「増し」を重ねたもの。ちなみに、電通大に最も近いラーメン二郎は仙川(調布駅から京王線で 5 駅)にある。また二郎インスピア系の「郎郎郎(さぶろう)」や「夢走家」などのラーメン店が調布駅前にある。

7 preinstall(ed)。pre-:「前」の意。メーカーによってあらかじめインストールされているソフトウェア群。そのパソコン独自の機能を使うために必須のため消しては困るものと、実質広告に近いさっさとアンインストールすべきものまで様々。最近では Lenovo がプリインストールのプログラムにバックドア(メーカーがユーザーの情報を裏口から盗み出せる機能)があったとニュースになっている。

8 学内関係者専用の wifi です。調布祭期間中だとしても来場者用に開放などはしません。念のため。

9 Apple の Mac とは関係ない。Media Access Control address。

表 2: Windows 7 環境にインストールしたソフトウェアの一覧(一部)

ソフトウェア名	説明
Google Chrome	Google 製のブラウザ. ちなみに, Windows に初期状態で入っている Internet Explorer は Google Chrome をダウンロードするためだけのソフトとよくいわれる.
Firefox	普段使いのブラウザは Chrome などの, NPAPI ¹⁰ 廃止に伴ったまに見られないサイトがあるので念のため入れた.
Google 日本語入力	IME, 漢字を変換するためのソフト.
LINE	PC でファイルを受信したいときに PC 版 LINE を使うことがある.
GitExtensions	git ¹¹ の GUI クライアント.
LibreOffice	オープンソース ¹² でありフリー ¹³ のオフィススイート. 表計算・ワープロ・プレゼン作成などができる. OpenOffice から派生した.
美咲フォント	8x8(スペースを含めないと 7x7)のドットフォント. こんな感じのピコピコした FONT.
源ノ角ゴシック(Noto Sans CJK JP)フォント	Adobe と Google が開発したフリーの日本語フォント. こんな感じのきれいな FONT.
Softalk	AquesTalk 社の音声合成ソフト. いわゆるゆっくり.
Domino	MIDI 編集ソフト. 筆者は音楽打ち込みが趣味の一つであるため入れた.
AtmelStudio	工研でメインで使われているマイコンである AVR のプログラムを開発するための環境=IDE. VisualStudio ベース. 真っ赤なてんとう虫アイコンが目立つ.
7z	7zip. 世の中の大多数の圧縮ファイルを解凍・圧縮できるすごいやつ.
HSP	HotSoupProcessor. BASIC ライクな Windows 用プログラミング言語. 筆者はこれでプログラミングを経験したので入門としてお勧め. しばしば中級者以上からは敬遠されるきらいがあるが, GUI アプリケーションを手軽に作れるのが魅力. 最近はブラウザや iPhone/Android 上でも動かすことができる.
Stirling	スターインと読む. バイナリエディタ. さまざまなデータを 16 進数で覗いたり編集することができる. 情報系の人には欠かせないツール.
TeraTerm	有名なターミナルソフト. マイコンとシリアル通信でデバッgingしたり, 大

10 Netscape Plugin Application Programming Interface. ブラウザのプラグインの実装方法の一種. Flash, Java, Unity などはこれを用いているため, 最新版の Chrome ではこれらを用いているサイトは動作しない.

11 バージョン管理システム. プログラムなどの変更履歴を管理することができる. git を利用したものとしては GitHub(<https://github.com/>)が有名.

12 Open source. 誰でもソース=プログラムの中身を知ることができるという意味. 説明不足だが, あまり突っ込むとマサカリが飛んで來るのでゴメンナサイ.

13 free. 正確には無料と同義ではなく, 自由(制限がない)の意味. このあたりも面倒な話があるので突っ込みません.

	学のアカウントに SSH でつなぐなどよく使う.
RealTerm	こちらもターミナルソフトだが、シリアル通信メイン。通信の内容を 2,8,16 進数、ASCII、任意の文字コードなど多種のフォーマットで見ることができるなど、デバッグ ¹⁴ 時に便利。
Skype	音声・ビデオ・テキストでチャット・通話ができるソフト。
SoundEngine	音声編集ソフト。機能が多く便利。
RadioLine	音声を簡易的にミックスできるソフト。
MicrosoftSecurityEssentials	Microsoft 製の無料のウイルス対策ソフト。通知がうるさくない反面、検出率が低いといわれている。
Java	あなたと Java, 今すぐダウンロード ¹⁵

3.2 ArchLinux 編

Windows のセットアップが一通り終わったら、こんどはデュアルブート¹⁶で Linux を入れたくなりました¹⁷。ただ、一口に Linux といっても、Ubuntu, Debian, Cent OS, Fedora, Linux Mint, Open SUSE, Gentoo¹⁸, ...などなど、選択肢はかなりある中で、どれにしようかと小一時間悩んだ結果、ArchLinux(アーチ リナックス)を入れることにしました。

...ここで筆者は原稿〆切の時間切れとなってしまいました¹⁹。というわけで ArchLinux 編は次回部報に掲載します²⁰。

4. ✎

部報のような文章を書くことははじめてでしたが、なかなか技術っぽい記事にできなかった気がします。精進します。

14 発音的につい「デバック」と言ってしまいそうになるが、debug なので”グ”が正しい。simulation をシミュレーション(正:シミュレーション)と言ってしまうのと同じ類い。言語学は奥が深い。

15 「ド」だけれているのはわざとである。Java のダウンロードサイトにあったこのキャッチコピーが、最後の文字「ド」だけはみ出て改行されてしまい、体裁が崩れたまま何年も放置されていたために有名。

16 1 台のマシンで複数の OS を起動できるようにすること。dual boot。Dual は 2 の意味だが複数の場合でも Dual を使うような気がする。

17 入れたくなるもんです。(説明になってない)

18 電通大と Gentoo をかけて Dentoo 大と呼んだり呼ばれたりすることがある。ただし実際に Gentoo を入れている人はそこまでいるのではないだろうか(それでも少なくとも東京都内での人口密度は屈指のはず)。MMA(電通大の情報系サークル)主催の LT(Lighting Talk, 5 分程度の短い発表)大会は Dentoo LT という名前がついている。興味がある人は Twitter でハッシュタグ #dentooLT で検索。なお MMA の回し者ではない。

19 だいたい脚注を書き過ぎたせい。エレクトロニクスコンテストとロボメカ工房 小中学生ロボコン担当を掛け持っていて忙しかったのもある。

20 たぶん。

工研ラジオ製作記

S科 一年 永田 門

1 はじめに

工学研究部には、サーバー用のパソコン以外に2つWindowsのPCがある。そのうちの一台にはこれまでの工研部員たちのものであろうCDのデータがなんと一万曲以上も入っている！これをインターネットストリーミングラジオにしない手はない！そこで、ちょうど興味を持っていたサーバーの管理、そして自分の家でやっていたUbuntuにIcecast2+Ices0.4の技術を応用させて実現することにした。これは、その一部始終を記録したものである。

2 用意するもの

PC … Linuxディストリビューション、CentOSかUbuntu。私はUbuntuをおすすめする。

Icecast … バージョンは2.0以降であること。

Ices0 … バージョンは0.x系であること。もし、最新版の2.x系を使ってしまうとmp3エンコーディングができない。

Perl-dev … これをインストールすることで、リクエスト機能を実装することができるようになる。

libMP3, libOgg, libxml, libxslなど … これらをインストールしなければストリーミングする際のコードができなくなる。また、これら以外にも必要であればインストールする。

Icecast2とices0はUbuntuではapt-getですぐにできる。が、CentOS系だと、実際に公式ホームページ(<http://icecast.org/>)にアクセスし、それぞれのtarファイルを落とし、解凍した後

```
./configure -> make -> makeinstall
```

をしなくてはならない。

3 icecast2の設定

icecast2をインストールし終えると、自動的に/etcディレクトリの直下にicecastというディレクトリが作成される。これの中にicecast.xmlというファイルがあるので、それを編集することで設定を行う。ちなみに、Debian系ではインストール時にadmin-passwordとsource-passwordの設定を行う画面が自動的でてくるので、あまりicecast.xmlでの設定はしないのだが、Redhat系ではその画面が出ないので自分で設定する。

authenticationのadmin-user … Icecastの管理者権限のユーザー名。

authenticationのadmin-password … Icecastの管理者権限のパスワード。バレない程度のものを設定すること。

authentication の source-password …Icecast にストリームを提供する際に使用するパスワード。やっぱりバレない程度のものを。

listen-socket の port …お好きなポートを。ただしウェルノウンは避けたほうが良い。

master-server …おそらく 127.0.0.1 が正解。

master-server-port …listen-socket の port といっしょ。

mount の mount-name …これが、ストリーミングのデータとなる。基本は/live というようにかならず最初にスラッシュを入れる。

mount の charset …Icecast のストリームデータで文字化けしてしまうときに、これを UTF-8 に設定することで回避できる。

パスワードはバレない程度のものを指定する。忘れてしまうと今後に関わってしまう。ちなみにデフォルトでは hackme である。セキュリティ性がなさすぎる。

4 ices0 の設定

ices0 の設定ファイルは Debian 系では /etc/ices.conf、RedHat 系では /usr/local/etc/ices.conf である。この設定は多いのでそのままコピーする。

```
<?xml version="1.0"?>
<ices:Configuration xmlns:ices="http://www.icecast.org/projects/ices">
  <Playlist>
    <!--Type で builtin を指定した時、
        ここに「ファイル名が改行で区切られている」
        プレイリストテキストを指定する。-->
    <File>playlist.txt</File>
    <!-- Type が builtin のとき、1 であればランダムに再生する -->
    <Randomize>1</Randomize>
    <!-- builtin、perl、python の 3 つのうちのどれかを指定。
        perl であれば perl-dev、python であれば python-dev が入っていないと使えない。-->
    <Type>perl</Type>
    <!-- 基本はそのまま。 -->
    <Module>ices</Module>
    <!-- libmp3lame による再エンコードが可能なとき、クロスフェードができる。
        その秒数を指定する。-->
    <Crossfade>3</Crossfade>
  </Playlist>

  <Execution>
    <!-- 1 のときデーモンで起動する。 -->
    <Background>0</Background>
    <!-- Set this to 1 if you want to see more verbose output from ices -->
    <!-- 実はよくわかっていない。-->
    <Verbose>0</Verbose>
    <!-- デーモンモードの時はここにログが記録されるといっているが、
        デーモンでなくとも記録はされている。-->
    <BaseDirectory>/tmp</BaseDirectory>
  </Execution>
  <Stream>
    <Server>
      <!-- ホストの名前、今回は localhost の ip アドレス -->
      <Hostname>192.168.10.10</Hostname>
      <!-- Icecast が開くポート -->
      <Port>11235</Port>
      <!-- icecast.xml の source-password -->
      <Password>hackme</Password>
      <!-- http で配信するので http 一択。しかし、他のも有るらしい。-->
      <Protocol>http</Protocol>
    </Server>
  </Stream>
</ices:Configuration>
```

```
</Server>
<!-- icecast.xml の mount-name と同じ名前を指定 -->
<Mountpoint>/live</Mountpoint>
<!-- 基本的にこのまま
<Dumpfile>ices.dump</Dumpfile>
-->
<!-- 素敵なラジオのお名前 -->
<Name>BOKU NO RADIO</Name>
<!-- ジャンル -->
<Genre>Genre IPPAI</Genre>
<!-- 説明 -->
<Description>BOKU NO RADIO DAYO</Description>
<!-- ストリームへの URL -->
<URL>http://hoge.foo.bar.org</URL>
<!-- 1のとき、Icecast が世界中に公開する。 -->
<Public>0</Public>

<!-- ビットレート -->
<Bitrate>128</Bitrate>
<!-- libmp3lame がインストールされている時
     再エンコードを有効にするかどうか。1なら有効 -->
<Reencode>1</Reencode>
<!-- 再エンコードが無効の時、サンプルレートを指定する。
     <Samplerate>44100</Samplerate>
-->
<!-- 再エンコードが可能なとき、チャンネル数をどうするか。 -->
<Channels>2</Channels>
</Stream>
</ices:Configuration>
```

5 流すためのデータ

部室を漁って出てきた HDD160GB にデータのあるパソコンから WinSCP などで送信した。しかし、かなり窮屈であった。

6 まずは、builtin で流してみる

たとえば、自分のホームディレクトリに playlist.txt というファイルがあり、
/var/music/aaa.mp3
/var/music/bbb.mp3
/var/music/ddd.mp3
/var/music/www.mp3

という内容のとき、builtin はこの内容を上から、また Randomize されるときはランダムに選択され、ストリーミングされる。されないときはファイル指定を間違えているか、Type を builtin にしていない。しかし、これではただ垂れ流してるので perl を使った動的なプレイリストを作成する。

7 perl による動的プレイリスト

本使用しているモジュールを載せるとあまりに長くなってしまうので概略のみ記述する。モジュールは Debian 系、RedHat 系どちらも /usr/local/etc/modules/ に存在する ices.pm.dist を同ディレクトリに ices.pm という名前でコピーする。モジュールには CPAN などからの perl のパッケージを使えるのでいろいろと変化に対応しやすい。また、モジュールは ices_get_next というサブルーチンが存在しないと動かない。このサブルーチンで次の曲のパスを返せば良い。

あとは、ices.conf の Type を perl にすればうまくいく。はず。

8 プレイページ

ここは、お好みであるがリクエストフォームや、現在再生している曲の情報を載せるのは Apache を使って作ったほうがいいかもしれない。しかし、これ以上はどちらかと言えば Apache によるウェブサーバーの記事になってしまないので今回は控える。次号の部報を乞うご期待！ちなみに、その記事は多分中井くんとの共同記事になると思う。

9 おわりに

この部報をラジオを聞き流しながら書きました。作業が捲ります。最後に、このサーバーを聞いてくれた皆さんのが感想を。

このサーバーのおかげで仕事(研究)が捲りました！ありがとうございます!! (24歳男性) - れいてんし
許すまじ CentOS - なぞのひげおじさん

ウェブページを作ったのはほとんど僕だけどね - マルチバイプレーターの擬人化ちゃん

艦これコス用の装備

先進理工学科 1年 林 唯奈

友人がコスプレしたいから作ってくれと依頼してきたので試しに作ってました。

試作品なので工具、材料はすべて100均縛りで作成しています。

また、背負っている大型装備も作りたかったのですが、

会場まで運べないからと断られてしまいました(´・ω・｀)

なので今回手持ち装備のみの作成です。

依頼されたキャラ

- ・磯風
- ・叢雲改二



使った材料

- ・EVAボード (スポンジみたいなシートです)
2.5cm, 4.0cm 青 (途中から売り切れてしまったので緑)
- ・水鉄砲等、おもちゃの銃
- ・グルーガン 黒
- ・黒スプレー
- ・パステル
- ・その他カッター、ハンダゴテ、ライターなどの工具 等

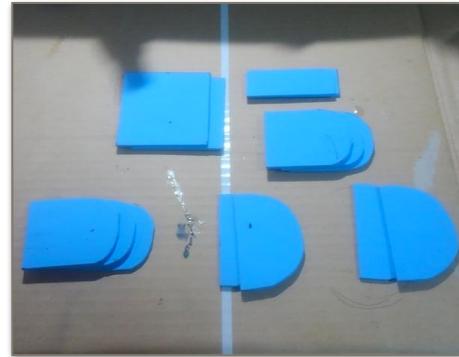
消耗品だけなら総額1000円程度でしょうか 安く済みそうです

大雑把な作り方

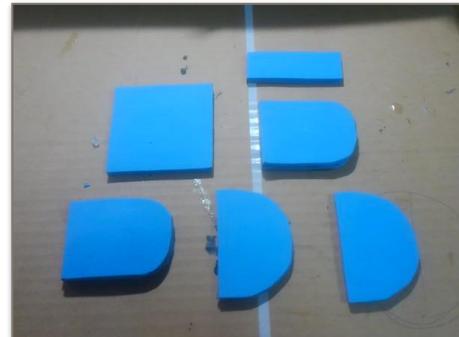
まずは、資料集めて想像から簡単にお絵かきします。試作品なので設計図は無いです。

例えば、叢雲と磯風改の装備は12.7cm連装高角砲(後期型)、いわゆる浦風砲だと思うので、その元ネタと言われている「四〇口径八九式十二糰七高角砲B1型」の画像を拾ってきたり、プラモ眺めたりしてました。

まず、目分量で必要な大きさに切り抜きます
下書きをほとんどせずに直接切り出していくので
なるべく同じ形になるよう頑張りました



好みの厚さが無かったのでボンドで貼り付けます
一番めんどくさい作業です



貼り付けた面がガタガタだったので
ライターで炙って角を落とします
やり過ぎると燃えて解けてしまうので注意です



コンビニの黒ストローを適当に組み合わせた
銃身ぽいものと先程までの青いのを組み合わせます
ストローは 1Ω 抵抗を切った針金で固定してあるので
一定の角度まで回転できるようになっています



グルーガンで鉛みたいな模様を書きます
接着以外の方法でグルーガンを使うのは初めてでしたが
思ってたより難しかったです
角度を変えつつクルクル回すと上手く丸が描けました

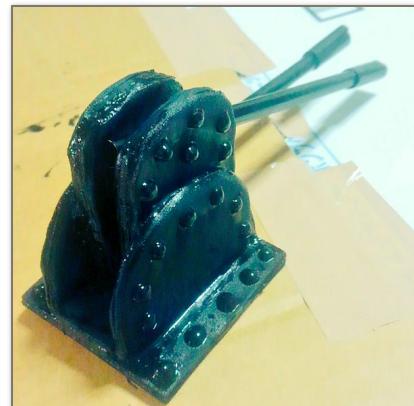


窓の外の物干し竿に吊るしてあるダンボールの中で

塗装します

100均のスプレー

予想よりいい色でした



ここからは飽きるまでパステルで金属っぽく見えるよう試行錯誤しながら影を付けます

正直金属っぽさは無いですが最低でも質感がスポンジ以外に見えればいいかなーと

100均の銃に適当な丸い蓋みたいなものと一緒に貼り付けました



写真の光度とかばらばらですが比較になりそうな画像あったので一応載せてみます



完成形の写真を撮り忘れたので最後の塗装と装飾前ですが

だいたいこんな感じに仕上りました

ここまで作業時間は約3時間です

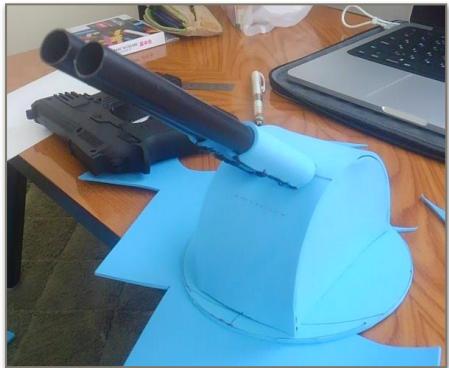
実物は粗が目立ちますが写真撮るぶんには問題なさそうですね

試作品の予定でしたがこのまま友人に提供しても

大丈夫そうでよかったです



他の装備の作り方はほとんど同じなので製作中撮っていた写真だけ公開します
Twitterで晒しながら作っていた間の写真しかないので雑ですが



とりあえず浦風砲はこんな感じに仕上りました
この下に100均の銃を切断や装飾加工したものを持ちて代わりに取り付けます



叢雲ちゃんのミミノアーレ等の写真は撮り忘れたのでありません
カチューシャに固定するつもりですがバランスが悪く試行錯誤しています
先端は中に仕込んだペンライトの電源を入れると一応光るようにしました

余った材料で連装砲ちゃんも作ったのですがこちらも画像が行方不明です

電子的な工作をしばらくしておらず、このようなことしか記事に出来なかったので
時間があったら電気で動くものも作りたいです

予算次第ではちゃんとライオンボードにジェッソ塗装してから作る
丈夫で発色のいいモノを作りたいですが、
安い素材でも塗装と造形しだいでなかなかよさ気なものが作れることがわかったので
今後も依頼が来たら作ってみようと思いました

ペン立て製作

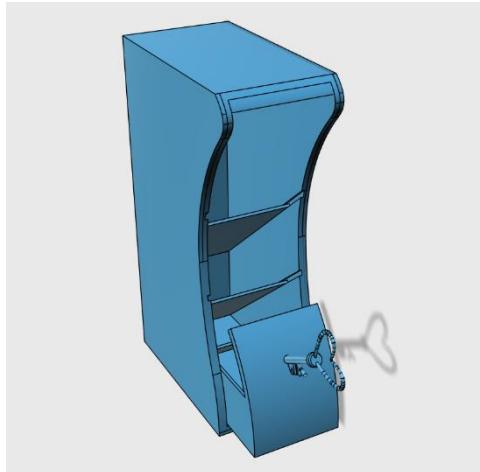
知能機械工学科1年 眞下結衣

1. 製作理由

筆記類をずっと放置していても、ほこりを被らないペン立てがほしかったので作ってみました。CADの練習と3Dプリンターの使用も兼ねています。

2. 製作過程

123D Designという無料3Dアプリを使ってペン立てのデザインをしました。



斜めにペンが入ります。
屋根が大きく、
ほこりが入りにくくしてあります。
一番下の引き出しには、鍵の形をした取っ手
が付いています。



鍵には自分のイニシャルを。

3. 3Dプリント

MakerBot Replicatorという3Dプリンターを使って印刷をしました。

印刷後はニッパーでサポート材を除去し、紙やすりをかけました。

印刷の途中で一度フィラメントが詰まってしまい、本体を2回に分けて作りました。写真は接着剤でつなげたものです。未完成ですが、これから接合部分を埋めて、塗装をしてアンティーク調にする予定です。

実際に印刷してみると鍵が大きめだったので、取っ手は作り直す予定です。



パーカッショントバイス 「BEATTECH」

Team.にくろむちゃん

私たちはチームを組み、ものづくりコンテスト@調布祭(エレクトロニクスコンテスト)に出場した。私たちの製作物について、概要を部報にまとめた。

1. はじめに

私たちは聴覚だけでなく体全体でビート、リズムを体感できるようなデバイス開発を目指した。新感覚のパーカッショントバイスなので、既存のドラムパッドなどを使用した場合とは違う、新たな音楽の開発も目的としている。

2. 概要

上半身全体でビートを刻むことで、音との一体感を味わいたい。それを実現するデバイスを開発した。下図は簡単なデバイスの仕組みである。

ユーザーが服に内蔵されたスイッチを叩くと、スイッチの入力情報がマイコンボード(Arduino Uno)に送られる。スイッチはマトリクススイッチを使用し、Arduino Uno のピン数に間に合わせた。スイッチが押されたバイト列で外部のコンピュータにシリアル通信を用いて転送し、コンピュータは送られたバイト列を元に設定された音を鳴らす。

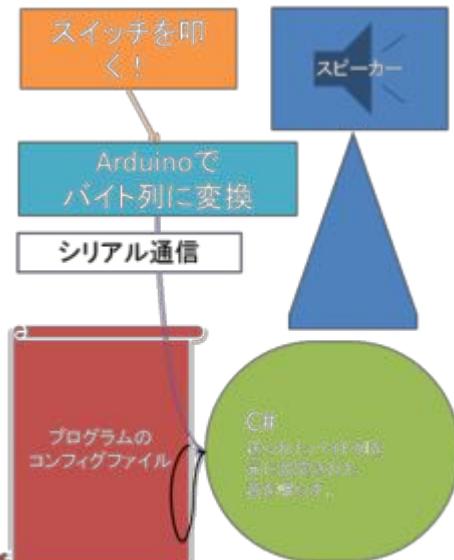


図 2(左). チャート

3. 特色

デバイスはスイッチなどを埋め込んだパーカーである。パソコンと繋いでプログラムを立ち上げるだけで音を鳴らすことが出来る。さらに、対応する音や、ループサウンドなどを自分で設定できるので思い思いにビートを刻むことが出来る。

4. 展望

Arduino Uno を使用している現状を、Raspberry Pi などに代え、スタンドアロンで使用ができるようになりたい。また、音源の追加、鍵盤の実装を目指す。

さけるチーズ

知能機械工学科1年 三村 京太郎

1.はじめに

先日、ふと思いついたアイデアがあります。それを形にしようということで、考えがまとまりましたところまで書きます。記事作成時ではまだ構想段階です。形になり次第報告したいと思います。

2. アイデアの概要

タイトルから察しのついた方もいらっしゃるかもしれません。そう、避けるチーズです。出落ち感を振りまいてしまします。ある日の夜遅くに、TVを観ていたところ、さけるチーズのCMが流されました。さけるチーズ... そうだ、避けるチーズだ。食べようとチーズに手を伸ばすと、ささっと逃げてゆく、"さける"チーズだ！と思いつきました。センサーで近くにものがあるかどうかを判定して、チーズを行動させる予定です。

3. 必要なもの

構成を考えて、パーツリストを作りました。ひとまずこんなところでしょうか。

外装	さけるチーズのパッケージをプラ板で補強
センサー	赤外線 LED、フォトトランジスタとその信号增幅用にトランジスタ等
マイコン	ArduinoUNO 互換 Arduinoは使ったことがあるので開発がスムーズに出来そう
動力	モーター (FA130) 2個をモータードライバーTA7291Pで駆動
電源	単3乾電池、100均で購入した乾電池式携帯充電器により5Vに昇圧

4. 構成案

パーツの配置案を考えてみました。

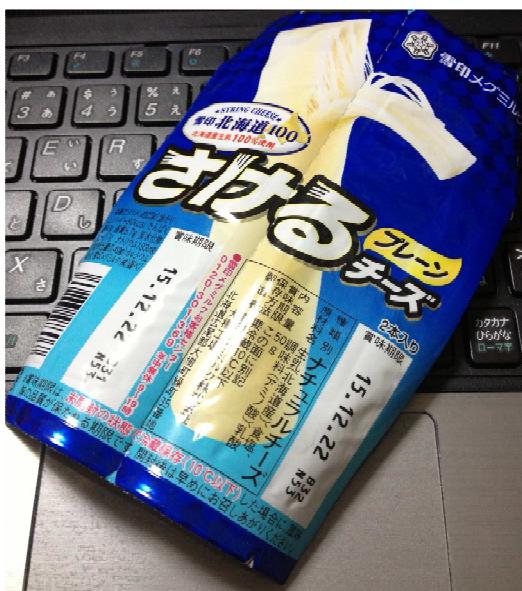


図1. さけるチーズ



図2. さけるチーズ 裏面



図3. 配置案

フォトトランジスタ
赤外線 LED

このように、チーズが2本入っているバージョンのパッケージを用いる予定です。

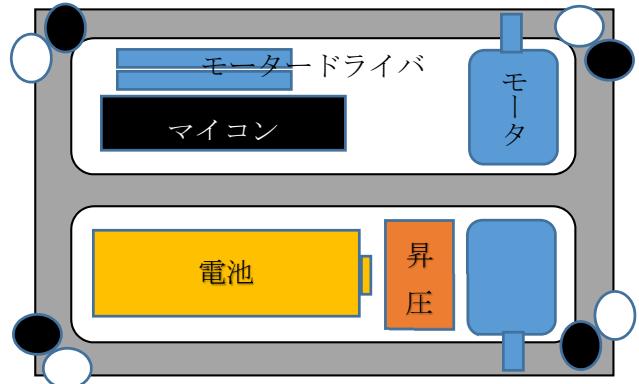


図4. 配置案

5. 今後について

あとは実装あるのみです。さけるチーズのパッケージに詰められるかという問題がまずあるので、より小さなパーツに置き換える等工夫していくらなと考えています。

6. 先行例

すでに、同じ発想に至った例がいくつかあるようです。調べてみて初めて知りました。しかし実際にものを作った例はまだないようです。実装するという点では新しいといえると考えます。



図5. 先行例[1]



図6. 先行例[2]

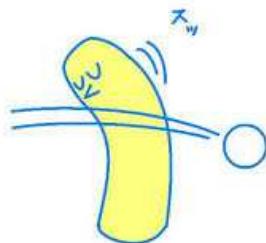


図7. 先行例[3]

7. 参考文献

[1] (▼▽^0)<マホダヨ-@qquuiieett 「TLで避けるチーズとか聞いたから…」

<https://twitter.com/qquuiieett/status/514414344742789120>

[2] T-SHIRTS TRINITY, 避けるチーズ, <http://ttrinity.jp/sp/shop/iromonoya/design/1554509>

[3] Twitter もーす@moatrice, 「避けるチーズ」,

<https://twitter.com/moatrice/status/436417169065127936>

エレキギター制作記

知能機械工学科2年 ジジイ

1. はじめに

みなさんこんにちは。前回の部報ではエレキギターの構造と制作案について書きましたが、今回はその製作案を元に実際に作ってみました。製作案についておさらいすると、P-90のリアピックアップ一発、ワンボリューム、ワントーン、ネックは他のギターからの流用、キルミーでペイバーである（ここ大事）といった具合です。

キルスイッチは時間と加工の関係から割愛しました、すみません。それでは早速製作過程を見ていきましょう。

2. 木材の選定、寸法決め

まずはギター用の木材を仕入れるところから始めます。今回はギター用木材通販サイトのakaseさんから木材としてはスタンダード、安価であることから3Pアルダーを購入しました。厚さが50mmと少し厚めなのがこの後の加工で泣くことになる。製作には40mm程がよろしいでしょう。

次に寸法決めです。エレキギターの制作に重要になってくる寸法はブリッジ部からナットまでの長さです。その長さによってロングスケール、ミディアムスケール、ショートスケールと分かれていきます。今回は流用するネックの関係上ロングスケールを採用します。ブリッジを取り付ける場所がずれてしまわないように気をつける以外の部分に関しては割りと適当にやっています。適当ですみません。

楽器としての最低限の寸法が決まったら、デザインに沿ってどのように切り出すか決まつたらおしまいです。



図1. 寸法を書きこんだ木材

3. 木材加工

寸法が決まれば次はいよいよ加工です
まずは外形に合わせて木材を切っていきます。使った工具はバンドソー、円ノコ、木工用のグラインダーです。前述のとおり厚さが50mmであり自分の加工技術が未熟なのでかなり苦労しました。

外形が切り出せたら穴あけ、溝加工です。ここでは電動トリマーと卓上フライス盤を使って加工しました。溝加工の一部にかんしては、正直ちゃんとした巨大なフライス盤を使えばよかったですと後悔しています。

4. 塗装

一通り加工が終われば塗装です。正直この作業が一番楽しいですよね。しかし、今回は時間が無かったので雑な塗装の仕方をしています。最初に下塗りとしてプライマーを二、三回塗り、乾いたらデザインに沿ってマスキングしていきます。



図2. キルミーでベイバーなマスキング

この上からまずは黒のラッカーを吹いて、マスキングを剥がし、またマスキングをして黄色のラッカーを吹いてマスキング剥がして、最後にクリア吹いて終わりです。もう少し手順を考えたほうが良かったですね。

5. 組み立て、仕上げ

塗装が乾いたらあとちょっとですね。ピックアップ、ネック、その他のパーツを取り付けて完成！

6. 最後に

最後のほうが滅茶苦茶雑になってしまいましてすみません。というのもですね、この部報を書いている時点ではまだ完成していないですよ。うん、夏休みに家で過ごしていた自分をぶん殴りたいですね。ということで完成品の写真をこの部報に載せるのは厳しいです。すみません。

ただ、調布祭の展示には絶対に間に合うと思う、いや間に合わせるのでぜひ実機を見てもらえる有り難いです。

ギターを実際に作ってみて、その大変さや趣味で作っていらっしゃる人の凄さを感じました。まだまだ稚拙な作品しか作れていませんが、これからも何かギター関連で作れたらなあと思います。

またキルミーでベイバーなアニメも最近色々ありましたが、原作ともども長く続けばと心より願っています。

7. 参考文献

[1]ギター・エフェクターの自作改造
松美庵

<http://www.matsumin.net/>

[2]キルミーベイバー 限りなく公式HPに近いサイト

<http://killmebaby.tv/top.html>

1 導入

とある動画で管楽器にエフェクタを使い音に様々な変化を与えていた様子を見てエフェクタに興味を持ち、どうせならデジタルでやってみようということで手持ちの Atmega328 を使いました。8bit マイコンでどこまでデジタル信号処理を実現できるかあまり検討がつかなかったため、とりあえずディレイやエコーあたりから始めて様子を見て最終的にはピッチ変換を実装してみたいと思っています。今回は AD 変換, DA 変換, ディレイとエコーの実装をしました。

2 マイコンで行うデジタル信号処理の基本

2.1 信号の入力

アナログ信号にデジタルで信号処理をかけるためには信号をアナログからデジタルに変換する AD 変換という作業が必要です。多くのマイコンには AD 変換の機能がついており、部品を追加することなく AD 変換を利用することができます。AD 変換では入力する信号とは別に変換する際の基準となる電圧を用意しなければなりません。今回はその基準電圧として 5V を採用しています。マイコン自体の電源に 5V を使っているので用意しやすいということが理由です。入力された信号はこの基準電圧と比較され、離散的な値（今回は 256 段階）としてマイコンに入力されることになります。製作中の音源として iPod を使用しましたが iPod は出力が大きくて 2V_{pp} 程度ですのでそのままのレベルで AD 変換をかけると 5V ある基準電圧に対して信号が小さく、振幅の情報を十分に変換できないため iPod とマイコンの入力端子の間にアンプを入れて信号の大きさを入力前に調整できるようにしました。

2.2 信号の出力

デジタル信号に処理を加えたら次はアナログの信号を出力するたまに DA 変換を行います。ここでは PWM と RC ローパスフィルタを組み合わせて実装しました。図 1 は PWM 信号と生成されるアナログ信号の対応を表しています。この信号を直接出力のジャックに繋げていますが、オーディオ帯域と

PWM 周波数の帯域は結構近いのでフィルタで信号も結構減衰してしまってそのままの信号をイヤホンで聞くと音量が随分小さいです。

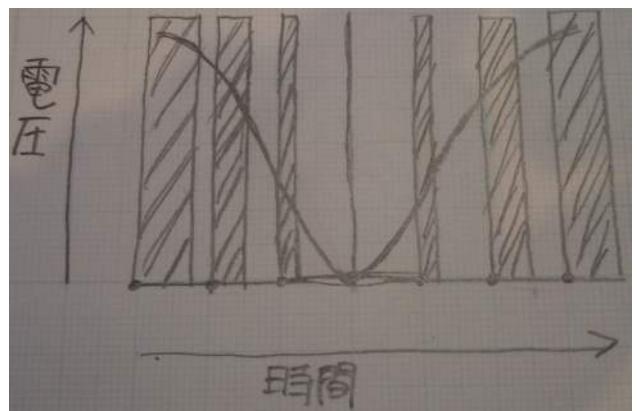


図 1 PWM 出力と波形の関係

この入力と出力をを使ってマイコンに信号を入れた信号をエフェクトを加えずそのまま出力するモノをつくることができます。これだけでも意外と感動します。

3 ディレイ処理

入力と出力ができたのではその間にいろいろ処理を加えてあげればエフェクタになります。まずはディレイです。ディレイといつても実際にすることは入力された信号をマイコンに保存して、今度はその信号を読みだして出力するだけです。その保存した信号を読み出すまでに間を開ければ出力にはディレイがかかることになります。原理は簡単なのですが今回用いたマイコンではそう簡単にはいきません。マイコン内蔵の SRAM が 1kbyte 程度の容量で十分ではないからです。そのため外部に SRAM を追加する必要があります。23LC1024 という外付け SRAM はマイコンとシリアル通信（今回は SPI）で接続することができます。この SRAM は 128kbyte の容量を持っていて、十分な量のディレイをかけることができます。

図 2 では上が入力、下が出力の波形です。入力信号として 1(Hz) の方形波をいれていますが波形は入力も出力も随分様子が違います。これは今回の回路の入力部分にハイパスフィルタを形成されてしまってそのため波形がなっています。とりあ

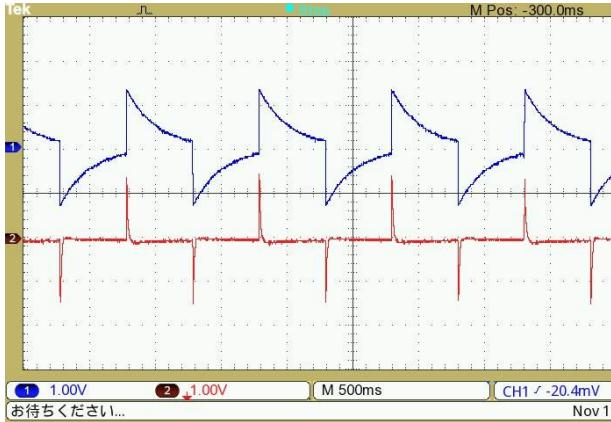


図2 ディレイ無し

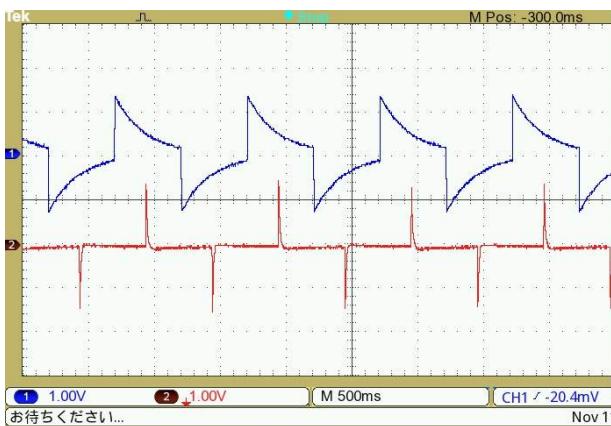


図3 ディレイ有り

えずディレイの様子を伝えるためには問題なしとして使っています。次に図3では出力が入力に対して明らかに遅れていることがわかると思います。これが今回実装できたディレイの様子です。

4 エコー処理

エコーはディレイを応用でディレイとして出力される信号を入力に戻します。この際、戻す信号は元の1倍未満にする必要があります。今回は0.5倍にしました。このエコーはハードウェア的に実装しました。ちなみにこのエコーはリバーブとも言うらしいです。

5 回路について

動作の流れを図4にまとめました。

デジタル信号処理というだけあり、あまりハードウェア部分に複雑な箇所はありません。回路図は最

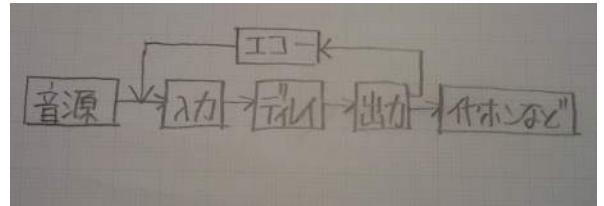


図4 動作の流れ

後に添付します。

6 まとめ

8bitマイコンの限界を知ってみたいと思い始めたこのエフェクタですが実際にやってみると、十分使えるレベルのデジタル信号処理ができるとわかりました。当初の目標であるピッチ変換は今回はまだ実装できていませんが、方針は決まっているのでこれから取り組みたいと思っています。

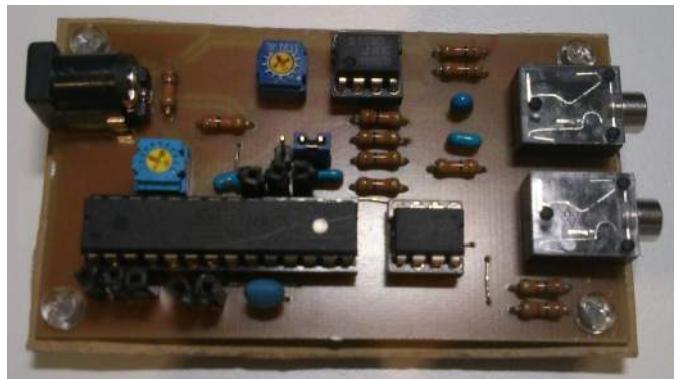
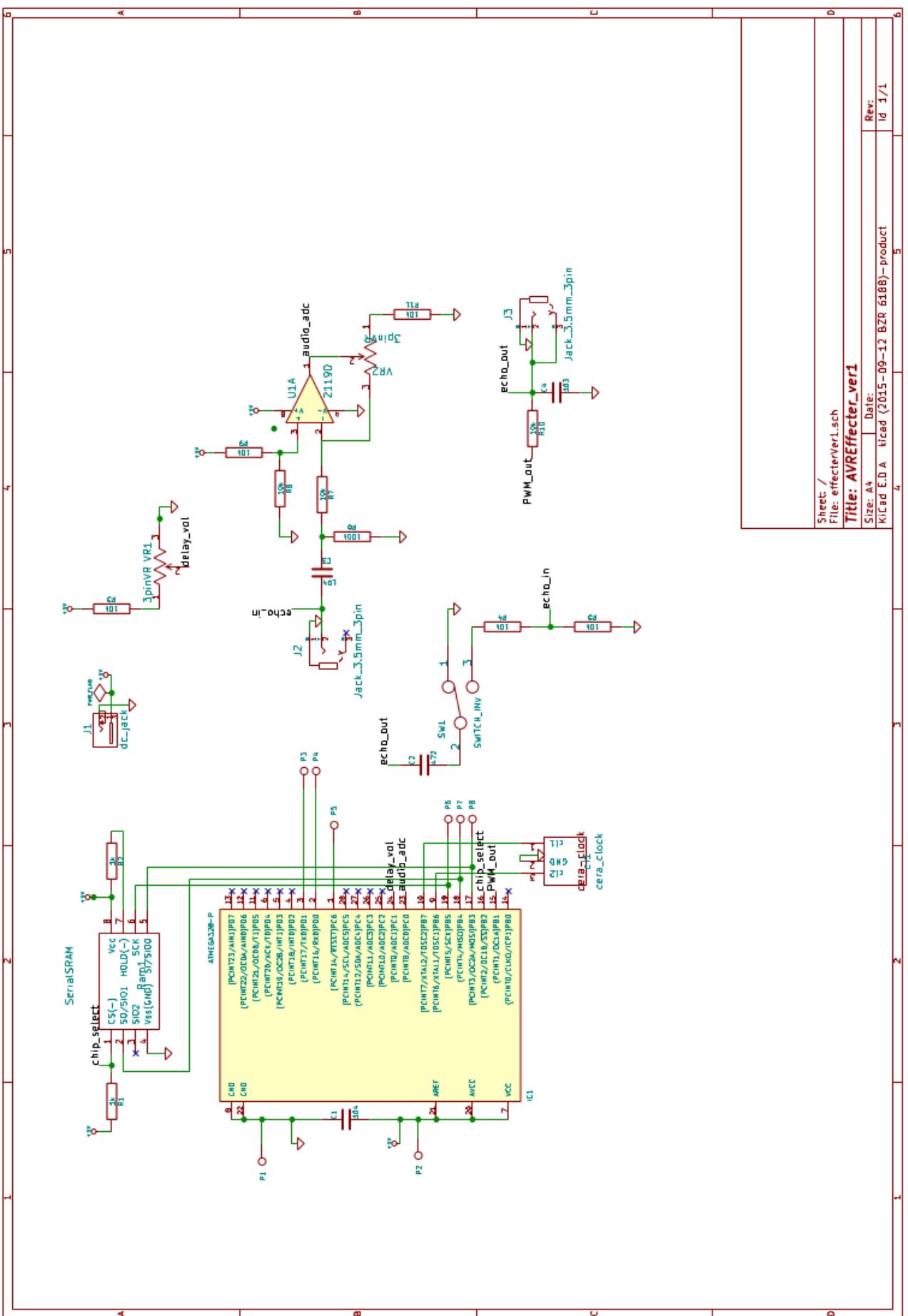


図5 完成品

参考文献

[1] デジタル信号処理の実験

http://elm-chan.org/works/vp/report_j.html



1 はじめに

昨年度、ついに勢い余って Agilent のネットワークアナライザ 8702B(図 1)を購入してしまいました。8702B は、光コンポーネントアナライザですが、ハードウェア的には同世代のマイクロ波ネットワークアナライザ 8753 シリーズと全く同じであり、300k~3GHz までの S パラメータを測定することができます。しかしながら、見ての通り USB はおろかシリアルポートもフロッピーディスクもありませんが、GPIB(General Purpose Interface Bus)と呼ばれる計測器の分野で昔から使われてきたインターフェイスがついています。巷で GPIB-USB 変換器なるものを購入するこどもできますが、需要が少なく 5 万円を超えるものも珍しくありません。そこで、とりあえず作ってみることにしました。



図 1: ネットワークアナライザ (Agilent 8702B)

2 設計

図 2 に制作した GPIB-USB 変換器のブロック図を示します。今回、GPIB の制御には ST マイクロの STM32F103 を、USB インターフェイスには FT2232D を使用しました。FT2232D は、様々な用途に使用できるバスを 2 系統持っています、1 系統を GPIB コマンドの入出力に、もう 1 系統を STM32F103 書き込みのための JTAG インターフェイスとして使用しています。GPIB-USB 間のデータ転送速度はできるだけ早くしたかったので、GPIB 側のバスは 245FIFO

インターフェイスというパラレルインターフェイスで動作させています。

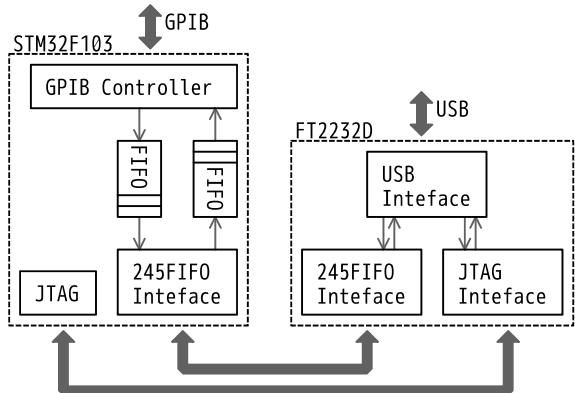


図 2: ブロック図

3 制作

GPIB-USB 変換器の基板パターンと回路図を図 4 および図 3 に示します。回路図で GPIB 入出力ポートにプルアップ抵抗とプルダウン抵抗がついていますが、これは GPIB の規格がオープンコレクタとなっています。STM32F は出力ポートをオープンコレクタ動作させることができますので入出力バッファ回路を省略することができます。

基板は両面基板で設計し FusionPCB に発注しました。価格は送料込み 10 枚で 23 ドルくらいでした。最後に図 5 に完成した GPIB-USB 変換器の外観を示します。変換器を使ってネットワークアナライザから取得した測定例を図 6 に示します。

4 おわりに

本当は、GPIB のインターフェイス仕様や、FT2232 のパソコン側のプログラム等についても記載したかったのですが、全く時間がなかったので書けませんでした。また、今回制作した変換器には GPIB のシリアルポートやパラレルポートといった機能が実装されていません。今後、時間を見つけて実装を進めていくかと思います(現状状態で使えてしまっているのなかなか進みません)。

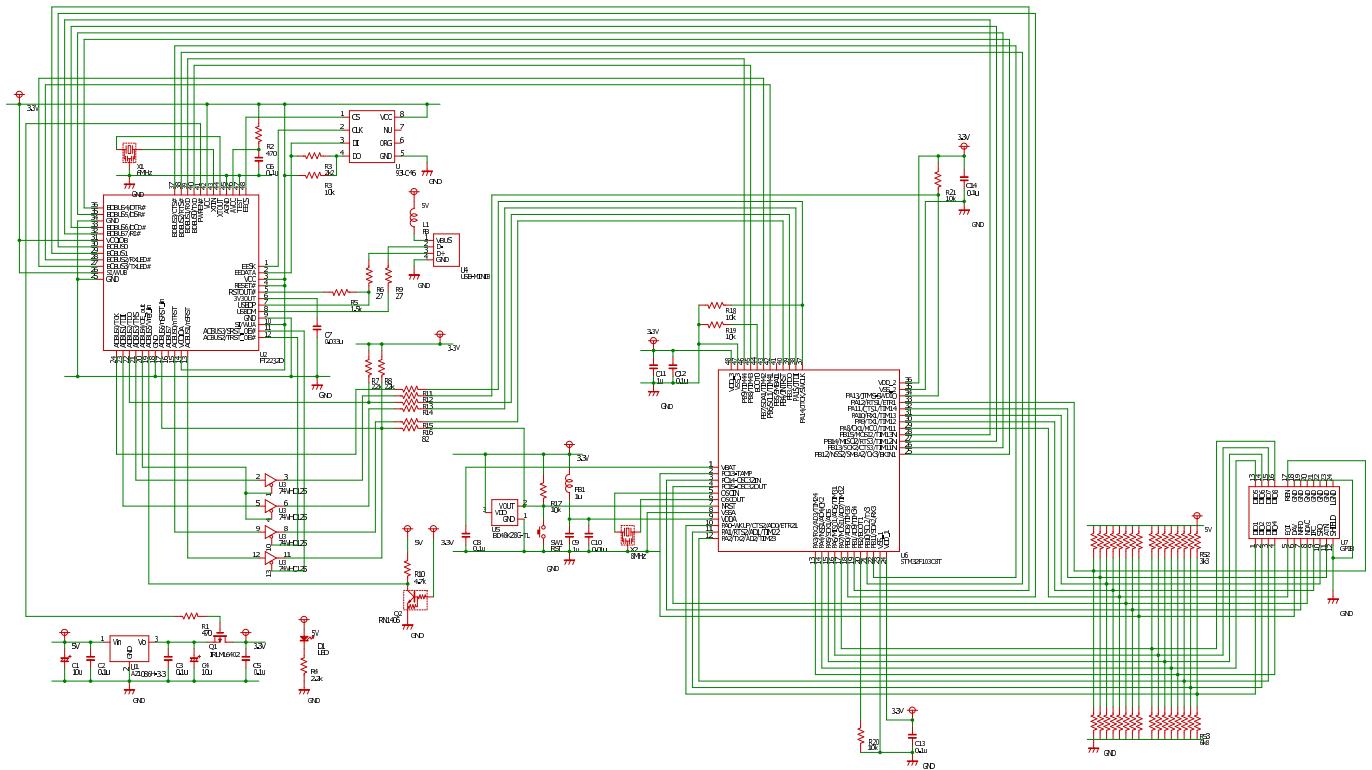


図 3: GPIB-USB 変換器の回路図

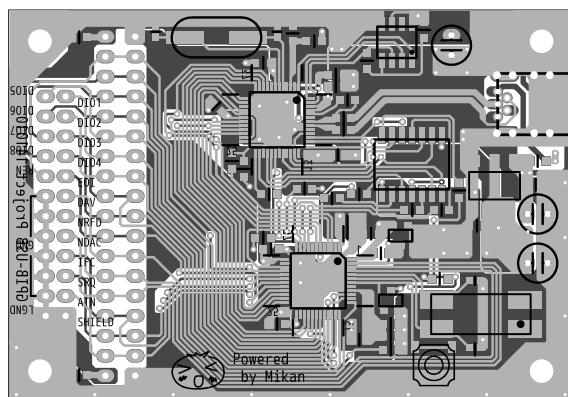


図4 基板パターン

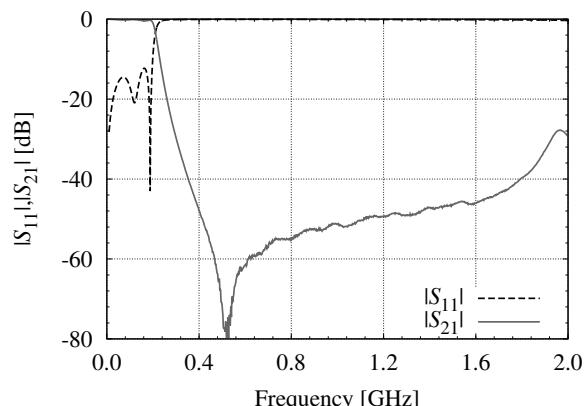


図 6: LPF の測定結果 (おまけ)

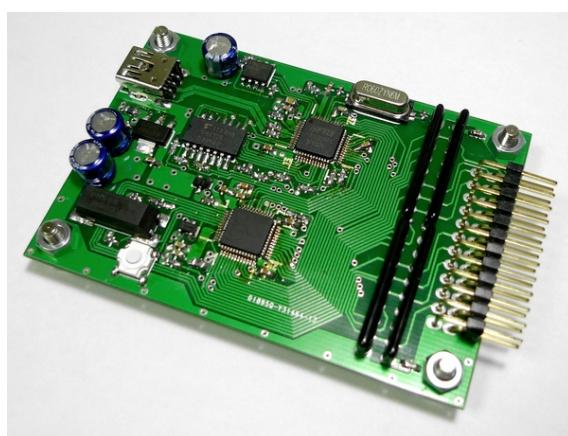


図 5: 完成した GPIB-USB 変換器



図 7: 測定した LPF

工学研究部部報用 TeX テンプレート整備計画

情報・通信工学科 4 年 横田 嶺

2015 年調布祭

1 はじめに

工学研究部部報は作品の紹介や書評、セミナー等の参加レポート、資格取得記、物事に対する考察など部員による自由な書き物として発行している。自分が部長のときの経験からして部員に部報を書くよう喚起してもネタがない、時間がない、面倒だ、などと宣ってまるで書こうとしない人間が一定数存在する。しかし部報を書くことによって自分の作品の目的や工程、完成度などをまとめることができる。また、ネタがない場合は今後作りたいものを部報に書くことでモチベーションが上がるのではないかどうか。他にも自分の生活の中で部報に書けるようなことはたくさんあると思う。

ではなぜ部報を書かないのか。それは部報を書く作業が大変だからではないだろうか。工学研究部部報はその自由さゆえ部報の書き方も自由である。したがって、部報の見栄えに拘るあまり時間がかかるため筆を執りたくないと考える部員もいると思う。

そこで組版ソフト TeX を用いることで見栄えを気にすることなく文章の執筆に専念できるようテンプレートを整備しようという計画がある。テンプレートを整備するメリットとしては執筆者の負担の削減と、部報に統一感をもたせることができ読み易くなることである。しかしテンプレートを整備することで自由がウリの部報であるのに味がなくなってしまうという批判

があるかもしれない。そこで今回は試験的にテンプレートを作成してみることにした。部員の反応によってはさらなる改良が加えられ公式に工学研究部で採用されるかもしれないと考えている。

テンプレートは工学研究部の部報用として使用することを想定し、2段組かつページ番号を振らない設定とした。パッケージは最低限のものをテンプレートに書いておき執筆者のはうで適宜追加して使用するものとする。

この部報そのものがテンプレートかつ簡易的な TeX リファレンスになっており、改変することで部報を楽に執筆することができるのではないかと思う。TeX に不慣れな後輩がたくさん部報を書いてくれることを期待するとともに、普段のレポートなどに活用してくれると幸いである。この部報の.tex ファイルは内部向けに公開する^{*1}予定（予告なしに公開を終了する場合があります）。

^{*1} ※ 2015 年 11 月現在 / 内部/個人/よこた

2 TeX の導入

TeX は組版ソフトであり

1. .tex ファイルの編集
2. platex によるコンパイル
3. dvipdfmx により .dvi ファイルから .pdf ファイルに変換

が作業の流れの基本となる。必要なものとしては

- TeX 本体
- テキストエディタ
- コンパイラー
- pdf ビューア

である。それぞれ自分で揃えることもできるが初心者へのオススメは TeX の統合開発環境を使うことである。

2.1 TeX 本体の入手

TeX Live や W32TeX というソフトをインストールする必要がある。容量が大きいので注意されたし。

2.2 TeX 用統合開発環境の場合

TeX 用統合開発環境として「TeXstudio」、「TeXworks」などがある。TeX に必要なものが揃っており編集からコンパイル、pdf 表示まで一括で行ってくれる。また、コマンドなどの入力補完が強力である。

2.3 コマンドラインの場合

vim や Emacs のような使い慣れたテキストエディタが使いたい場合はコマンドラインなどを使って編集やコンパイルをする。

3 TeX の基本的な使い方

document 環境内に文字を書けば出力される。以下に例を示す。

```
\begin{document}
```

ここに文字を書く

```
\end{document}
```

このように \begin{ 環境名 A } と \end{ 環境名 A } で挟まれてる間はその環境 A となる。また環境 A の中に環境 B を入れ子にすることもできる。

文字を書くだけでインデントや改行は自動で行われる。自分で改行したいときは \\ を入れる。

次の段落へ移りたいときは 1 行空ける。

4 タイトルの書き方

タイトル欄に題名、名前、日付を表示するときは \title{ 題名 } \author{ 名前 } \date{ 日付 } を \begin{document} より上に書き document 環境内で \maketitle とすればタイトル欄が出力される。

5 見出しと箇条書き

見出しあとでは \section{ 見出し } と書くことで見出しができる。箇条書きは itemize 環境や enumerate 環境内で \item AAAA などと書く。

6 数式の書き方

数式環境中でないと数式は使えない。eqnarray 環境や equation 環境などがある。文章中で数式や記号を使いたいときはインライン数式モードを用いると表示できる。

$$\nabla \cdot \mathbf{D} = \rho \quad (1)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (2)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (3)$$

$$\nabla \times \mathbf{H} = j + \frac{\partial \mathbf{D}}{\partial t} \quad (4)$$

表 1 TeX と MS Word の比較

$$F(s) = \int_0^\infty f(t) e^{-st} dt$$

$$X(z) = \sum_{n=-\infty}^{\infty} x_n z^{-n}$$

$$x_n = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz$$

	TeX	MS Word
種類	組版ソフト	文書作成ソフト
編集	テキストベース	グラフィカル
数式	綺麗	汚い
レイアウト	自動	手動
修得	慣れが必要	簡単

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (5)$$

7 図表の挿入

figure 環境を用いる。図の挿入には \includegraphics を使う。できれば図は拡大しても綺麗なベクター形式を使うべきである。

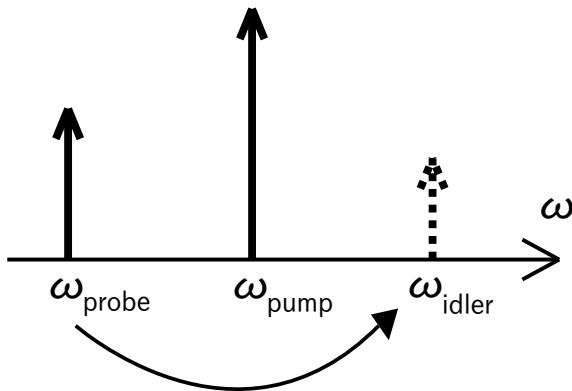


図 1 縮退四波混合の概要

表の挿入では figure 環境内で table 環境を使用する。

8 参考文献の書き方と引用

thebibliography 環境を使用する [1][2].

参考文献

- [1] 「TeXwiki」, <https://oku.edu.mie-u.ac.jp/~okumura/texwiki/>
- [2] 「LaTeX コマンド集」, <http://www.latex-cmd.com/>

国立大学法人 電気通信大学
工学研究部 部報 2015年調布祭号

発行所 国立大学法人 電気通信大学工学研究部
〒182-8585 東京都調布市調布ヶ丘 1-5-1 サークル棟 2階
Email koken@koken.club.uec.ac.jp
URL <http://www.koken.club.uec.ac.jp>

発行責任者 粟島 裕大
編集者 片岡 竜馬
発行日 2015年11月19日
執筆 工学研究部 部員

