

回路図集

2007

麻布学園物理部 25 線班

トランジスタ

ゲルマニウム・珪素などの
半導体の接合を利用して、
増幅・発振・スイッチングの
作用を行う素子。

transfer of signals

through par...torの造語。

エミッタ・ベース・コレクタの
3部からなり、極めて小型
加熱電圧が不要で、
必要電圧が小さい。
シリコントランジスタが発見。

はじめに

『Core Duoの登場でCPUクロックは3GHzを切って下がってます』と去年の回路図集にありましたが、何にせよ『G』という単位を使っている時点で、我々とはあまり関係ない話、というのが現状です。しかし、先日 USB、LAN 接続機能搭載のマイコンが発売され、これによって我々も『G』の世界を垣間見る事が出来るのかもしれないね。

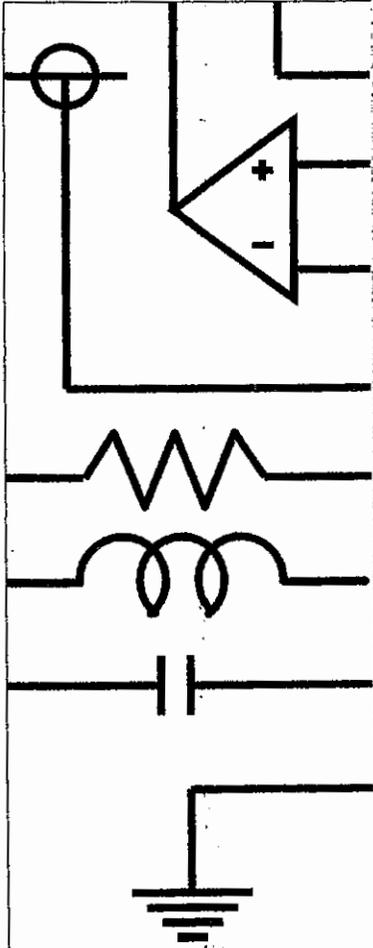
この回路図集は麻布学園物理部無線班の部員が一年間研究、製作したものを記したものです。中には未完成のものも有る為、この本の通りに作ってもうまくいかない場合がありますが、当方は一切責任を負いかねます。あらかじめご了承下さい。

H2 会計:安部 遼馬

目次

-CONTENTS-

目次			p,4
アナログ系			
BUILDING STREET FIGHTER ～兄葉建築士の戦い～	M3	渡邊	p,7
ロボット系			
真昇竜技術 特集 無線班員が明かす飛行技術のコモンセンス	H2 H1 M3	井上 木村 菅原	p,10
プロジェクトX 先行者達 ～物無産2足ロボット「廃人28号」を歩かせる～	H2 H1	安部 佐野	p,26
ARM or...?	H1	大塚	p,30
CLIMBER-2 -A Climbing Addict-	H1	竹島	p,34
Teacher -3- -Last Teacher~	H1	中嶋	p,40
習字ロボット de "write"	M3	西村	p,45
FIERCE FLIGHTER	M3	早川	p,50
ゲーム系			
ヘヴィー級ゲーム	H1	川崎	p,54
エレキサイティングホッケー	M3	村瀬	p,59
アウトバーンレーシング	M2	飯田	p,64
Don't Stand Up Late!	M2	池上	p,67



1 0 4 1 0 5 1 0 6 1 0 7

1 0 4 1 0 5 1 0 6 1 0 7

1 0 4

1 0 5

1 0 6

1 0 7

BUILDING STREET FIGHTER

～兄葉建築士の戦い～

製作者:M2-4 渡邊翔

協力者:物理部の皆様

これは、人の耳の中にある傾きを知る規管、三半規管を電子的に再現しようとしたものです。仕組みとしては、3軸加速度センサからの信号をADコンバータで変換し、それをPICに読ませて角度を計算するというものです。詳しいことは、後々説明していきます。

The prologue

「く……。何てことだ……。」

つぶやく一人の男、兄葉一級建築士。兄葉は、人通りの多い夜の繁華街の真ん中で、まるでキチガイのように四つんばいになっていた。

「この私が……。この私が……。欠陥設計をしたとオツ」

建築履歴15年。今まで一級建築士の中でも、さらにそのトップに立っていたとも言える男、兄葉が欠陥設計など、スキャンダル中のスキャンダルである。

いったいどうしたのだろう？

ついに建築士生命も終わりか？

「えー、またしても先ほど、兄葉一級建築士が設計したとみられるマンションが、耐震強度が足りないことが分かりました。このマンションは……。」

くそ……。

近くの電気屋のテレビから、速報のニュースが聞こえてくる。

ニュースもそればかりだ。

そんなもの、もう聞きたくないかない。

だるい体に鞭を打ち、何とか両足のみで立つ。

くらっ。

まただ。

最近兄葉は立ちくらみに悩まされることが多くなっていた。

別に、髪の毛がすべてなくてかつらではあるが、そんなに年というわけでもない。

これはまさか……。アレが原因か！

唯一兄葉に思い浮かぶ原因はただ一つ。

それは、三半規管の故障である。

三半規管がだめになれば、もちろん傾きなんかは分からなくなる。そうすれば、本来の自身の強さなんてのも分かるわけがなくなる……。

建築士としては致命的な欠陥が一ヴェートーヴェンが耳が聞こえなかったみたいに――この男に生まれてしまったのである。

これでは本当に、建築士生命はおしまいだ。

どうする……。

そこでこの単細胞、考えていることは建築のみという男は打開策を見出す。

そうか、作れば……。

三半規管、それは人体の神秘である。

しかしながら、最近の科学は人をも凌駕しようとしてるほど、発展してきた。

よし、これならいけるかもしれない！

早速兄葉は作業に取り掛かった……。

About Sanhan kikan

さて、三半規管についてなのですが、まずは加速度センサーから説明します。加速度とは、単位時間当たりの速度の変化率のことです。速度がベクトルなので、加速度も同様にベクトルとなります。加速度はベクトルとして平行四辺形の法則で合成や分解ができます。ここで加速度を測り、PIC に読ませるやり方は次の図の通りです。

図 1

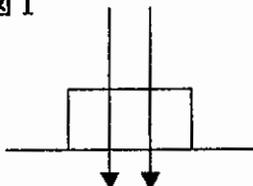


図 2

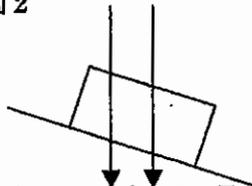
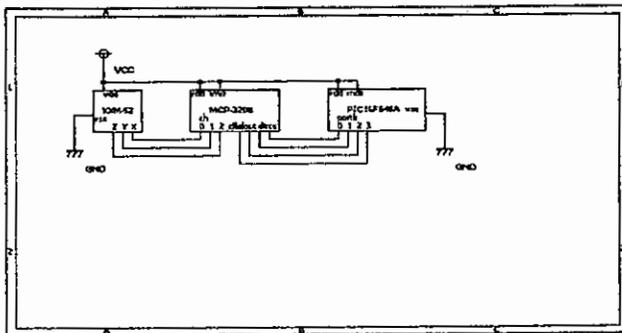


図 1, 2 の四角い物体は加速度センサです。下の長い線は取り付けしている基盤とも思ってください。そしてこの矢印は、重力加速度というものです。地球の地表付近では、どんな物体でも地面の方向への力(重力)を受けており、その大きさはその物体の質量に(ほぼ)比例している。この比例定数を重力加速度(じゅうりょくかそくど)と呼び、これは自由落下する場合の加速度でもあります。

まず図 1 なのですが、これは加速度センサを平らな地面に置いた状態です。このとき加速度センサは 9.8m/s^2 の重力加速度を受けており、それを AD コンバータに送っています。

それを、図 2 のように傾けたらどうなるでしょうか。当然加速度センサには斜めの加速度がかかるわけですから、垂直の力は減ります。すると AD コンバータに出力される値も減り、AD コンバータから PIC に送られる数値も減るわけです。この違いをプログラムで読み取り、角度を判断しようというものです。

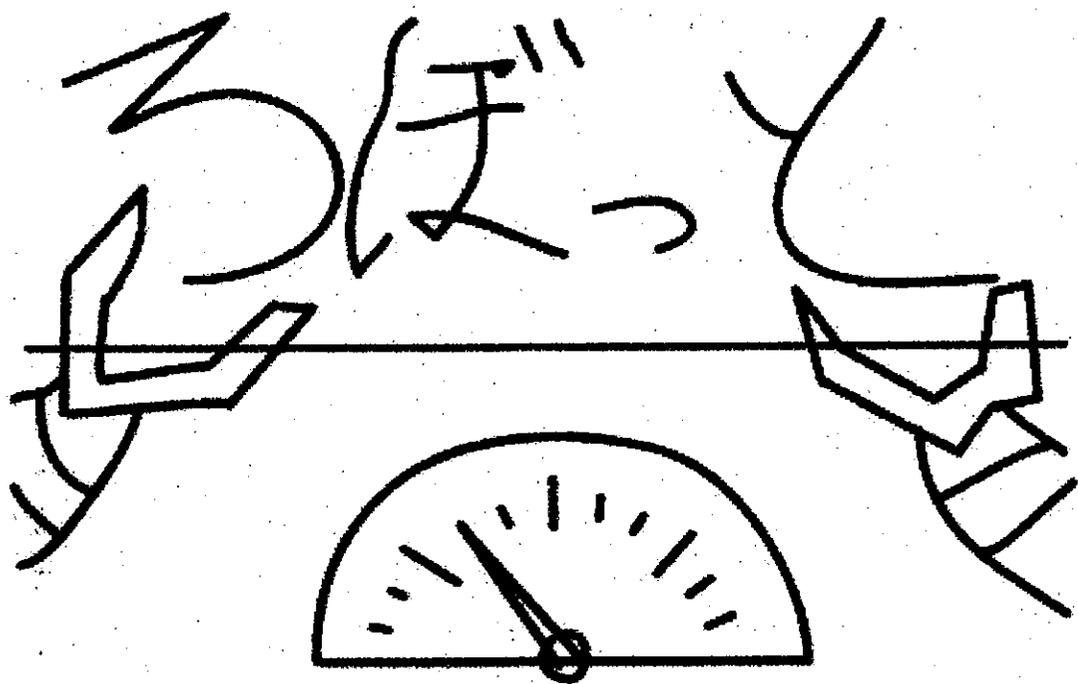
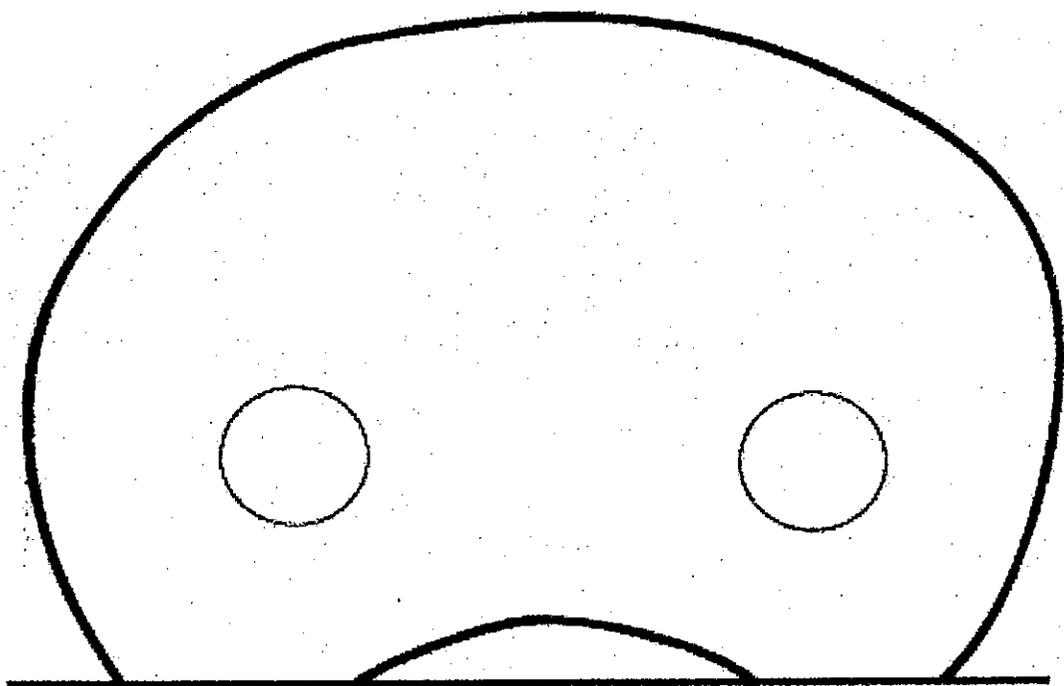
The Circuit



The Last

最後になのですが今現在(3月31日)完動していません。まあ、なかなか挑む時点で結構な難題であったのですがここまできついとは思いませんでした。状況としては AD コンバータで変換まで入ったのですが、なかなかうまく PIC がよんでくれないということです。タイミングが悪いのか、それとも単に遅いのか……。まったく原因不明です。なかなかこの時期にその時点ということは、完動も難しいでしょう。今の状態としては、新たに出来た売り物のほうに力を入れているという次第です。なかなかこれがないと Aerosmith のほうで困ったりと結構まずそうなので……。まあ、出来ればどこかでポーンと悟って、先へ進みたいものです。

* 注意:この回路どおりに作っても動かない可能性がございます。あらかじめご了承下さい。



役に立つかも？エレクトロニクスの総合誌

2007年3月1日発行(新年1期1日発行) ついでに卒業生に贈られて電卓？それとも只の電卓？

真昇電技術

2007

5

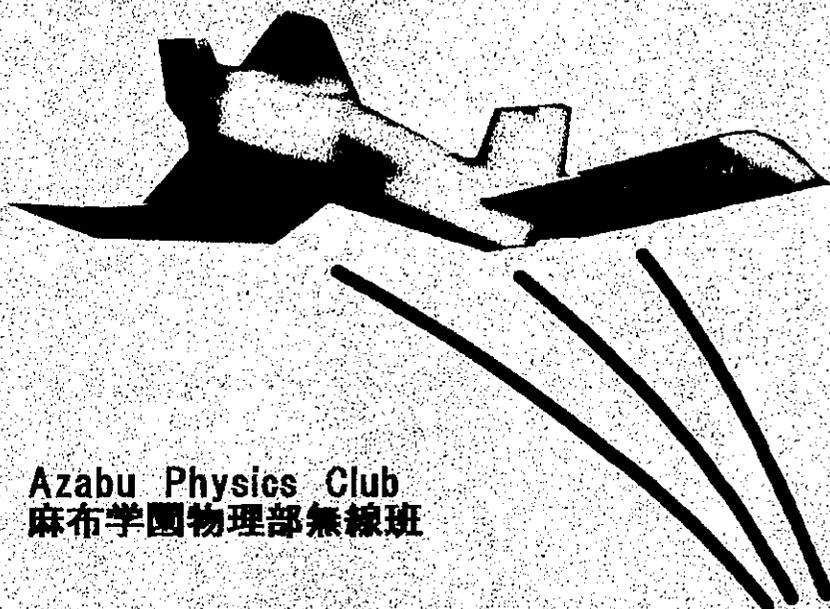
特集 無線班員が明かす飛行技術のコモンセンス

<新連載> FMワイヤレス通信のススメ

◆ 新説 サーボモータの制御

こんな自立飛行ロボほしかった

ASXXX



Azabu Physics Club
麻布学園物理部無線班

バックナンバーのご案内

このページでは現在のところ、2007年1月号以降の目次を月号別に掲載しています。

2007年1月号以前の内容につきましては、過去の回路図集を参考にしてください。

なお2007年1月号以降は、記事の一部を紹介致します。

2007年

2007年1月号 特集:実験研究!屋根への誘い

第一に作り始めた機体。製作員達にとっては小手調べというところだろうか。試行錯誤を重ね(というか2年前を大分パクリ)作り上げる。感動の嵐の中ついに初飛行へ!…結果は見るも無惨に体育館の屋根へ不時着(4階非常階段からのフライト)。救助を待つも世知辛い世の中のため、風に煽られ地面へ落下。本体大破により絶版に。

2007年2月号 特集:すぐ見つかる!校庭の木の上に。

第二に作り始めた機体。1月号を手本として、ほぼ同じ形で作る。さすがに同じものだからか1月号よりもはるかに高性能な機体ができあがり試した時にはプールのネットまで余裕で届いた。しかし、プラズマを吐いていた製作員が復帰した時に調子に乗ってもう一度テストしたところ、暴風の影響で機体が流され校庭の木に衝突。機体がひっかかり何をしても落ちなくなる。結果2ヶ月間救助できず、暴風雨で落ちた際に奇跡的に生還したのはサーボモーターだけだった。臭いので絶版に。

2007年3月号 特集:サーボモーターで体験する操縦可能な世界

第三に作り始めた機体。2月号が木に引っかかったため作り直した機体。趣旨に沿って翼を逆ガル翼に、本体の長さを小さくなどの改良を加える。現在は2月号ほどは飛ばないものの、頑丈さからテスト機として使われる。好評発売中。

2007年4月号 特集:無線班員が明かす飛行技術のコモンセンス

今月号。詳細は内容を読んでもらいたい。好評発売中。

*~月号は~号機と変換してお読みください。ただの機体紹介です。

無線班員が明かす飛行技術のコモンセンス

製作者：H2 井上 忠明, H1 木村 朴, M3 菅原 寛生

協力：物理部無線班の皆様

ここでは、本体製作に関する技術を主に取り扱っていく。

■本体

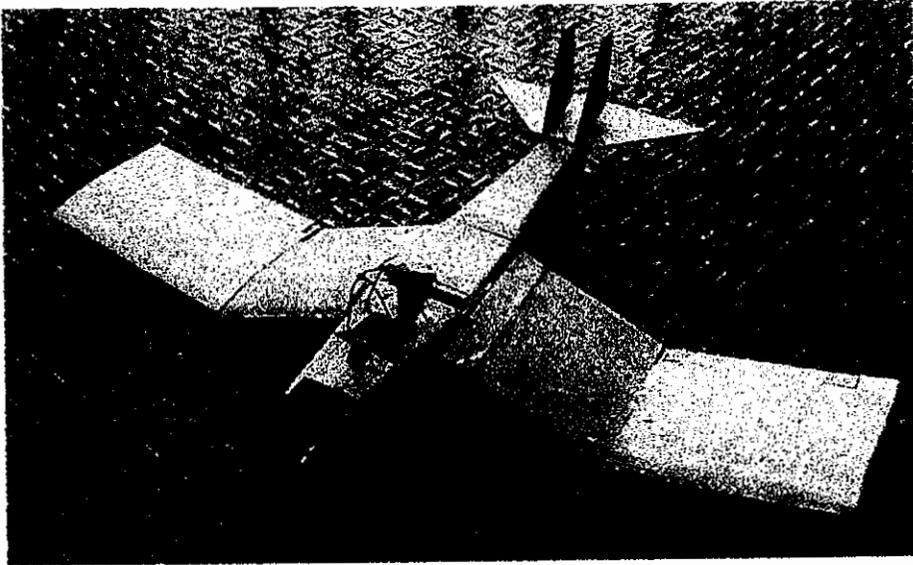
●はじめに

まず、あなたは飛行機という言葉聞いて何を思い浮かべたでしょうか？おそらく人によって千差万別、共通する例えばゼロ戦のようなものもあれば、マイナーなそれこそ人前で言えば引かれるだけのようなものもあるだろう。この製作物は名前からして飛行機なのだが、具体的に何を目標にしているかということを書きおこななければ多少疑問に思われるところがあるかもしれないので先に述懐しておく。

この製作物の目指すところは漫画「ジョジョの奇妙な冒険」の第5部に出てくるスタンド、「エアロスミス」の完全再現を目指したものである。ただ、この「完全再現」という言葉が指すものは形ではなく、機能である。しかし、形に関しても単行本を読みながら寸法を測りある程度近づけようとはしている。何故微妙に異なるかということ、やはり元が漫画であるからと面倒臭いからである。

●実物

口で語っても百聞は一見にしかずというので、まずは本体の画像をのせてみたい。



以下に搭載する機能等に関しての解説を入れたい。

フラップペロン…羽の両端に付いているもの。それぞれを上下させることで旋回、上昇下降を行う。

エレベーター…後部の水平尾翼を上下させることで、機体を上下させるもの。

ラダー…垂直尾翼を左右に動かすことで、機体を旋回させるためのもの。安定な旋回のため。

コクピット…無用の長物。空気抵抗になる、とプラズマの人は怒る。エアロスミスに近づけるため。

搭載させたかったもの

銃…エアロスミスには必要。ただ、毎秒数十発打つのは無理。

魚雷…一発ならできるが、本物は自然復活らしいので…。

CO2 レーダー…温度センサーで代用できなくはないが…。

■機体製作方法

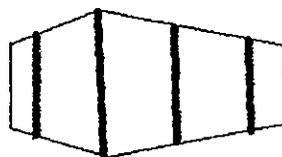
以下に簡単な機体製作の方法を記してみたい。部員向けの話だが、無知の人は2年前のものも参考にさせていただきたい。また、あくまでこれは自分達が行ったものなので参考程度に。

●本体製作方法

まず、本体の大体の外観を決め、そこから下地となる板を切り出す。製作は加工を考えてバルサで行うため、ある程度の厚み・強度を持ったものを使う必要がある。また、この工程で回路を入れるスペースなども考えておく。

次に、先ほど切り出した板に垂直にバルサを付ける。このバルサの横幅は下地の板に貼り付ける位置いっぱいの幅、縦幅は作ろうとする機体の厚み分のものとする。これは、横の壁を付けるために必要となるのである程度の数を付ける。(図1の太線部分参照) この際、機体の形の折り目となる部分には付けなければいけない。楊枝などを下から挿して強化しておくといだらう。

図1



そして、その後で下地の周りに横壁を貼っていく。貼り方は、下地と工程2で貼った垂直なバルサの側面に接着剤を塗り、前か後ろから順々に貼っていく。この際に、洗濯バサミなどで圧着することで下地通りの形となる。また、ここで工程2が生きてくることとなる。

最後に上蓋を付ける。これは、工程3で出来上がったものの上に接着剤で、下地よりも大きなサイズのバルサを貼り、乾いた後で削って下地と同じ形にしていく。工程1で決めた回路を入れる部分は、工程4が終わった後に設計図にそって上蓋を切り取ってあける。

●羽製作方法

こちらは単純に作っている。まず、羽の形を計算しておおまかに決め、それにそって型紙を作る。次にそれを用

図2



いて厚めのバルサを羽の形に何枚か切りだしていく。その際、予め容易したヒノキ棒のサイズの穴をバルサの下方に前後左右 2 つ開けておく。そして、それをヒノキ棒（細めで可。図 2 の細線）の上に決めた個数だけ先に用意したバルサを差し込んでいく（図 2 の太線）。最後にバルサとヒノキの結合部を接着し、できたものの表裏に薄めのバルサシートを貼って、それをテープ（今回使用したのは透明な梱包用テープ）で覆ってコーティングする。今回は逆ガル翼なのでこれを 4 個作って、2 つを角度をつけてテープで止めて 1 つの翼とした。

■レーダー

●使用理由

原作を確認していただければ分かるが、エアロスミスは本体のナランチャが遠隔操作するスタンドである。当然ナランチャの手にリモコンなどは無い。よって原作を再現するのならばこの飛行方法も再現する必要があり、個人の意思による遠隔操作は技術的に無理があるので自動操縦をさせることにした。そのため、壁までの距離などを測って曲がるためにレーダーをつけた。また、今回は一応リモコンも製作しており、これは実験用と自動操縦暴走防止用である。

●仕組み

とはいえ、これも簡単なものであり、超音波センサー（これに関しては同じものを H1 中嶋が使用しているのでそちらを参照していただきたい。）をサーボモーターの首振りによって回し、サーボが首振りの限界点までいったら距離を測って 3 メートル以下だったらセンサーが向いている方向とは逆に向かうようにフラップロンなどを動かすことを繰り返しているだけである。実際、超音波センサーが不安定であり、また飛行速度が速いのであまり正確では無い。

FM 無線通信技術

さて、ここではこの自立型飛行ロボットの緊急時制御用に製作した FM 無線通信—所謂ラジコン—を紹介しよう

■ FM ラヂヲについて

● 電波とは

今日、私たちは家電量販店や雑貨店等へ赴き、製品のラジヲを気軽に買うことができる。そうして家に帰って電池を入れるとすぐにラジヲ放送を楽しむことができる。

子供心にこれは魔法なのではないか??と、思ったことは残念ながらないが、確かに不思議な現象である。全く、人間は目に見えない現象を不思議であると思いがちである。そう、ラジヲ放送に用いられている電波は人間の目には見えない光なのである。

そもそも電波とは如何なるものであろうか？

皆さんは電流を流している電線の近くに磁石を近づけたらどうなるか知っていますか？はい、右ネジの法則によって方位磁針は正しい方位を示しませんね。それはもちろん電流によって磁界が発生しているからである。また導体によって繋がっていかなくとも磁界は発生する。それは例えばコンデンサなどに電圧をかけた場合、変異電流によって電界が発生する為である。

同時に、磁界は電界を発生することができる。即ち、電流によって生まれた磁界が新しい電界を形成し、その電界がまた新しい磁界を形成するのである。この様にして電磁力によって放射するものが電波と呼ばれている。

●AM ラジオ FM ラジオ

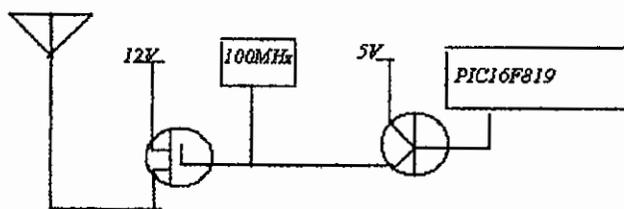
皆さんは AM、FM の違いを知っているだろうか？中波帯か VHF かの周波数の違いと言う事なかれ。この二つはそもそもの変調形式が異なっているのである—いや、見たまんまなのであるが—AM とは振幅変調。信号波と搬送波を混ぜることによって、同じ周波数であるが、振幅が異なる電波に変調するというもので、中波帯を使うことによって、FM と比してより遠くに電波を送ることができる点がメリットである。FM とは周波数変調。信号波によって搬送波自体の周波数を変化させる事によって同じ大きさの振幅であるが、異なる周波数の電波に変調するというもので、より確実に高音質を手に入れることが出来る点がメリットである。

今回私たちは FM ラジオによる無線通信を採用した。

■回路

●送信側回路

今回の送信側回路は手軽に高周波を手に入れるため、100MHz の水晶発振子に、トランジスタで増幅した PIC の発信出力を混合することによって FM 変調を作った。しかし、受信側では思うように音量が上がらず、苦労した。

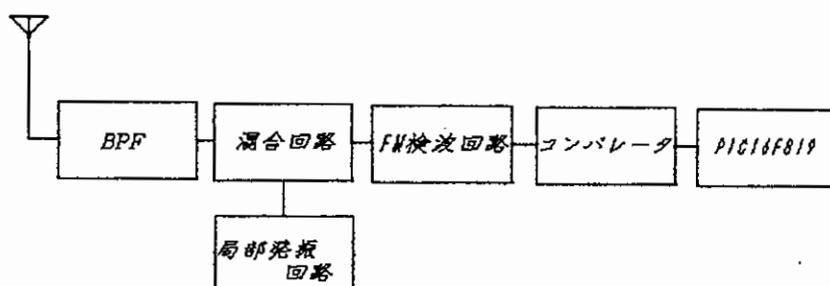


そもそも私はアナログ回路が得意でない為、この様な頭の悪い回路のほうがシンプルで解り易かったのである。

また PIC からは 100kHz の発信を出すか出さないかによって、アンテナからの電波の周波数を 100MHz と 99.9MHz に FM 変調している。

●受信側回路

一般的な FM ラジオを製作し、そこから出てきた信号を OP-AMP によるコンパレータによって増幅し、PIC で読み取っている。私が製作したラジオより、市販のガイドブックに記載されているラジオの製作例のほうが恐らくは優れているであろうため、ブロック図によって割愛とさせていただきます。



重ね重ね言うがアナログ回路は苦手なのである。皆さんには、幾らデジタル回路主流の時代になったからとはいえアナログ回路も勉強することをお勧めする。

■プログラム

●通信プログラム

送信側プログラムは、発振を混合させるか、させないかによってH、Lをつくり、コードとして送信している。また、受信側プログラムは全ての音の振幅を拾えるわけではないので—またノイズを拾ってしまう恐れもあるため—一定期間を設け、何回以上拾うことができればH、それ以下ならLということでコードを読み取っている。私は過去赤外線通信プログラムを手がけていたが、しかし、現在(3月30日)思うように同期できず、未だ改良の余地があるといえよう。



I
Can
Fly!!

(I AM FALLING DOWN.....)

サーボモーター制御 の新理論

要はサーボモーターによる舵の角度制御及びモーターの回転数制御を行う物である。サーボモーターとは0度から180度までの角度に軸を動かすことのできるモーターである。

■サーボモーターの角度制御

簡単に説明すると750usec~2250usecの間1を出力しそれから0を1を出し始めてから20msec出し続けるということを繰り返すものだ。私はこの信号を出すのに750usecのWAITをつくり其れを2~3回繰り返しTMR0割り込みで周期を作っている。詳細は三年前のTITANをごらんあれ。また現在PORTAの0~3bitまで舵用のサーボ制御用のピンで木村さんのPICからの信号で角度を決定し、PORTAの7bitは井上さんのレーダー用サーボとして1秒ごとに0度180度と回転を繰り返している。そしてPORTBの0bitをレーダーPICへの同期信号出力となっている。

(超音波式レーダーについては井上さんのページをご覧ください)

■PWMによるモーター制御

PWM(パルス変調)とは上のサーボモーターへの信号の応用で周期とデューティの比率でモーター出力を制御するものである。つまりデューティ;周期=1;1の時にモーター出力がmaxになり0;1の時にminになるというものである。私は0のみと1、0を1;1の割合で出力する定周期のパルス、そして'1'の3種類の信号を出すことで総合的なエネルギーの全体量を0%、50%、100%の3段階で制御している。今はプログラムによって制御し


```
BCF PORTA,6
WARIKOME2
DECFSZ 21H
RETFIE
DECFSZ 22H
GOTO $+7
MOVLW D'3'
```

```
MOVWF 20H
MOVLW D'50'
MOVWF 21H
BSF PORTB,0
RETFIE
MOVLW D'1'
MOVWF 20H
```

```
MOVLW D'50'
MOVWF 21H
MOVLW D'1'
MOVWF 22H
BSF PORTB,0
RETFIE
END;ENDRI
```

* \$+(-)n とはプログラムカウンタ+(-)n 行後に飛ぶことを示す。

●プログラムの解説

処理をどのように進めているかというと

- 1、モーターONorOFF
- 2、ラダーサーボ制御(0度 or90度 or180度)
- 3、エレベーターサーボ制御(0度 or90度 or180度)
- 4、モーター50%orMAX
- 5、フラッペロン1制御(0度 or90度 or180度)
- 6、フラッペロン2制御(0度 or90度 or180度)
- 7、入力信号確認(割り込みがあるまでずっと)
- 8、20msecのTMR0割り込み

1からの繰り返し

というものだ。4月現在完動したものなので応用するなりして使うといいでしょう

■PIC16F819

いままではPIC16F648Aを使っていたが、より高機能なPIC16F819を使っている。今までとの変更点及び注意点は、いまままでPORTAの4~5bitPORTBの2~7bitを使って入力をしていたが、SPIを使ってシリアル通信をするので入出力が3ピン(Din、Dout、CK)で済む。よって今まで使っていた入出力を他にまわすことができる。

- ・出力を2本使えばチップセレクトによりSPIを使い簡単に大容量データ(256bit)ADコンバータやPICとの通信を行える。
- ・10bitのADコンバータを内蔵している。よってアナログセンサーを使うことができる。
- ・内部クロックが4MHZだったのが8MHZに上がりより多くの命令を実行できる。

回路のモータードライバ部分には MOSFET の 2SK2782 を 2 個並列にして使っているが便宜上 1 個になっている。また電圧の変更およびノイズ防止のため PC817 というフォトカプラーを使っている。また、819 版はまだ詳細がきまっていないので後々追加する予定。

■感想

H2 井上

ついに最後の製作物の製作が終了しようとしています。結局中一の時以外、3 年連続で共同制作となりました。まあ何というか共同制作も個人制作もいい面もあり、悪い面もあり、という感じですね。僕の経験的に共同制作者同士というのは、製作物というより個別パートごとにはあまり干渉し合わない方がいいかなあという気はします。まあ、どうしてかはやってみれば分かると思いますが、僕はそんな気がします、というだけで。

まあ、しかし共同制作はやはり面白くはありますね。製作物に関して一緒に考えたり、悩んだり、暴れたり。そんなことは一人の活動では味わえないことでしょう。今にして思えばあの木に引っかかったときに 7M の棒を振り回して救出しようとしたこともいい思い出だと思います。

製作物にエアロスミスなんて掲げて、普通の人なら笑い飛ばすところを必死こいて作っている、という光景。やはりいいものですね。夢を追いつづける姿勢は人間に必要なものでしょう。しかし、このままでは第二製作物として構想していたツェッペリン号の製作まで行けなかったのが残念ですが。何にせよ菅原君はこれから飛行機を作りつづけるといっているなのでこの文化は続いていくことでしょう。最早飛行機はひとつの文化として確立されたようですね。いつしか本物に肉薄したエアロスミス、ツェッペリン号、その他夢のある飛行機が大空を飛ぶことを祈っています。

H1 木村

今回の無線製作はかなりの手間を掛けてしまった。しかし、私は未だ、まるで意味もなく大量についているコンデンサや抵抗の意図をつかめずに居る。今回のことを反省し、もう一度アナログ回路製作について勉強しなおそうと思う。

M3 菅原

いや難しかったです。はじめは「サーボモーターなんか1ヶ月で動かしてやるぜ!!」と息巻いていたにも関わらず、サーボモーター1個を任意の角度に固定するのに1~2ヶ月もかかり、動かすのに2ヶ月かかりで……。ここから4個のサーボを動かすにはさほどの時間はかからなかったのですがモーターを動かしたりアンプが動かなかったりとかで学年末後の休みでやっと完動しました。しかしこれからもSPIを使ったりとか慣性誘導装置を完成させたりとかやるのが沢山あるので最後まで精一杯やっています。

それにしてもプログラムとは回路に比べたら楽なものです。僕の回路はPIC1個と三端子レギュレーターのみの非常に回路も簡単なものになりモータードライバー以外は比較的楽に作れました。またプログラムに関しても他への応用の広いプログラムに仕上がったと思います。ただ、部品面に関していくら部品は来年も使うことのできるものだといってもいくらか殺したりして金をかけすぎてしまったので来年はこの分節約します。なお、財政難なのにお金を工面してくださった会計さん有難うございました。



プロジェクトX 先行者達

～物無産2足ロボット「廃人28号」を歩かせろ～

H2 安部 遼馬 H1 佐野 祐士

前半部分はH2 安部がお送りします。

・概要

簡潔に言えば廃人28号は2足歩行ロボットです。というかそれ以外の何者でもないです。

・事の発端

平成17年。物理部は合宿を終え、バレーボールとトランプの時期を謳歌していた。だが、一人浮かぬ顔をしている男が居た。安部 遼馬、代替わりで会計になって間もなかった。彼はある一つの不安を抱えていた。

結局昨年の象型ロボット*は失敗し、先輩の2足歩行型ロボットも歩く事はなかった。それら2つの失敗の共通の要因は、致命的な電力不足であった。当時の物理部に配備されていた電源は20W程度。中1のゲームを動かすのには十分でも、モーターをいくつか繋ぐと、悲鳴を上げた。

だが、その夏のハムフェア**に、転機が訪れた。安部がなにを考えるでもなく歩いていると、一人の後輩が叫び声を上げた。

「安部さん、見てください。60Wの電源が1500円で売ってます。」
彼らは歓喜の声を上げながらも、値切り交渉をして1000円でその電源を買った。「これなら行ける。」安部はそう思った。だが、彼は金属加工と重心移動の計算以外、回路設計もプログラミングも出来なかった。だが、共同製作者はなかなか見つからなかった。安部は、早くも行き詰った。

* 去年の私の製作物。重心移動をしながらの歩行を目指したが撃沈。

** 無線オタクの祭典。私のように知識が無い人にはよく分からない世界。

・プロジェクト始動へ

8月末、安部は相変わらず回路、プログラミングを担当する共同製作者が見つからず、悩んでいた。この際一人でやって、サーボモーターをアルミで繋げて形だけ作って誤魔化してしまおうか*、そんな事まで考えていた。だが、そんな訳にもいかず、「2足なんて無理なのか・・・。」と、思わず呟いた。

その時、一人の男が現れた。佐野 祐士。2足歩行ロボット開発を夢見るM3だった。

「2足をやるなら、僕にやらせて下さい」佐野がそう言うと、安部は「ああ、良いよ。」と即答した。佐野は回路設計、プログラミングもこなせ、安部に不満が有るはずもなかった。9月には企画書**も提出され、ついに、プロジェクトは動き出した。

* ロボットってのは、形さえ出来ればとりあえず文化祭で展示出来ちゃうんですよ。(子どもたちが「これ動かないのー?」とか言ってきて萎えますが)

** 総務さんに提出する設計図みたいなもの。

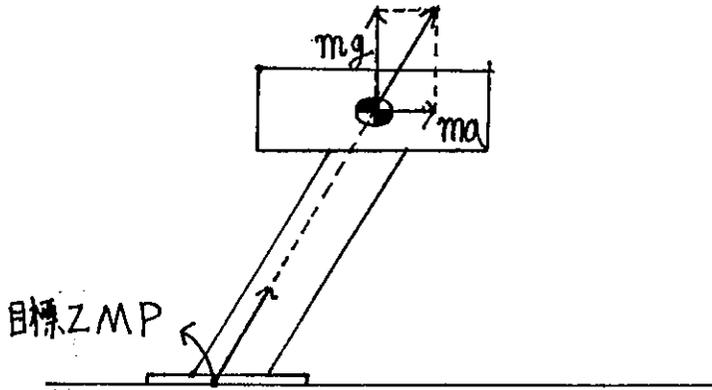
・ZMPについて

あー、もう疲れた。こっからは普通に書くぞ。つーかもう物語じゃないしね。去年は正直よく分からなかったから煙に巻いた(爆)ZMPの話をしちゃいます。今はもう分かってる・・・はずだー。

ちなみに、重心とかモーメントとかニュートン求める式とかが出てくるので、それが分からない人は、物理の参考書を読んでからこの文章を読んでね。

まず、ZMPって言葉には2つ意味があります。ただ単に「ZMP」と言った場合に意味するところは、足裏全体にかかっている床反力の法線成分を、ある一点にかかっているとして置き換えたときの作用点のことです。もっと分かり易く言うと、ロボットが足を床につけている時、足は床からの反力を受けています。ですが、足の裏は面であり、つまりは無数の点に少しずつ反力を受けていることになるのです。しかし、そのままでは具体的な数字を使ってデータを扱う事が出来ないのです。床反力を足裏のある一点で全て受けていると仮定するのです。それがZMP。

もう一つの意味は「目標ZMP」という意味です。これは本来「目標ZMP」と言うべきですが、しばしば「ZMP」と省略されるので注意して下さい。これは何かというと、ロボットにある行動をさせる時、ZMPがそこにあれば、その動きが可能となる、という点です。少々分かり難いので図で説明します。まずロボットの質量を m とし、加速度 a で右に移動したいとします。すると右向きにかかる力は ma 。また、地球上には重力が有り、下向きに g の加速度がかかっているため、ロボットが上下には動かない場合、上向きに mg の力がかかります。そうした場合この2つの力を合成し、その力の延長上で床と交わる点が、「目標ZMP」となります。もし、同じ方向、同じ量の床反力を受けても、延長線上以外の点で力を受けると、モーメントが発生するためロボットはバランスを崩し、最悪転倒してしまいます。



そして、この目標ZMPが支持多角形内に収まらない場合、その動きは実際には不可能、ということになります。

この目標ZMPが分かれば、ロボットの姿勢を調節したりして、ZMPをその目標地点に持っていけば良い訳です。ですが、肝心のZMPを算出するには圧力センサという高価な部品を何個か買わねばならない上、その算出をマイコン上で行い処理するというのは、我々の技術的にはギリギリ可能ではあっても、かなり難しいというのが現状なのです。その為、今回は目標ZMPのみをあらかじめ算出しておき、支持多角形はかなり大きく、重心は可能な限り下げるという方法で、バランスは崩しながらも、どうにか転倒はせずに歩くことを目標としました。何故こんな事が可能かと言うと、実は、目標ZMPが支持多角形内に入っさえいれば、目標ZMPとZMPのズレによるモーメントが、バランスを保つ方向に働くのです。が、そのモーメントによるバランスの立て直しにはタイムラグがる上、目標ZMP算出時には計算には入っていなかった力が働くわけで、あまりに目標ZMPとZMP間の距離が長いと、バランスを立て直す前に、ロボットは転倒してしまうのです。

・感想

僕は金属加工及び本体製作、重心云々を担当していた訳ですが、今期の製作は、途中で電動糸ノコをブッ壊して、ガン萎えしながら修理して、直ったと思ったら、今度は後輩に壊され、結局僕が修理したりと波乱万丈でした。重心関連については、人生で始めて物理の授業内容が実生活に役立ったと思えましたね。佐久間先生有難うございます。(笑)あと、結局回路部分等は全て佐野君に丸投げで終わってしまったようで・・・。ごめんよ佐野。回路もプログラムも出来ないから、君が居てくれて助かった。

ここから先はH1佐野がお送りします。

・プログラム要旨

歩く→加速度センサの入力により姿勢が崩れていないか確認する→歩く→……

なんて簡単なプログラム(殴

※プログラムは長すぎるため掲載されていません。知りたい方は製作者にお尋ねください。

ですが今のところデモプログラムしか作られておりません。

・仕様

重量 2kg

身長 40cm

自由度

脚部 6自由度×2=12自由度

片足あたり ロール×2

ピッチ×3

ヨー ×1

3軸加速度センサ搭載(重心を計算して自分を倒れないように自動制御するためのもの)

できること 坂道を縦横無尽に歩いたり、平坦な道を走ったり、座禅を組んで宙に浮いたりできる……はず。

・感想

今回はロボットの回路部分を担当させて頂いたという事で、2年連続プログラムをやることとなりました。回路は回路と呼べるほど複雑でもなくプログラミング重視です。プログラムは汎用性を高くしてあります(ただ単純にモーターを複数制御するだけのもの)。いい加減アセンブラも上手くなってきたのでC言語をやろうと思っていますが、アセンブラでどこまで出来るかというのもやってみたいと思っています。来年は二足歩行ロボット(上半身も)を作ろうかと密かに企てている次第であります。きっと人間出来ないことはないです。まあその辺は次回に持ち越しという事で。

ARM

or...?

製作者 … H1大塚 雅人
協力者 … 物理部無線班の方々

・ ARM or...? とは…

〔ARM〕 …腕のことです。

この製作物はずばり『腕』です。

しかもただの腕ではなく、なんと人間と同じ動きをします。

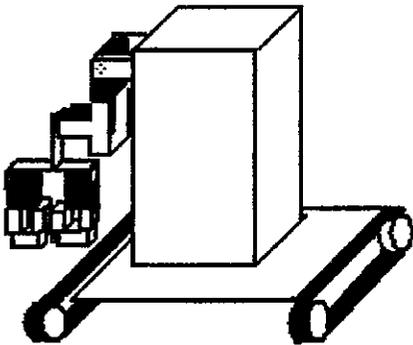
そして製作した結果、『Armor (装甲)』のようになってしまったため、このような名前がついてしまいました。

・ 何をするの？

別に人類の役に立つようなものではありません。ただ動くだけです。

まあ当初の予定では2台作って闘わせるつもりだったのですが…。

・ どんな形？



…本体はこんな形。

・どんな仕組みで動くの？

まず、人間の体にポテンションメーター（可変抵抗）を取り付けます（手術するわけではなく、スーツで）。そのポテンションメーター（可変抵抗）の抵抗値から、現在の腕（肩、指なども）の角度を算出します。その角度をPIC というもので処理し、サーボモーターに出力します。すると、ロボットも人間と同じポーズになるのです。

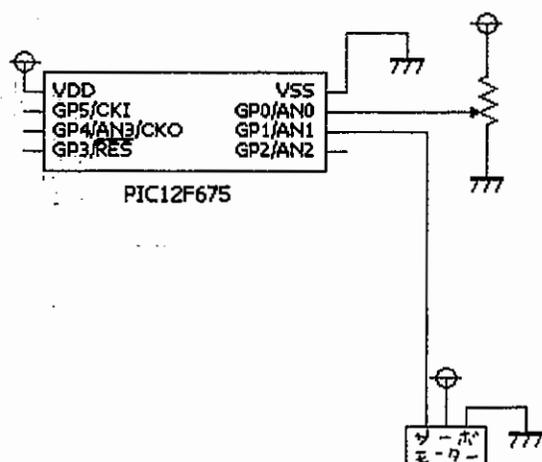
上文に出てきた『サーボモーター』についてなのですが、簡単に説明すると、「プログラムしだいで角度を自在に操れるモーター」なのです。このサーボモーターは、各社からいろいろなものが出ていますが、ARM or ...?ではもっとも安価なものを使用しています。

・仕様

仕様…。特にないですけど、言うならば人体に取り付ける部分（スーツ）の部分が伸び縮みするようになっており、大体の人間に取り付けられるようになっています。まあこんな手のこんだことをするような人はいないと思いますが…。

・回路図

モーターひとつにつきこの回路ひとつ



・プログラム

全部大体こんな感じのプログラムです。

```
#include <12f675.h>
#DEVICE ADC=10
#fuses INTRC_IO,NOWDT,PUT,NOPROTECT,
NOMCLR
#use Delay(CLOCK=4000000)
#byte PIR1 = 0x0C
long kakudo;
#int_timer0
void Interval(void)
{
    long width;
    set_timer0(178);
    width=900+(1280-kakudo-(kakudo>>2))
;
    PIR1 = 0;
    set_timer1(65536 * width);
    output_high(PIN_A1);
    while((PIR1 & 0x01) == 0){};
    output_low(PIN_A1);
    PIR1 = 0;
}
void main()
{
    setup_timer_0(RTCC_INTERNAL|RT
CC_DIV_256);
    set_timer0(178);
    enable_interrupts(INT_TIMER0);
    setup_timer_1(T1_INTERNAL|T1_DIV
_BY_1);
    setup_adc_ports(aAN0 | VSS_VDD);
    setup_adc(ADC_CLOCK_DIV_8);
    while(1)
    {
        set_adc_channel(0) ;
        delay_us(50);
        kakudo = read_adc();
    }
}
```

900+(1280-kakudo-(kakudo>>2)のあたりをうまく変えてちょうどよい角度に調節します。

・感想

このロボットはどちらかといえば簡単なほうなので、我こそはと思う人はどうぞ作ってみてください。(まあ道具等は必要ですが…。)

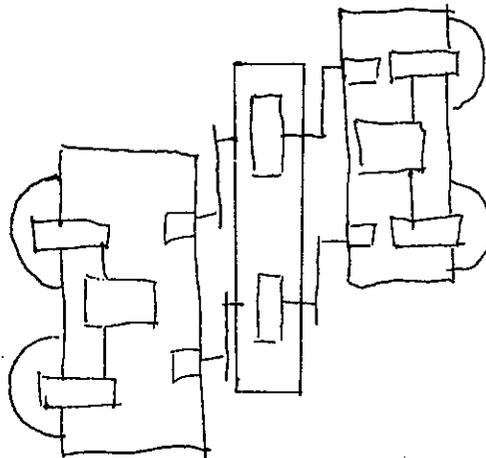
CLIMBER-2 ~A Climbing Addict~

製作者 H1 竹島 啓純
協力者 物理部の皆さん&OBの皆様

~The Outline~

このロボットは窓に引っ付いて窓を登るロボットです。
窓には吸盤で引っ付きます。
ちなみに題名には”Climber-2 A Climbing Addict(登り中毒)”
とかいてあります。

~The Appearance~



全長約30CM.

* 絵が汚いのはつこ
まないでください。
美術で学年で下から
5番目(30人中)なもので。

~The Stories~

1999/11/31

俺は今窮地にたたされている。
なぜか？それは余命が後24日しかないからだ。いや確定したわけではない
、あることをすれば寿命が伸びる。
そのあることとは

強盗。そう強盗をすれば寿命が延びるのだ。しかもおもちゃ屋に・・・
なぜか？事の発端は昨日にさかのぼる。昨日、突如後ろから男に羽交い絞めにされ、ハンカチを口に当てられた。もがくと男の声が聞こえた。「やめてよね。君が僕にかなうはず無いだろう。今お前に遅効性の毒をかがせた。一カ月後にお前は死ぬ。死にたくなかったらこのテープの指示に従え」「くそっ、このガノタがっ・・・」この言葉を最後に意識が遠のいていった。しばらくして目がさめた。男に渡されたテープを聞いてみることにした。

「おはよう。目がさめたようだ。ゲームをしよう。今回の君のゲームだが、12月24日、銀行強盗に入ってくれ。そうすれば解毒剤が手に入る。もちろんこのゲームに関して、君もしくはその仲間が負傷または死亡しても当局は一切責任をとらないのでそのつもりで。検討を折る。ガチャッピー— このテープは3秒後に自動的に消去されます。」

1、2、3、バンッ。女性の機会音声の3秒後テープが本当に爆発した。「ほんとに爆発するのかよ」と、このわけのわからない状況に悪態をつきながらも俺は冷静だった。まず病院にいき毒の検査をうけた。しかし、結果は陽性。しかも該当する解毒剤なし。こうなったらマジでクリスマスに銀行強盗するしかない！まずは偵察だ。よし偵察ロボットを作ってやる。あと脱出用のつなぎもいるな。それに・・・

こうして俺はClimber-2を作りはじめた

毒の発動まであと**24**日。

(いったい何個の映画が混ざってるんだ？5つぐらいか？)

~The Circuit~

回路図自体は次のページにあります。でかくて入らないので・・・去年の回路の可変抵抗がロータリーエンコーダーになっただけの回路です・・・簡単簡単。あえて説明するならモータードライバのハイサイドとローサイド

でFETの種類が違うことと、TD62083についてぐらいですか・・・

まずモータードライバのハイサイドとローサイドでFETの種類が違う件について説明します。なぜこうしたかという、ハイサイドにPチャンネルのFETを使うと、ドレインがVccにつながっているため、完全にONにするためには

Vcc以上の電圧をゲートにかけなくちゃいけないわけです。しかし、シナ事は電源2つ以上使うか昇圧電源乗っけるぐらいしないといけないわけで、無駄でしかないわけです。しかしですよ。NチャンネルのFETを使用すれば、ソース電圧が0V以上のこの場合（下のモーターとローサイドのFETにオームの法則で電圧が発生しているため）、完全にONにするにはゲートを0Vにつなげばいいため、Vcc以上の電圧を用意しなくても完全にONにすることができるんです。

まあなんとすばらしいことでしょう。ではなぜローサイドはPチャンネルな

のか？それはローサイドには直列でモーターやハイサイドのFETとつながっている為、ハイサイドと逆にドレイン電圧がVcc以下のこの場合ゲートをVccにつなげば完全にONになるわけです。

次にTD62083についてですが、これはただのダーリントトランジスタアレイです。COMMON端子はついていません。トランジスタアレイでオープンコレクタ出力なので100Ωのラダー抵抗(千石店員には伝わらなかった…。集積抵抗のほうが一般的?)でプルアップしています。20Ωまで耐えられるんで20Ωにしたかったんですけど20Ω売ってなかったんで100Ωです。

ちなみに、FETは秋月電子で購入しました。選択基準はON抵抗の低さです。耐電流5A以上ON抵抗100mΩ以下を基準にして店のカタログで調べました。ディスクリート部品は規格表で調べるよりも店のカタログで探したほうがいいです。無い部品の方が選かに多いんで…

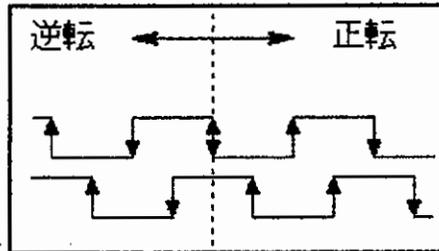
後、PICのRA4はオープンドレイン(オープンコレクタと何が違うんだ?)なので1kΩでプルアップしてます。

~The Programs~

~PIC1 ロータリーエンコーダー制御部分~

まずロータリーエンコーダー制御用のPIC1のプログラムの説明をします。

ロータリーエンコーダーはの図のような2つの信号を出力します。便宜上、上の信号を信号A、下の信号を信号Bとすると、正転と逆転は先にどちらの信号が立ち上がるかによってわかります。つまり信号Aによってマイコンに立ち上がり割り込みを入れ、信号BがHiなら角度をプラス1、Lowならマイナス1といったようにすればいいんですが、うちの部で主流のPIC16F648Aには立ち上がり割り込みは一個しかないんです。



と、いうことでポートB下位4ビットの変化割り込みを使用し割り込みがあった場合、前回のログとEX-ORをとりどのピンが変化したかを確認して、変化した後がHiだった場合、データセクタを切り替え信号Bを読みそのHi or Lowにより角度を変化させます。まあこんなこと言うよりも割り込み時に実行される関数を載せます。ちなみにコンパイラはCCSコンパイラです。ちなみにport_b_dataはport_bの下位4ビットを表して、port_bの4ビット目はデータセクタの選択ビットにつながっています。下に割り込み後のプログラムを掲載します。

ロータリーエンコーダー制御関数抜粋

```
Signed long angle[4]={0,0,0,0}; //角度保存用変数
#int_rb
void rb_isr(){
    int changes;           //変化保存用
    int count=0;          //周波数カウント用
    output_low(PIN_B4);   //信号Aに切り替え
    delay_cycles(2);      //データセクタ反応待ち
    changes = last_b ^ port_b.data; //変化したビットを見つける
    last_b = port_b.data;
```

```

while(count<4){
    if(bit_test(changes,count) &&! bit_test(last_b,count) ){
        output_high(PIN_B4); //信号Bに切り替え
        delay_cycles(2); //データセクタ反応待ち
        if(bit_test(port_b.data,count)){
            angle[count]++;
        }else{
            angle[count]-;
        }
        port_b.enable=0;
    }
}
}

```

～PIC2 FET制御部分～

PIC2ではPIC1から角度データを読み出し、内部のデータと照らし合わせてモーターのOn、Offを切り替える役目です。

ただ例の如くPICのピン数は少ないんで4bitのD-FF二つで8bitを6bitで出力しています。こうでもしないと貧弱なPICのピン数は足らなくなってしまいます。

PIC同士の通信ですが、当初、16F819or16F88でSPIを使おうと思っていたんですけど、PortBの変化割り込みにかぶるんで独自規格でやってます。命令は4bitパラレルの半二重通信でやってます。

モーター制御はPWMなど面倒臭いことはやってません。モーターは回るだけ回ってください。” The sooner, the better.” です。

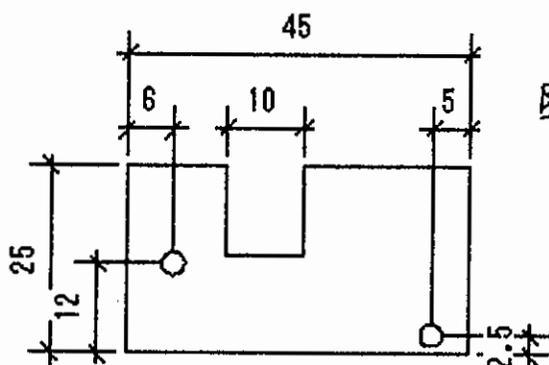


~The Fuselage~

ふう～。ついに本体説明ですね。実はこれを書いている今(2007/3/21)。本体に大変革が起きようとしているんです。そうでなくても本体系の話は苦手なのに・・・

何が起きているかという、レバー式吸盤であった吸着部に決定的な欠陥が見つかったのです。それは、レバー式は吸盤が窓に押し付けられて初めて強い吸着力が得られるんですがそれが得られそうに無い。じゃあどうする？真空ポンプだっ！ってことになるんですが、あれがまた糞高い。なるべく使いたくない。まあ試行錯誤していきます。ということで実は本体について語る事がまったく無い。困りました。まあ最初から回路メインの製作物だったんでご愛嬌ということで・・・

しかしです。一つだけ、ハードウェアで語れる部分があるんですよ。それは、モーターにいかにしてロータリーエンコーダーをつなげているか？ということです。それは下の図面をご覧ください。



図面手抜きで
すみません。

上の図のような鉄板をウォームギアボックスに取り付けて、そこにギアを取り付けたロータリーエンコーダーを取り付けることにより、ギアボックスの角度を読み取っています。

~The Postscript~

さて6ページに渡りおくりしてきました今回の製作物はいかがだったでしょうか？まあ想像どおり本体にはあまり手をかけず、回路重視の製作物になってしまいました。しかも締め切り前日の3月30日現在まだ動いていません。まあ回路は完璧なので文化祭当日には動いているとは思いますが・・・がんばりますよ

最近はまだマイコンの進歩が激しくなり、1チップの32bitマイコンまで出るようになりました。来年はそこらへんを使ってアザバックを作りたいですね。

Teacher - 3 -

~ Last Teacher ~

製作者 H1 中嶋剛大

協力者 物無の皆様

2004年下旬から、2005年文化祭に向けて、
川崎、木村、中嶋の三名によって製作された『射的ゲーム』。

初代 先生-Teacher-

2005年9月から、2006年文化祭に向けて製作され、
初代の精神を受け継ぎ、大きく進化を遂げた『本格射的ゲーム』。

二代目 Teacher2 ~帰ってきた先生~

2006年9月から2007年文化祭に向けて製作された、
今年の製作物は、今までのTeacherとは違う、『新しい射的ゲーム』。

そして、Teacher三代目にして、最後のTeacher…

三代目 Teacher -3- ~Last Teacher~

I. はじめに

Teacherについては上を読んで頂ければ大体分かると思います。今回は三代目のTeacherです。

まず、タイトルについてですが、この製作物のタイトルの由来は、初代の製作過程に使用した雑誌(金属加工等をする時に机や加工する金属などを傷つけないために使用します。)に載っていた漫画の、「先生！嘘だよな、先生！」という台詞が由来です。未だにその漫画は読んでいません。

そして、サブタイトルですが、外国の方は三部作が好き(?)なので、Teacherも三部作にしようということになり、Last Teacherということになりました。

それでは、詳しい仕様は次の頁から！！



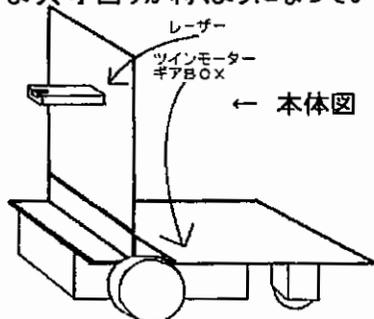
II. Teacher-3- の概要-外見

今回の製作物は、初代、二代目Teacher(銃のかたちのコントローラを用いた射撃ゲーム)とは少し違うコンセプトの射的ゲームで、自機を無線のコントローラで操作し、敵機を攻撃するというタイプのものになっています。



①自機

銃の弾となる、レーザーを搭載。赤外線を使うことにより、無線のコントローラで自由に操作可能。シンプルな構造により、小回りが利くようになっています。



②敵機

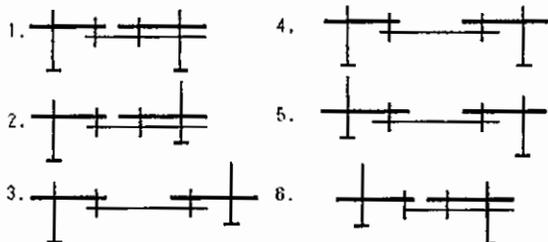
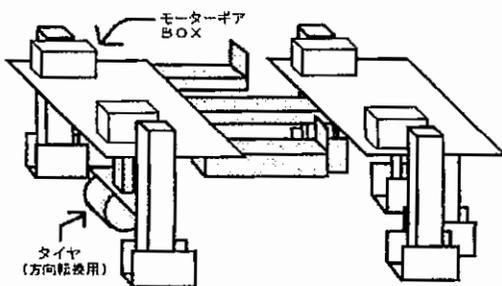
自機とは大きく違い、敵機はトリッキーな動きをする構造となっています。

フォトランジスタで自機のレーザー光線を受信できるようにしています。

また、超音波距離測定計を搭載することにより、自機との距離を測定し、逃げるように動くことができます。

そして、前のタイヤが回転することによって、方向転換が可能になっています。

また、真ん中の灰色の部分は『ピストン』と呼ばれる機構となっています。要するに、伸びたり縮んだりする機構のことですね。



移動は4本ある足を地面につけて滑らないようにした状態でピストンを動かすことによって、前or後ろへ進めるようになっています。

←移動方法の図

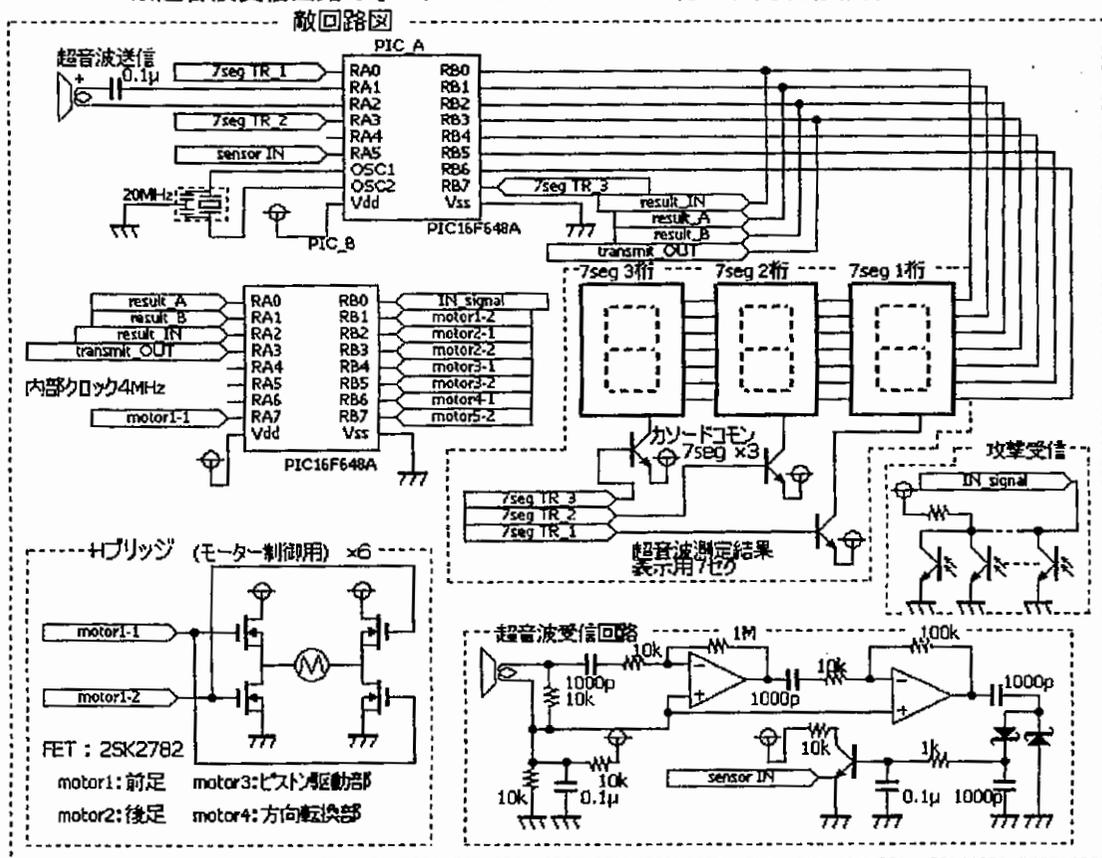
Ⅲ. Teacher -3- の回路図

①敵回路図 (見づらかったら申し訳ありません。)

PIC_A 超音波による距離測定。また、測定結果の表示、PIC_Bの制御

PIC_B モーター6個の制御、レーザー受信(フォトトランジスタ使用)

※超音波受信回路のオペアンプ2つは NJM4580(オーディオ用低雑音オペアンプ)。

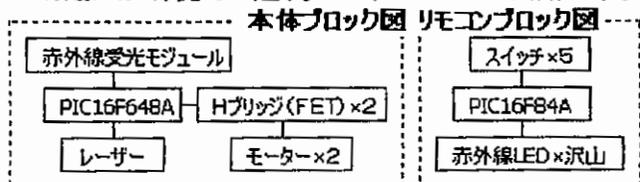


②自機、リモコン回路図 (単純なので、ブロック図で説明させていただきます。)

本体のPIC 赤外線を受信とモーター制御、レーザーの制御

リモコン側のPIC 赤外線を送信して自機移動攻撃制御

※とにかく、見ての通り。Hブリッジは上の回路にあるものと同じです。



IV. 超音波について

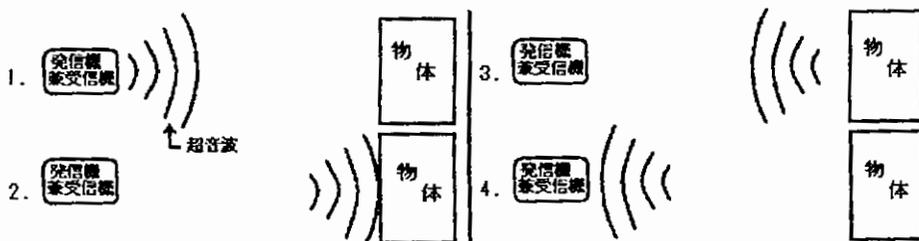
① 超音波とは？

誰もが超音波と言う単語を聞いたことがあるでしょう。

簡単に言うならば、超音波とは人間の耳では聞くことの出来ない音のことです。

コウモリやイルカが超音波を発して、物体との距離を測り、ぶつからない様に動いたり、餌を捕まえたりしているということは聞いたことがあると思います。

簡単に図示すると、下図のように音は飛んでいき、反射して返ってきます。

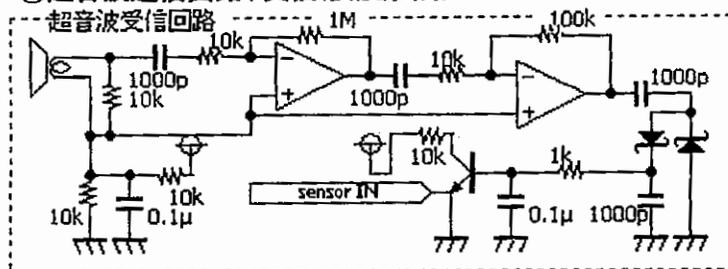


音を出してから帰ってくるまでの時間を計ることにより、自分と物体との距離を測ることができるのです。勿論、超音波でなく、人間に聞こえる音でも距離を測ることは可能です。目の見えない人はその様にして距離を感じるという実例もあります。

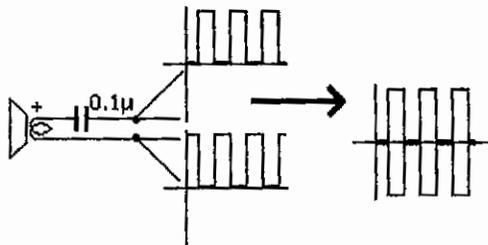
しかし、今回の場合は正確な距離を測るために、普段あまり飛び交っていないと思われる超音波(40KHzの高周波)を使用して測定します。

距離測定を光で行うのは非常に難しいので、音を使用します。空気中での光の速度は約30万km/秒ですが、音は約340m/秒です。比較して分かるとおりの音の速度は非常に遅いです。そのため距離を測ることが容易に出来るというわけです。(1cm先の物体に音を発射し、戻ってくるまでの時間が約0.0588秒(58.8ms)です。)

② 超音波送信回路、受信(検波)回路



上図は受信検波回路です。受信センサから入ってくる信号は非常に微弱な信号なのでオペアンプを2段重ねて増幅しています。増幅した後は、ショットキーバリアダイオード(高周波用ダイオード)を使用した半波整流回路を用いて検波しています。(図の1000pFはセラミックコンデンサ、0.1uFは積層セラミックコンデンサです。)

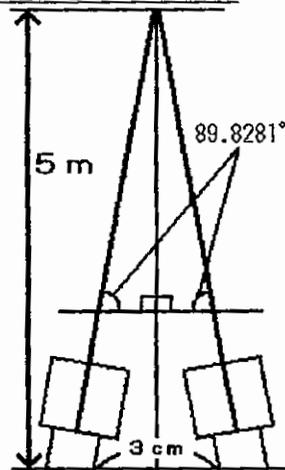


左図は送信回路です。
音は、遠くなるにつれて小さくなっていくため、遠くまで測定するためにはできるだけ大きな音を出したほうが得です。そのため、＋端子と－端子に逆の信号を送り、送信機にかかる電圧を2倍にして、音を大きくしています。

実はこの回路を動かすのに最も必要となるものが超音波送信センサと受信センサの成す角度です。

例えば5mくらいの距離を測りたいときは(右図)、二つのセンサの間を3cmと仮定すると、それらの成す角度は89.82811318度!! 最早殆ど90度ですね。

しかし、短い距離を測ろうと思えば思うほどこの角度は重要になってきます。と言っても、基本はほぼ90度。



V. プログラム

プログラムの全部を載せることは出来ないため、内容を簡単に書かせて頂きます。

①敵プログラム

PICA(超音波距離測定用)

PICBから信号受信 → 超音波送信 → タイマ割り込みを利用して距離カウント → 前回の結果と比較 → 比較結果をPICBに送信。

PICB(モーター駆動・レーザー制御用)

PICAへ信号送信 → PICAからの信号受信 → 受信結果に応じてモーター制御。

※ INT割り込みでフォトトランジスタによるあたり判定が入る。

②自機プログラム

本体プログラム リモコンからの信号受信 → モーター制御 or レーザー光線発射

リモコンプログラム 押されたスイッチに応じて信号送信。それだけです。

VI. 感想

今回は僕にとって初めてのロボット製作&アナログ回路製作となりました。

やはり金属加工は慣れるまでが難しいです。そして、計画をもっと立ててから作るべきでした。不具合連発で大変です。アナログとか無理です。ヤバすぎます。

点数を製作物につけるならば、65点くらいでしょうか…。(不具合連発&完動するかどうか微妙)来年は100点をつけられるように頑張っていきたいです。

色々大変でしたが、やはり新しいことに挑戦してみることは素晴らしいことです!是非皆さんも新しいことにチャレンジしてみてください!きっと楽しいはずですよ!

2007/03/31 H1 中嶋剛大

習字ロボット ^{ディ} ^{ライト} de "write"

製作者 M3 西村陽樹

協力者 物無の方々

～製作物について～

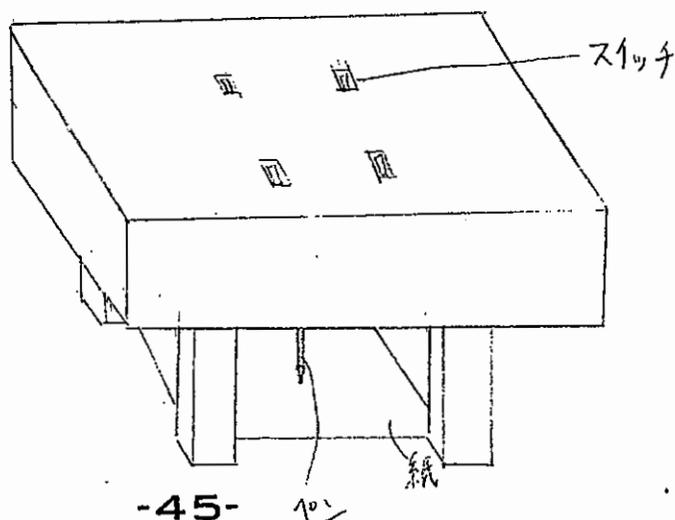
このロボットは、その名のとおり delight、つまり楽しみを与えるものです、なんて書きだしではじめましたが、このロボットが与える喜びは僕に自己満としての存在でしかなく、他人からみれば、これまた名のとおり紙に字を書くロボットです。聞いただけでは具体的な想像（人によっては創造）はできないと思うので、次に外観図と構造図、プログラムと回路図を載せました。

どのような仕組みで字を書くかという、基本的にはUFOキャッチャーと同じです。マイコンはPIC16F648を、モータはステッピングモータを3つ使っています。ステッピングモータにチェーン用のギアをつけ、そこにチェーンをつけてペンを動かします。1つ目のモータで縦移動を、2つ目のモータで横移動を行っています。これが平面移動です。3つ目のモータは字を書き終わった紙を移動させます。また、ペンの上下移動は3月31日現在まだ作っていませんが、今のところ電磁石を使い、ペンの部分に平たい永久磁石をつけて動かそうとおもっています。

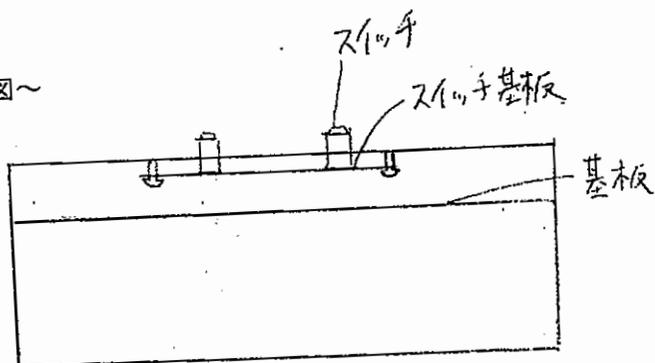
ちなみに製作物の名前ですが、家でJTのCMを見ているときに思いつきました。ひたすら delight、delight っているの、頭のこってしまいました。それが主な理由です。

進行状況については3月31日現在、プログラムは完成しましたが、3つのモータのうち1つにハイインピーダンスのような波形が見られるため、今の所直している状態です。また、最大の問題は、いまだにスイッチの入力をPICが読みこまないということです。非常にマズイです。文化祭までには必ず完動させます。がんばります。

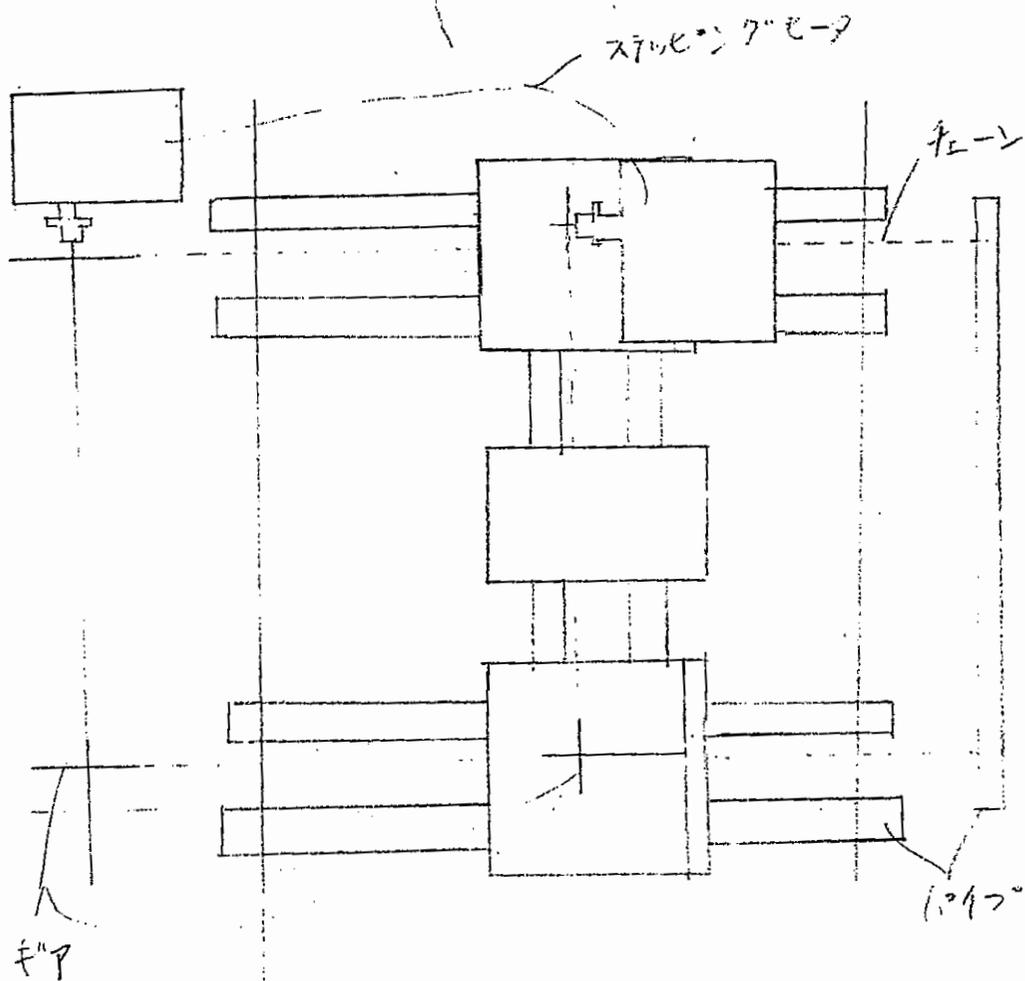
～外観図～

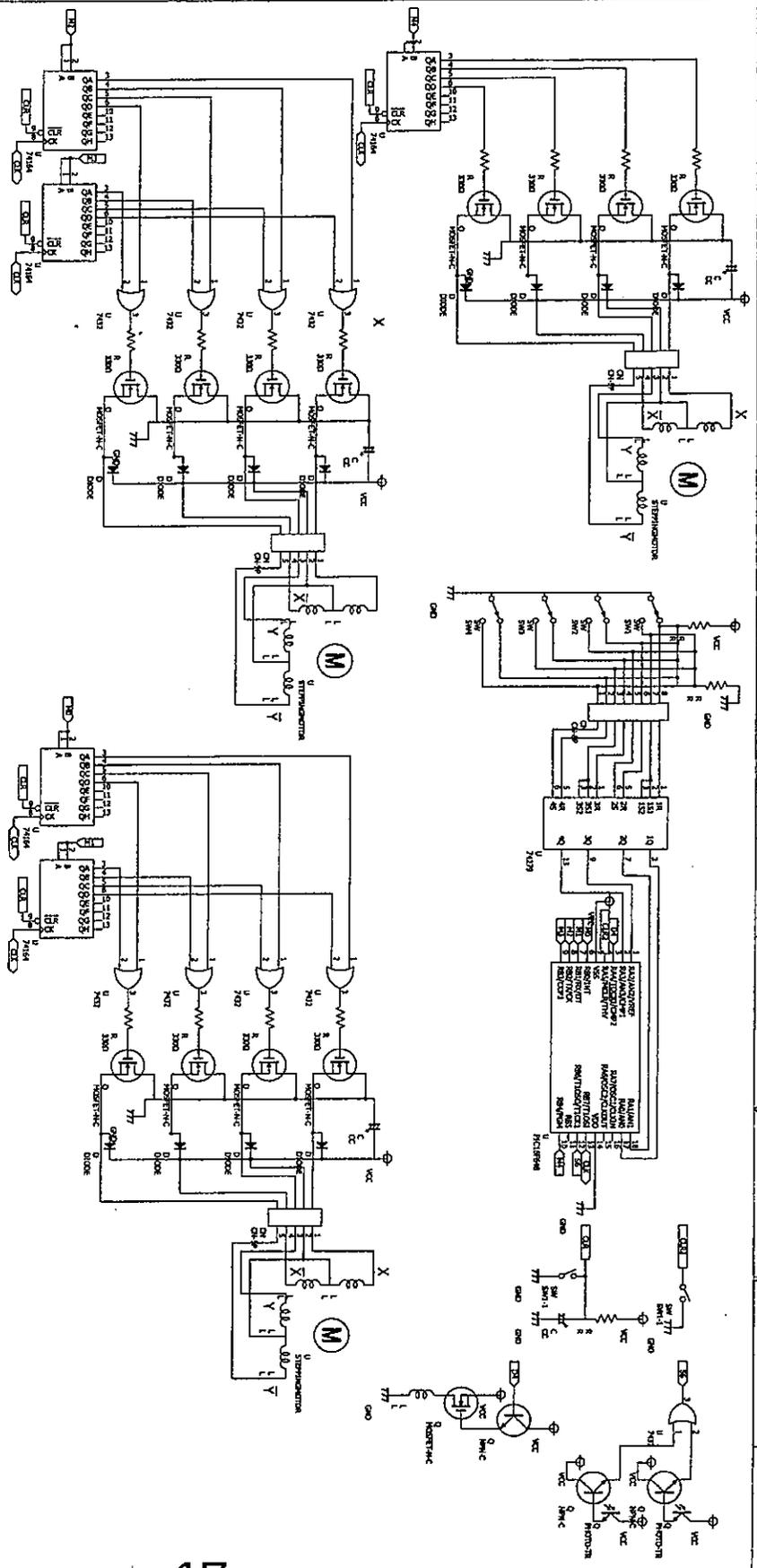


~構造図~



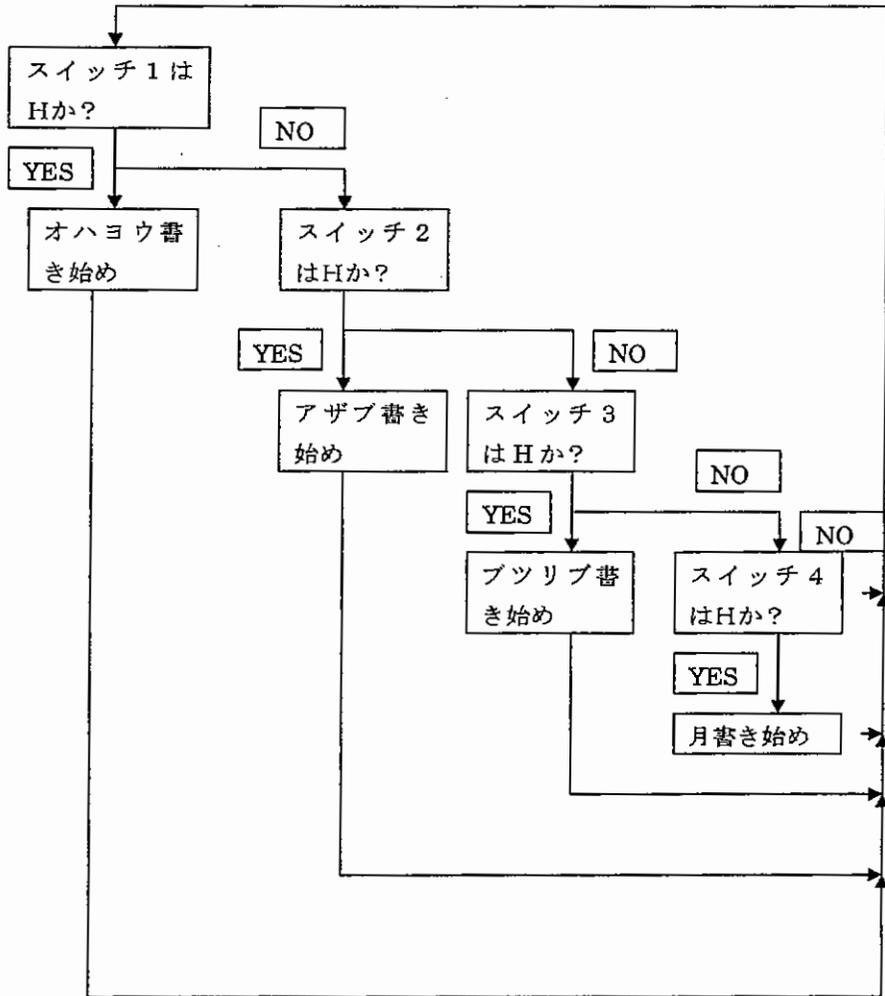
上から見た図





～プログラム～

プログラムは約1300行もあるため、とてもここには書ききれません。ということでとても簡単にフローチャートを載せました。



～ステッピングモータについて～

ステッピングモータ（図1）はパルスモータとも呼ばれ、パルスを与えることによって決められたステップ単位で回転するモータのことです。トルクがあまりでない代わりに、決められた角度分正確に回転することができます。プリンタの位置決め制御（ヘッド、紙送り）など幅広く使われています。回転の仕組みは、ローター（回転部分）に永久磁石をつけ、その周りの固定子（回転しない部分）に電磁石をおき、電磁石に信号をおくことで、ロータを回転させます。電磁石は円形の固定子に均等に配置されているので、ロータの数

により1ステップ数が変わってきます。たとえば4つの電磁石があったときに、1ステップは90度です。パルスの送り方にも何通りかありますが、今回この製作物で使用しているのは2-2相励磁です(図)。PIC16F648のPORTB,7に、タイマ0割り込みを使ってクロックを入れてそれを74LS164のCLKにつなげ、さらにSIにCLKの2倍のパルス幅のパルスを送ることで次のようなパルスを得ることができました。また現在使用しているステッピングモータの型番は17PM-K207-P2STです。



図

ステッピングモータについてもっと詳しく調べたいという方はこちらを参照してください。

<http://homepage1.nifty.com/rikiya/software/114stepping1.htm>

<http://www.interq.or.jp/japan/se-inoue/step1.htm>

<http://www.robotics.ee.shibaura-it.ac.jp/manual/chap2/motorhard.html>

=感想=

今回初めてロボットをつくり、本格的な金属加工や、プログラミングをしたわけですが、やはりロボットはゲームと違って、ただ回路が動けば展示できるようなものではなく、きちんと物理的な動作をしなければならないので、かなりむずかしいです。でもゲーム製作と違って、金属加工やプログラミングなど、いろいろな種類の活動ができるので、ロボット製作は楽しいです。もうそろそろ文化祭まで後一ヶ月となるのですが、このロボットはまだ動いてないです。このままだと本当にヤバイので、なんとか完動させます!

FIERCE FLIGHTIER

Fierce Flightier3

機種:ラジコンヘリコプター

最高速度:不明 km/h

航続時間: [不明]

空戦能力: [不明]

夜間戦闘 :可

武装1 :EDP5030 プロペラ 4門

全天候戦闘 :不可

爆撃能力: [不明]

ステルス :未装備

爆弾 :80gリポ2セルパック弾

必要生産資金 :12000ぐらい?

急降下爆撃 :やろうと思えばできるんじゃない?

必要生産アルミ :少々

上の装備画面は某海戦SLGのパケサ
本製作物とはあまり関係ないと思います。

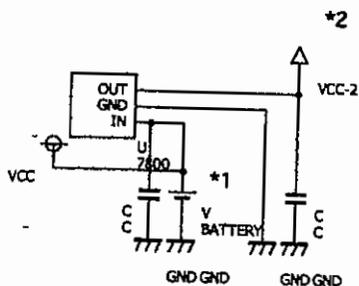
まず、ヘリコプターを作ろうと思ったのですが、実際のヘリコプターはご存知の通り推力を生むメインローターと、逆方向への回転を防ぐテイルローターが付いています。

ですが、メインローターの微妙な傾きによって姿勢制御するなど到底できなさそうなので、モーターを4つ使い、それぞれの出力を調整することで機体を傾け、前身するという方法を試すことにしました。

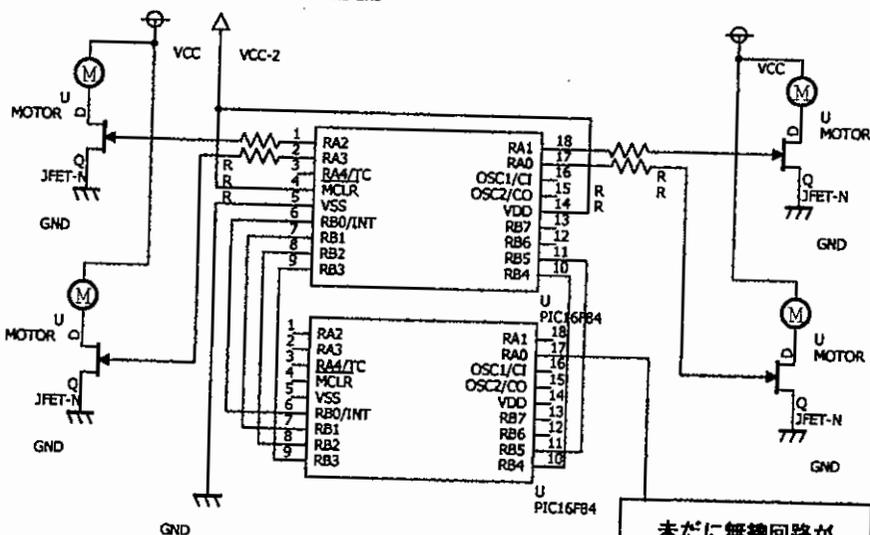
ほんとのSpec	
本体	バルサ製
メインローター	GWS 300H * 4
バッテリー	HYPERION LITESTORM 1500mAh 20C 2S
コントローラ	無線 (多分)
制御	PIC16F84A

製作 : M3 早川
協力 : 物無の方々

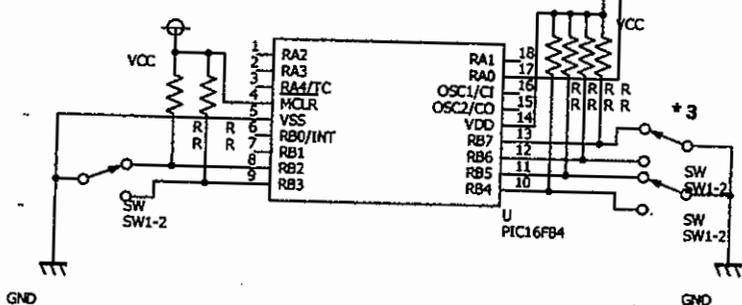
- 回路 -



- 本体側 -



未だに無線回路が
できていないんですが...
大丈夫なのかな



- 送信機側 -

*1 一つにつき、7.2V時180gもの推力を生み出す、この化け物のようなモーターに電力を供給するバッテリーもまた熱り。

そのパワーは本物ですが、もしプールに突っ込ませたり、不慮の事故でそれなりの高さから落下すれば大爆発してくれます。
(※)

*2 PICの電源供給用に5Vに変圧しています。

*3 スwitchの記号間違えたかも。ニュートラル、上、下の三方向に操作できるものです。

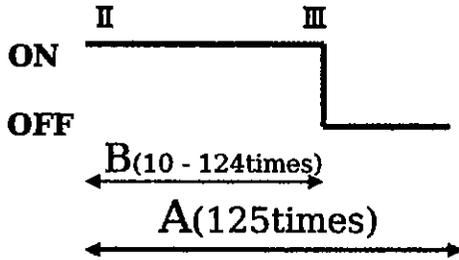
* 発振子は省略しています。

* 見にくいですか？(疑問形)

- Program -

- 本体側 -

→図解。



上昇、ホバリング、下降などの信号に応じて、モーターの回転数を変えてやるわけですが、今回はプログラムで擬似PWMを作り出しています。

全部書くのは面倒なので簡単に。

I：処理はすべてインターバルタイマとして使っている、タイマ0 割り込みの際行う。

II：A回カウントしたら、信号を確認し、モーターの回転数を設定。モーターを回転させる。

III：設定した回転数（Aより少ない、Bとする）をカウントし終わったら、IIに戻るまで該当モーターをストップ。

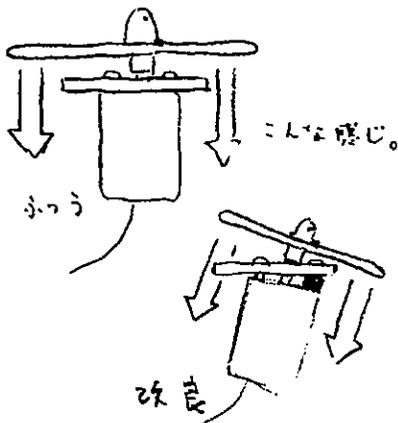
- 通信 -

今は有線でやっていますが（3月現在）、無線になる可能性があるので、シリアル通信に対応したプログラムを搭載しています。

本体、コントローラにそれぞれ受信、送信用のPICを。

誤受信を防ぐため、送信する信号に、コントローラのレバーの状態以外にもデバイス識別用の定数とかを含めたり、二重チェックを行っています。

- 飛行 -



さて、僕の手元にあるモーターは4つ。

すべてメインローターにすると、機体が凄いい勢いで回転しだして制御どころの騒ぎではありません。

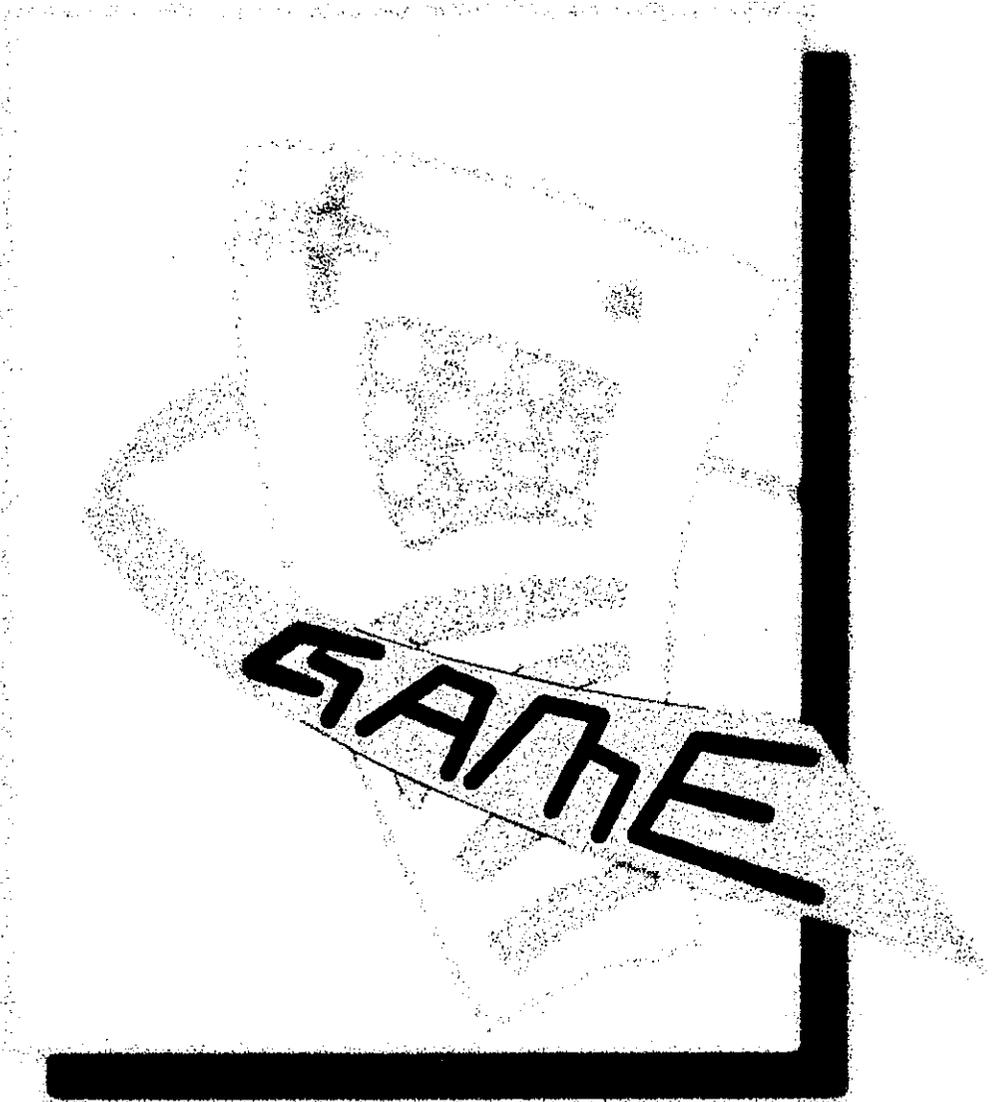
当初は某漫画の影響でうちわを付けて空気抵抗を大きくしようかとも考えましたが（

？つつ、順回転、逆回転にして回転を打ち消す方法も逆方向のプロペラがないため不可。

というわけで、一つ一つのモーターを微妙に傾け、自身のトルクを打ち消す方法を探ることにしました（左図参照）

あとは、あの化け物モーターががんばってくれるので重量のことはほとんど気にすることがありません。

飛びます。

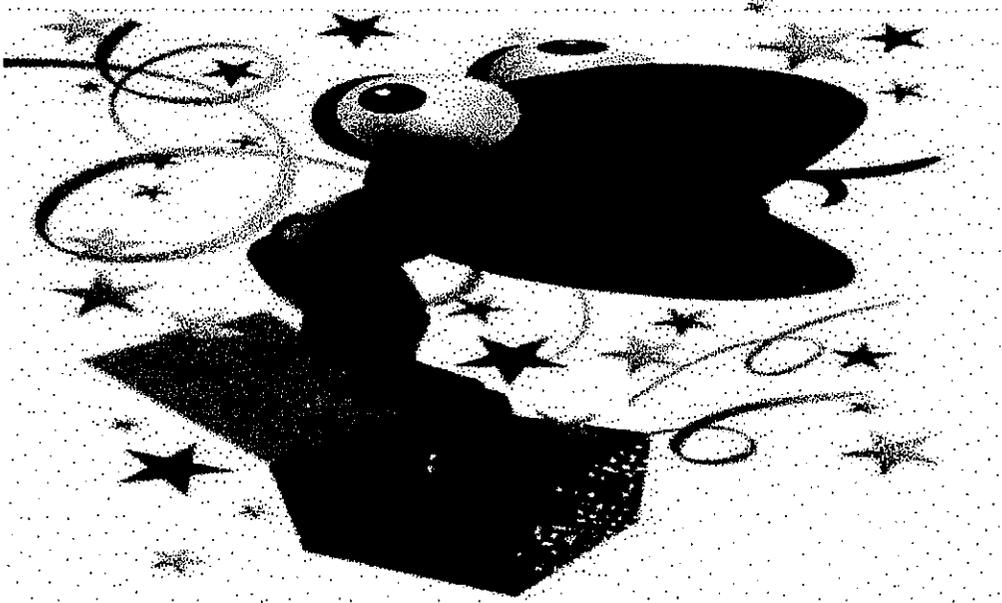


ヘヴィー級ゲーム

製作者：H1 川崎 敦史

協力者：物無の皆様、OB様

完成予想図！！



ある、ある、ねーよ・・・

すみませんが今からどう頑張ってもこのようにはなりませんね。つか根本的に、僕が作ったのはデジタルゲームでした。ではゲーム内容を。

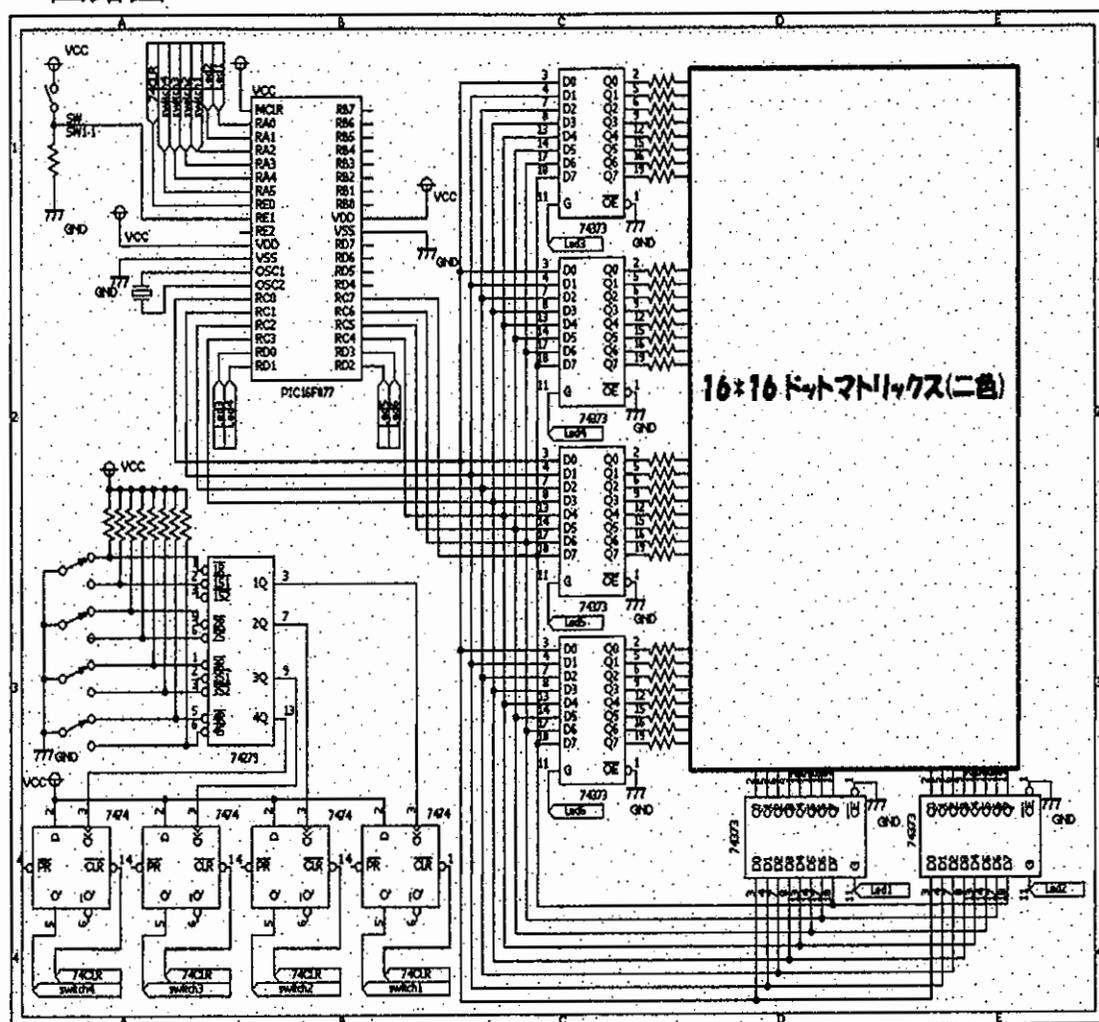
～ゲーム内容～

簡単に言えばヘビゲーです。ヘビゲーって何というそこの君のために説明しよう。ヘビゲーって言うには画面上にある餌みたいなのをとって自機が大きくなる。そして動きが早くなる。まあそれだけじゃゲームにならんので自機に重なったら負け、壁にあたったら負け、敵にあたったら負けなどの機能をつけて制限時間以内に何個餌を取れるかを競うゲームだと思う。(後に変更があるかも)それを16*16のドット

マトリクス（LEDの集合体みたいなもの）に表示させて引きこもりのヲタクちゃんを増やそうという計画です。

それではさくさくっと回路図でも

～回路図～



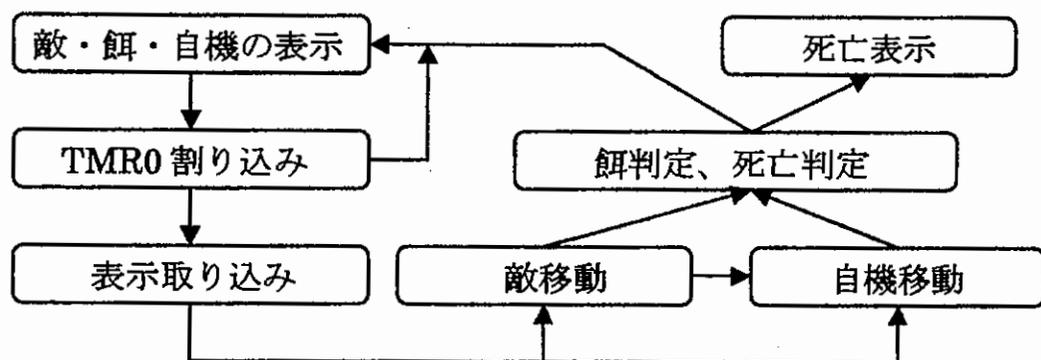
とまあ簡単な回路で、373 というDラッチ（データを一時的に保存するIC）を使ってPICのピンを増やしダイナミック点灯をしています。残りはスイッチ入力のためだけの回路で、ほとんどはPICのプログラム任せの回路となってしまう

した。

回路の点で語れるのはダイナミック点灯だけなのでその点について、簡単に説明を。今回は16*16の二色なのでアノード側8*2の二つ、カソード側は8*2*二色の4つの373を6つ使っています。373にはGというピンがあり、そこはデータを代入するかどうかを決めるものでそこがHならば代入、Lならば保存となっています。そこでデータの代入・保存を連続的に行い緑・赤を一列一列表示させています。ただおそらく超人的な目の力を持つ者がいない限り、普通に表示されているように見えます。何故なのか？それはPICの20M発振のパワーを使い常人には見えなようなスピードで先程言った表示を行っているからなのであります。

～プログラム～

あれ？おかしいな。プログラムが長すぎるぞ。このままでは僕一人で回路図集を埋め尽くしてしまう(嘘)。ということで部分的に省略したプログラムをのせたいと思います。しかしこのページのスペースがめっちゃ微妙なので、この小スペースにフローチャートでも載せようかと。



では次のページにプログラムの一部を載せます。

```

LIST      P=PIC16F877A
INCLUDE P16F877A. INC
  ___CONFIG _HS_OSC &
_WDT_OFF & _PWRTE_ON &
_CP_OFF & _DEBUG_OFF &
_WRT_OFF & _CPD_OFF &
_LVP_OFF & _BODEN_OFF
ORG       0
GOTO     SETUP
ORG       4
CLRF     INTCON
GOTO     WARICOMI
SETUP
BCF      ADCON0, 0
BCF      STATUS, RP1
BSF      STATUS, RPO
MOVLW   B' 00111100'
MOVWF   TRISA
CLRF     TRISB
CLRF     TRISC
CLRF     TRISD
CLRF     TRISE
MOVLW   B' 00000111'
MOVWF   ADCON1
MOVLW   B' 01000111'
MOVWF   OPTION_REG
BCF      STATUS, RPO
          省略
SETUP2
MOVLW   B' 10100000'
MOVWF   INTCON
MOVLW   90H
MOVWF   TMR0
;*****表示*****
MAIN
BTFSZ   HOLD7, 4
GOTO    MAIN3
BTFSZ   HOLD7, 1
GOTO    $+3
BTFSZ   PORTA, 2
GOTO    MAIN2
BTFSZ   HOLD7, 0
GOTO    $+3
BTFSZ   PORTA, 3

```

```

GOTO    MAIN2
BTFSZ   HOLD7, 3
GOTO    $+3
BTFSZ   PORTA, 4
GOTO    MAIN2
BTFSZ   HOLD7, 2
GOTO    $+3
BTFSZ   PORTA, 5
GOTO    MAIN2
GOTO    MAIN3
MAIN2
CLRF    HOLD7
BTFSZ   PORTA, 2
BSF     HOLD7, 0
BTFSZ   PORTA, 3
BSF     HOLD7, 1
BTFSZ   PORTA, 4
BSF     HOLD7, 2
BTFSZ   PORTA, 5
BSF     HOLD7, 3
BSF     HOLD7, 4
MAIN3
BSF     PORTE, 0
CALL    CLR
MOVF    TATE1, W
MOVWF   PORTC
BTFSZ   HOLD1, 0
CALL    HL6
BTFSZ   HOLD1, 0
CALL    HL5
MOVF    YOKO1, W
MOVWF   PORTC
BTFSZ   HOLD1, 1
CALL    HL4
BTFSZ   HOLD1, 1
CALL    HL3
CALL    CLR
MATO
MOVF    MATO1, W
MOVWF   PORTC
BTFSZ   HOLD9, 2
CALL    HL1
BTFSZ   HOLD9, 2

```

```

CALL    HL2
MOVF    MATO2, W
MOVWF   PORTC
BTFSZ   HOLD9, 3
CALL    HL3
BTFSZ   HOLD9, 3
CALL    HL4
CALL    CLR
TEKI1
MOVF    TEKI1_1, W
MOVWF   PORTC
BTFSZ   HOLD15, 0
CALL    HL1
BTFSZ   HOLD15, 0
CALL    HL2
MOVF    TEKI1_2, W
MOVWF   PORTC
BTFSZ   HOLD15, 1
CALL    HL3
BTFSZ   HOLD15, 1
CALL    HL4
CALL    CLR

```

省略

;*****移動判定*****

```

WARICOMI
DECFSZ  HOLD17
GOTO    WARICOMI_1
          敵移動省略
BCF     HOLD21, 6
DECFSZ  HOLD6
GOTO    WARICOMI4
BTFSZ   HOLD7, 0
CALL    LEFT
BTFSZ   HOLD7, 1
CALL    RIGHT
BTFSZ   HOLD7, 2
CALL    DOWN
BTFSZ   HOLD7, 3
CALL    UP
WARICOMI2
BCF     HOLD7, 4
MOVF    HOLD13, W
MOVWF   HOLD6

```

省略		HANTEI2	BCF	PORTE, 0
自機敵当たり判定		BTFSC HOLD1, 0	GOTO	WARICOMI2
ZIKITEKI		GOTO HANTEI4	LEFT2	
MOVF HOLD1, W		BTFSS HOLD9, 2	RRF	HOLD3
XORWF HOLD15, W		GOTO HANTEI5	BTFSS	HOLD1, 1
MOVWF HOLD19		GOTO WARICOMI4	GOTO	LEFT3
BTFSC HOLD19, 0		省略	BCF	HOLD1, 1
GOTO ZIKITEKI2		WARICOMI4	RETURN	
BTFSC HOLD19, 1		MOVLW B' 10100000'		省略
GOTO ZIKITEKI2		MOVWF INTCON	HL1	
MOVF YOKO1, W		MOVLW 90H	BSF	PORTD, 0
MOVWF HOLD19		MOVWF TMRO	NOP	
COMF HOLD19		RETFIE	BCF	PORTD, 0
MOVF HOLD19, W		省略	RETURN	
IORWF TEKI1_2, W		LEFT		省略
MOVWF HOLD19		BSF HOLD9, 0	HL3	
COMF HOLD19		BCF HOLD9, 1	BSF	PORTA, 0
BTFSC STATUS, Z		BCF STATUS, C	NOP	
GOTO SUB1		RRF MATO4	BCF	PORTA, 0
省略		BTFSC STATUS, C	RETURN	
*****餌取り判定*****		RRF MATO4		省略
HANTEI		BCF STATUS, C	END	;ENDRIO
BTFSC HOLD1, 1		RRF HOLD3		
GOTO HANTEI2		BTFSC STATUS, C		
BTFSS HOLD9, 3		CALL LEFT2		
GOTO HANTEI3		MOVF HOLD3, W		
GOTO WARICOMI4		MOVWF YOKO1		

とまあかなり省き省きなんですけど・・・プログラムが見たいという人は言ってください。

～感想など～

去年はロボットを作ったので今年はゲーム系を作ろうと思ってやった結果、やはりプログラムがメインだったためプログラムの腕も上がったと思う？去年同様、部品がバツコバツコと逝き 877 君すみません。プログラムの面では、新しい機能が追加されるためにどんどん足していったためにプログラムの長さ膨大になるほか、1ヶ月前に作ったプログラムなど何をしているのかがわからず、理解するのに1日をつぎ込んだことがあったようななかったような・・・まあ悲しいことばかりではなく、中1の竹下君と色々楽しくできたからよかったかな。

エレキサイティングホッケー

中3-6 村瀬

1, 概要説明とお詫び

これは、簡単に言ってしまうとゲームセンターによくある「エアホッケー」。(空気で浮いて滑る軽いプラスチック製の円盤を、2人のプレイヤーがラケットの役割をたす円盤。(以下「ラケット」.)で打ち合い、お互いのゴールに入れて点を競うゲーム。)を、16×16のLED画面。(ドットマトリックス、以下「ドトマト」.)でやろうというものです。ただし、ルールを煩雑でなくするためゴールに幅は設けず、プレイヤーに接する一辺全体をゴールとしています。

表示方法はダイナミック点灯もどきです。「ダイナミック点灯」はドトマトにはよく使う、高速で動かすことにより残像で複数個のランプが点いているように見せかける方法ですが、このゲームの場合、必要なのは円盤。(打たれる方。)を表すランプ一個のみなので、低速で、残像は使わずやっています。しかし、念のため必要となればダイナミック点灯もできないことはないような回路になっています。

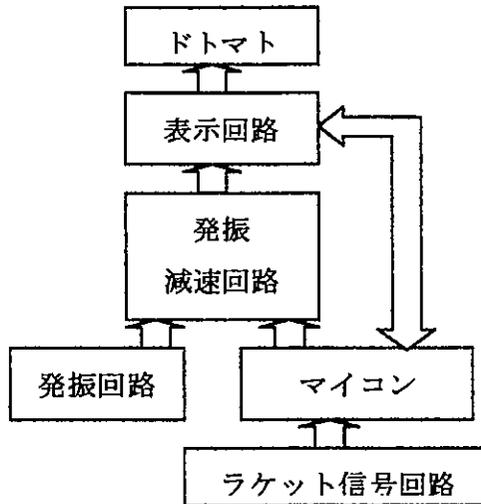
予定では、画面を円盤が駆け巡り、プレイヤーがラケット。(スイッチではなくプレイヤーが実際に手に持つものを用意し、そこから回路へと信号を送るつもりです。)を振り回し魔球を駆使して戦うというのを目指していたのですが…

正直に書けば、この回路は3月下旬現在、完成にはほど遠く、基本的な動作すらまなりません。無論、当初の野望だった「魔球」「障害物」も実現できていません。

回路としては、おおまかに分けてドットマトリックスに円盤を表示する部分と、プレイヤーが球を打った方向を信号としてPICに送る部分とがあるのですが、前者しか作っていないので、ここに載せるのはその部分の。(動作の保証されない。)回路図のみとなってしまいます。読んで下さる方には深くお詫び申し上げます。

2, 回路

ブロック図



(ラケット信号回路の回路図はなし.)

① ドットマトリックス

この回路の要, ドットマトリックスです. 僕はこれを自作したので, 実際に使っているのは市販のものではありません. 内部構造は見ての通りですが, 市販のものにはもしかしたら抵抗が内蔵されているかもしれませんので, その場合 VCC, GND に抵抗を通す必要はありません. ご注意下さい.

使い方は, 例えば左上端の LED を光らせたい場合, 縦の上端のピンに H, 横の左端のピンに L を流せば光ります.

② 表示回路・横

画面横方向を司る表示回路です. クロックが 7419 の COUNTUP に入れば右, DOWN に入れば左に動きます.

縦の表示回路も, 右上の AND が無い点と, 74138 と 373 との間に 74541 が挟まっ

ている点をのぞけば、これと同じです。縦の方では、右側のタグが「164QD」→「164QC」, 「LEFT」→「DOWN」, 「RIGHT」→「UP」, 「RA1」→「RA0」となります。

③発振減速回路

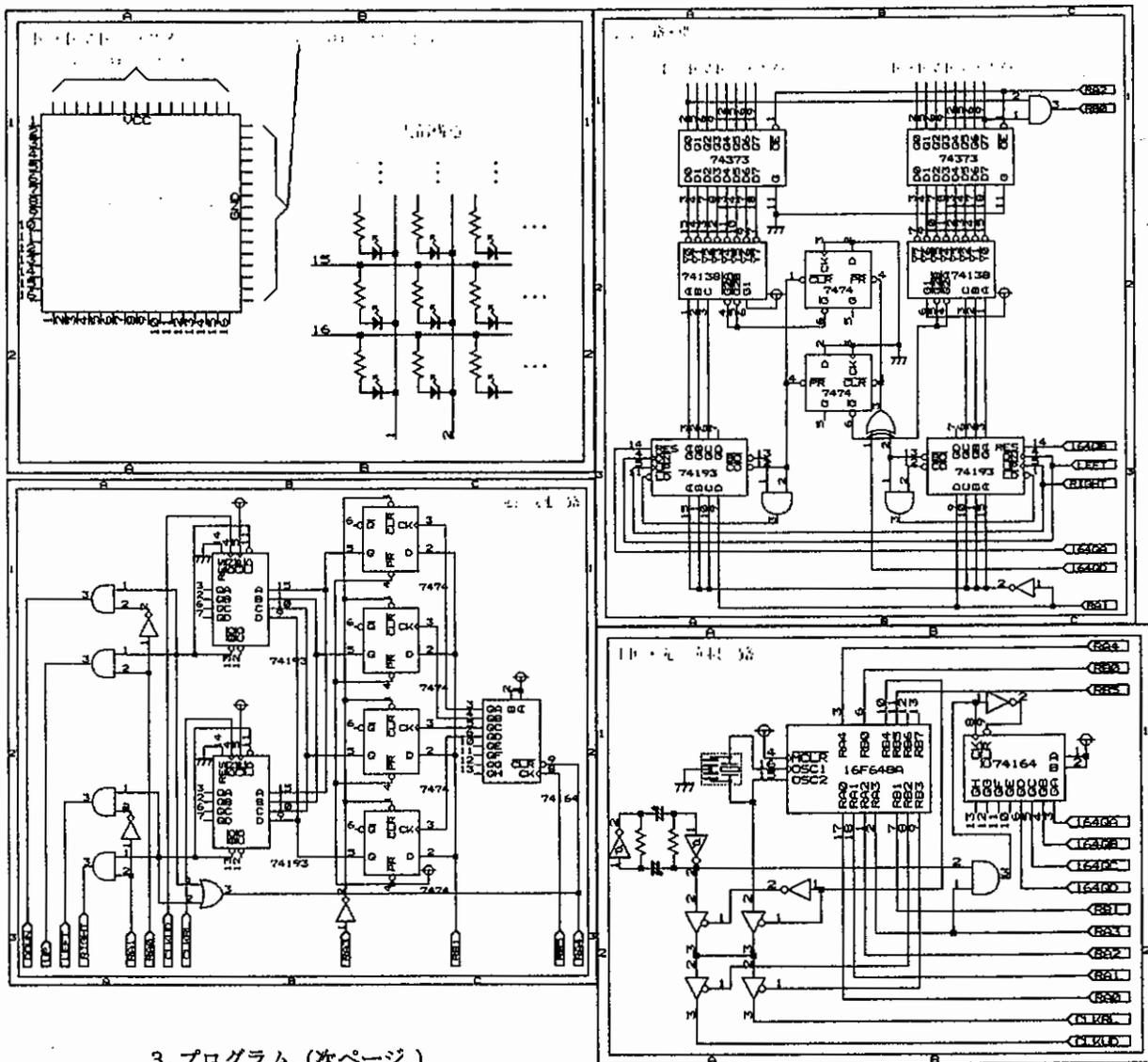
読んで字の如くです。74193 にあらかじめデータを入れロードさせておくと、COUNTDOWN にクロックがつないでありますが、「クロックの周波数×データに入れた数分の一」だけ BORROW が立ち下がります。BORROW は LOAD に直結しているので、BORROW が立ち下がればデータが再びロードされ、カウントダウンが繰り返されます。この BORROW を出力にとれば、「データに入れた数分の一」に発振を遅くしたものが手に入ります。

③ PIC・発振選択回路

マイクロコンピュータには PIC16F648A を使っています。入門者向けとされ有名な 16F84 (A.) とピン配置や命令数は変わらないのに、メモリ容量が大きく、タイマは 3 種類に増え、その他色々と機能が追加されているからです。しかも、84A より安い。(笑.)!

3 ステートバッファ。(74LS125.) が左下に並んでいますが、これは G に H を入れると出力が Z になるというものです。PIC の RB4 を H にするか L にするかで、低速で回路を動作させたい場合には左の発振回路、高速で処理したいようなことがあれば右の発振子、のいずれかのクロックを出力します。なお、この回路の IC は全て 74LS シリーズです。LS の IC は認識可能な周波数の最高が 25MHz なので、20MHz のセラミック発振子でも動作します。

右にある 74164 は回路をクリアするためのものです。



3. プログラム。(次ページ。)

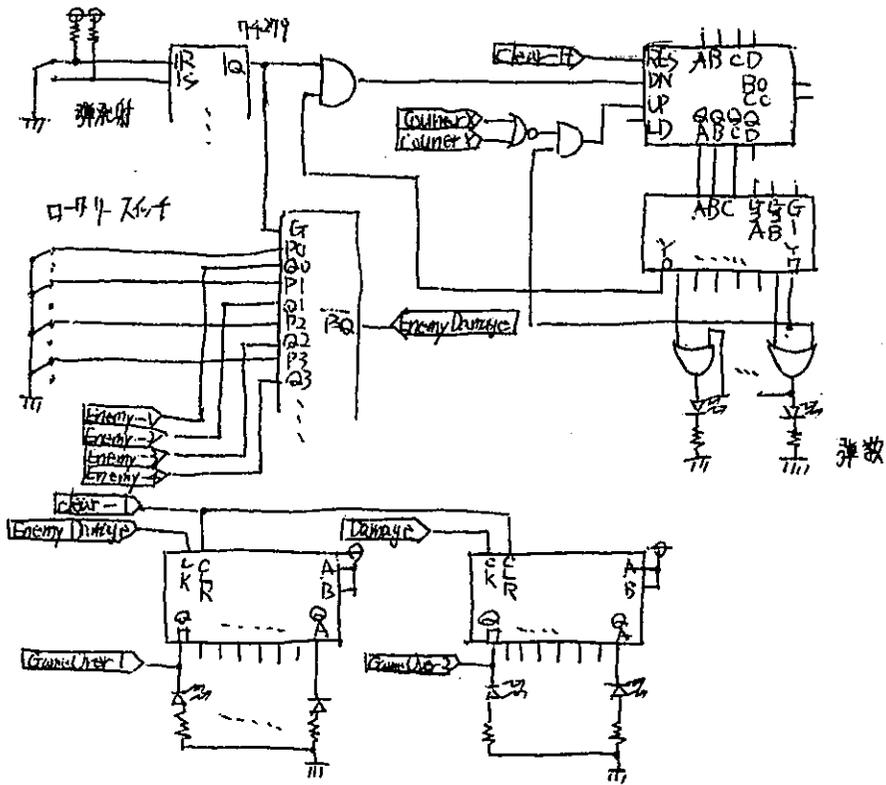
アセンブラによるこの回路のプログラムですが、冒頭でお断りした通り、基本的な動作すらままならないので、ここに書くのは現在唯一動作が確実な「回路をクリアし円盤を初期位置。(左上.)に表示するプログラム」です。クリアする以外にもしてくれないプログラムですが、ご参考までに。

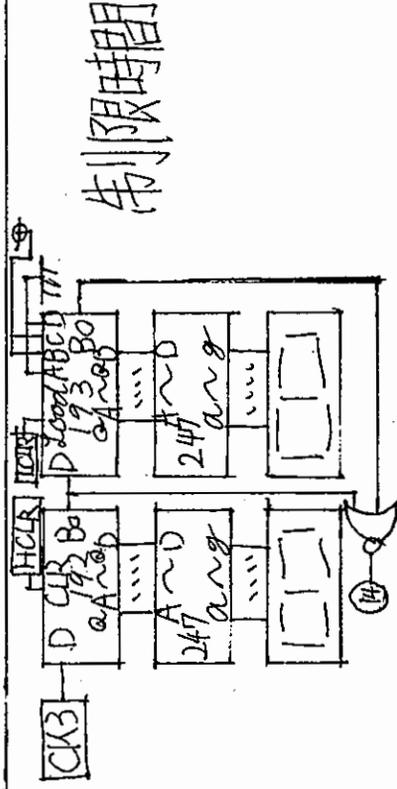
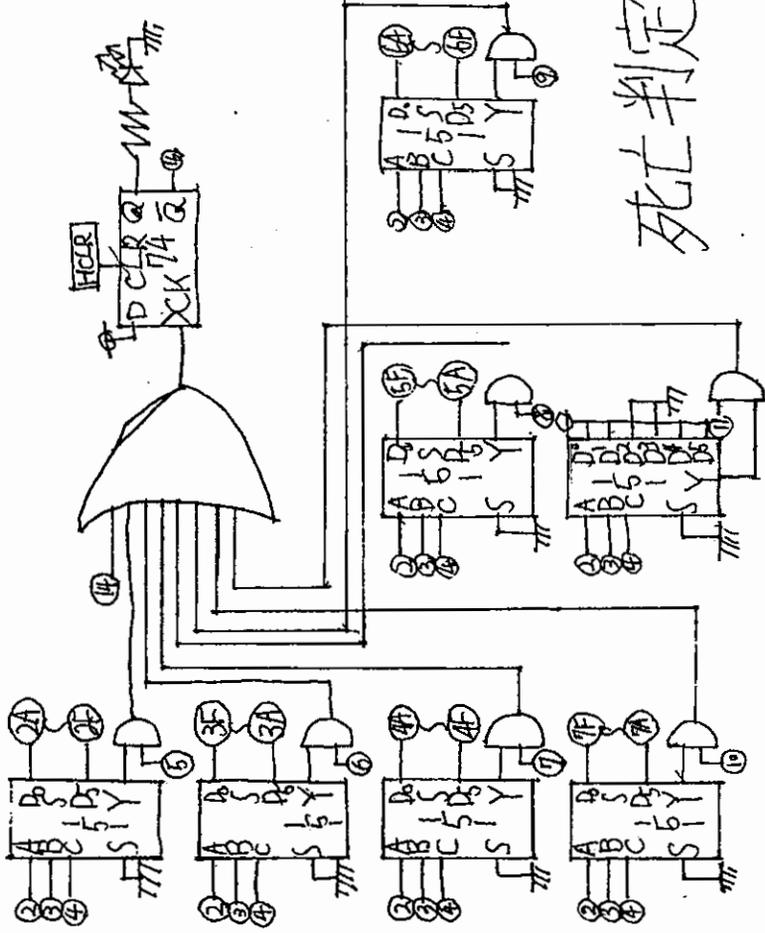
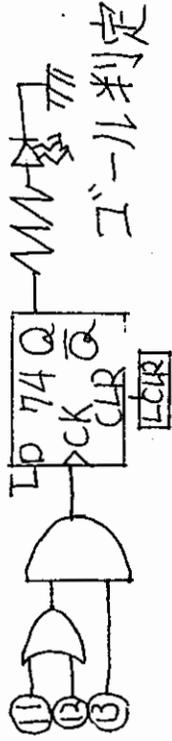
```

LIST      P=OIC16F648A
INCLUDE  PIC16F648A.INC
_CONFIG  _HS_OSC & _BODEN_OFF & _BOREN_OFF & _CP_OFF &
_DATA_CP_OFF & _PWRTEN_ON & _WDT_OFF & _LVP_OFF & _MCLR_OFF

ORG      0                MOVWF  PORTA
GOTO     SETUP           MOVLW  B'00011100'
ORG      4                MOVWF  PORTB
GOTO     SETUP           MAIN
SETUP                                BSF    PORTA,3
BSF      STATUS,RP0       MOVLW  056H
CLRF     INTCON           MOVWF  24H
MOVLW   B'10101111'      MOVWF  23H
MOVWF   OPTION_REG       MOVWF  22H
MOVLW   B'00010000'      NOP
MOVWF   TRISA            DECFSZ  22H
MOVLW   B'00000001'      GOTO   $-2
MOVWF   TRISB            DECFSZ  23H
BCF     STATUS,RP0       GOTO   $-5
CLRF    PORTA            DECFSZ  24H
CLRF    PORTB            GOTO   $-8
CLRF    22H              BCF    PORTA,3
CLRF    23H              BCF    PORTA,2
CLRF    24H              LOOP
MOVLW   B'00000100'      NOP
GOTO    LOOP            END

```

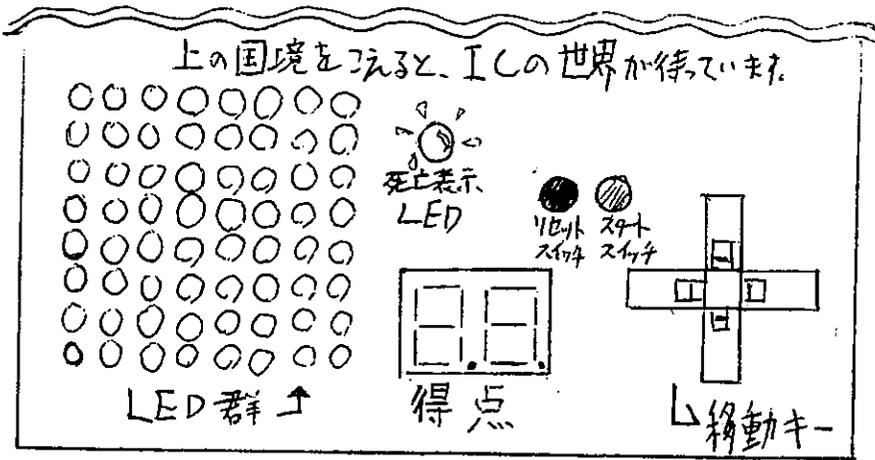


木ハークラック

～地獄～

制作者 M2 相辺 駿介 回路図 物無のみなさん
 協力 etc. 物無のみなさん.

⊗ ゲーム説明!



- ① 子供(緑)が点灯し消える。
- ② 自機(緑)と敵(赤)が出現する。
- ③ よけながら点灯した子供を助けに行く。敵に当たると死す。子供はかまはば、助けられ、1点。
- ④ ①に戻る。



時1は 2123年。

9/12 イギリス ロンドンに、ある施設がOPENした。

その名も『ネバーランド』。

そこは大人の入れない子供の世界が広がる
遊園地だった。

ネバーランドの子供たちの中にはリーダーがいた。
その名もパンピーター。皆の人気者だ。
皆にはパンピーと呼ばれている。

ある日、パンピーは右で作られた矢を見つけた。

「これは...いったい?---うわ?」

~~矢~~パンピーに矢がささり、倒れた。...

目が覚めると、宙に浮いていた。
「浮いている。」そう...パンピーは空を飛ぶようにな
っていた。すると、空からいつもは見えないボールを
発見した。「何だろうか...」中へ入って行きた。

中には2人の男がいた。「ミカエル・ジャワソノ様」

「何だい」「なぜこの施設を建てたのですか?」「それは...だね」

「?」「食うためだよ...」「あゝ...」ミカエルはハナをとぼし、男にHIT。

「...うてぶ」男は死んだ。「ん?そこにいるのは...ヒーター?」

「アハ...」パンピーはすぐさまにげる。

「そ、ヒーターに知られたら...生かしてはおけん!」

ミカエルは追う。

「うわ...来た!子供たちが危険だ!」

「ヒーター...園にはの子んじやないぞ!!ナイザーがちまけてやる」

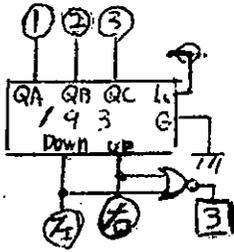
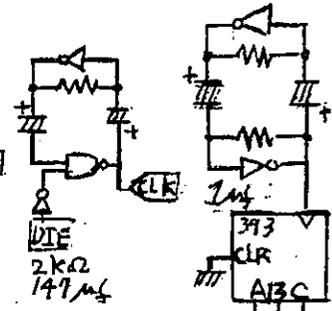
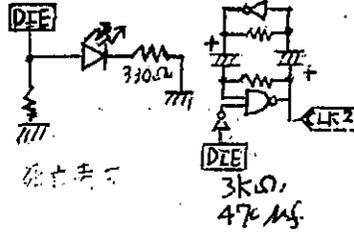
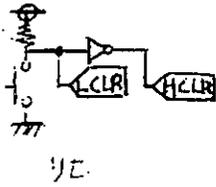
ミカエルはハナを飛ばしてくる。

「みんな、いくぞ」「どこに?」

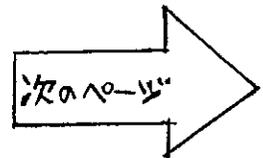
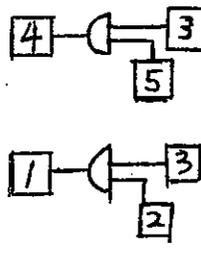
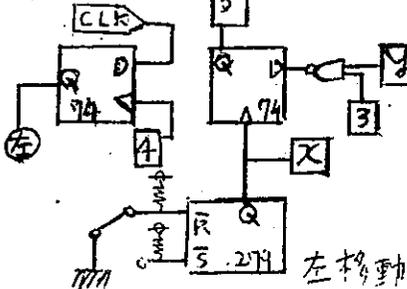
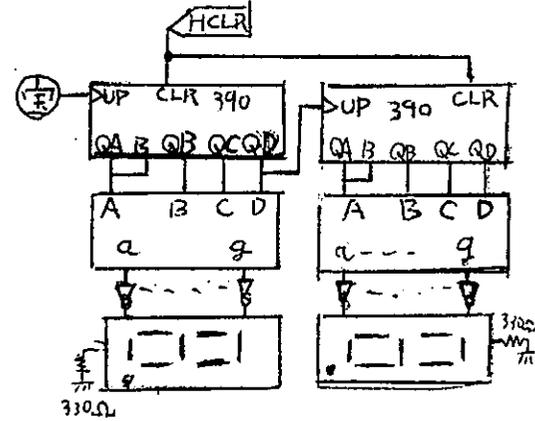
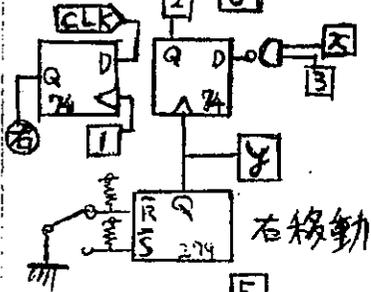
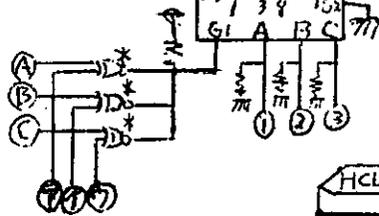
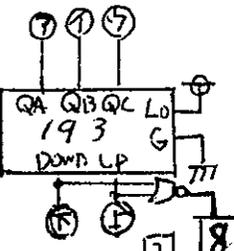
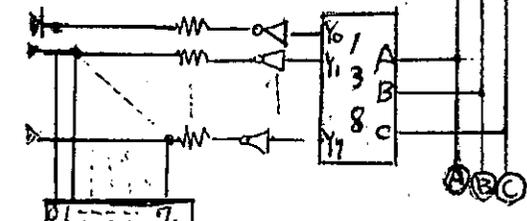
「ここから逃げけるんだよ...」

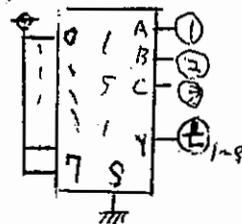
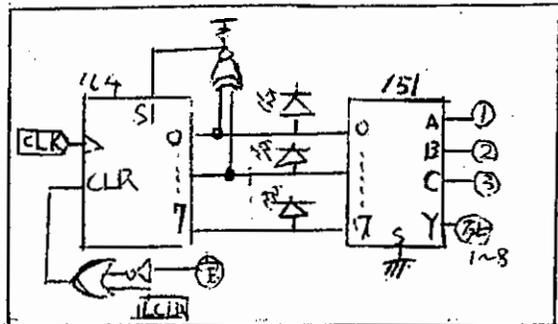
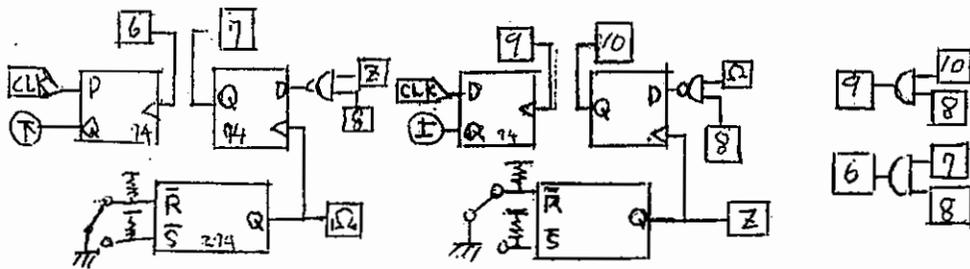
TO BE CONTINUED ➡

⑧ 回路図 ※基本抵抗は1kΩ以下

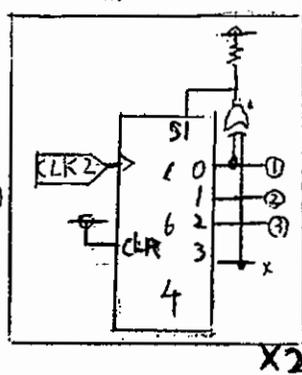


ダイナミック点灯

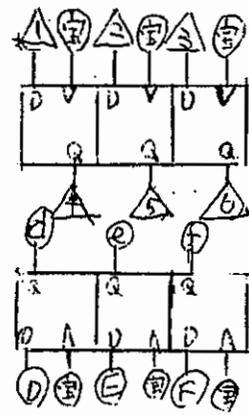
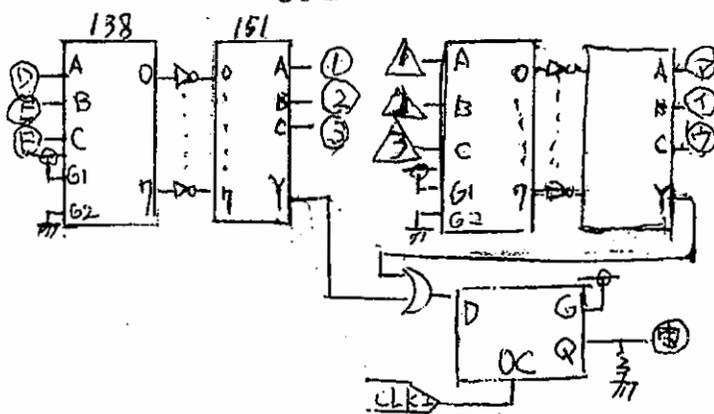
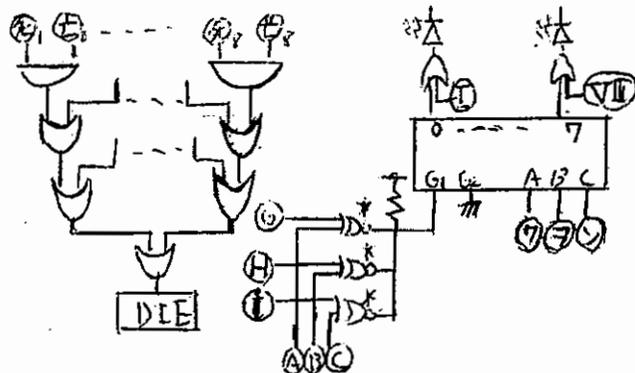




X8



X2



サーティ ミニット

30 minute

製作：中岡 (M)
設計：大塚 (H)
協力：物無の方

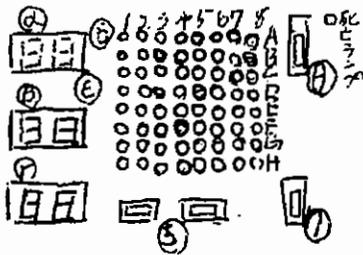
～ストーリー～

1945/8/16、海の上に一人の青年がいた。彼の
 名は、杉本。約半年前の空襲で失われる物はほとんど
 ど失ってしまった。もちろん家族も失った。そ
 の復讐のために出たのがこの航海のハズだった。が
 彼は、意欲を失っていた。それは、先程入手した
 情報によまものであった。「日本が敗戦した」とのこ
 とであった。彼は今、席リこみ過去の思い出にひた
 っていた。妹が生まれた時、小学校の発表会、中学校
 での演劇部、...家族をまだ燃え火の中から引
 出そうとしたこと...
 彼は下からの鈍い衝撃で我に帰った。船が下から
 の攻撃に当たってしまったようだ。彼は活気づいた
 かのように状況の確認をした。底には大きな穴が空
 いており修復は不可能。持って30分というところだ
 弾は残り3弾だが、敵を潰せば残骸から2発の弾が
 とれる。問題は無い。そして彼は下もむうに下に弾
 を投げ始めた。その目は復讐に燃えていた。

“残り時間は30分。その間に彼は
 どれだけ道連れにできるのでしょうか?”

※このストーリーはフィクション
 であり実在した出来事ではありません。

〜ゲーム説明と外観〜



各部位について

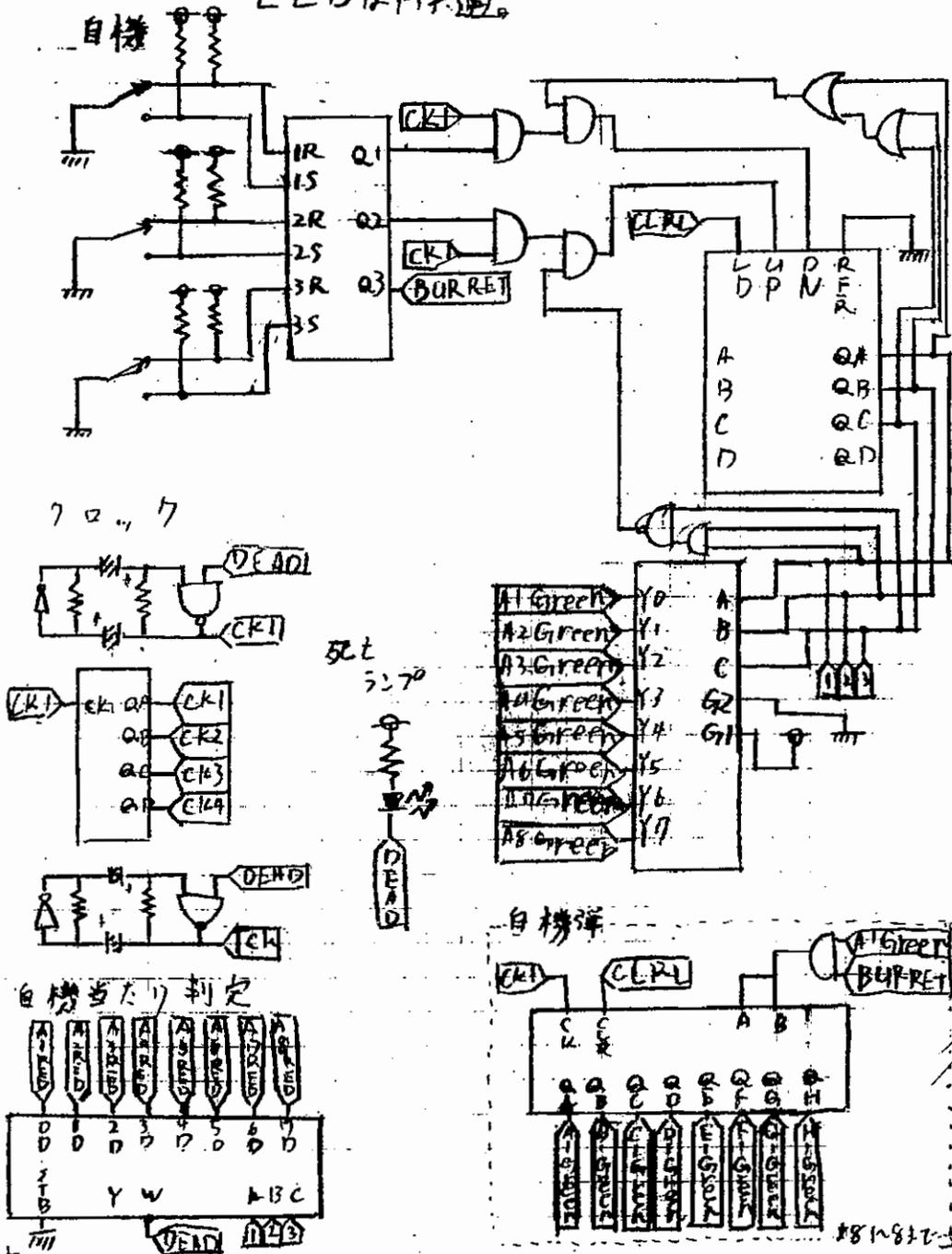
- ① 時間
ストーリーでは30分でしたが、ゲーム上30秒です。0になると船が沈みゲームオーバーです。
- ② 残弾
残りの弾です。はじめは、3で敵を潰すごとに2ずつ回復します。0になると死を待つのみです。
- ③ スコア
いあろきル敵数というやつです。0から始まり敵を潰すごとに1つずつあつまります。
- ④ 自機 (緑)
杉本くんはこの列(列)の中のめうごけます。下から来る敵の弾に当たると一撃で沈みます。つまり当たるとゲームオーバーとなります。
- ⑤ 敵機その他 (敵機赤) (敵弾赤) (敵の弾緑)
杉本くんを攻撃する者が通ります。又、敵の弾、自分の弾などもここを通ります。
- ⑥ 自機移動用スイッチ
これを使い敵の弾をおけたりします。
- ⑦ 弾投下スイッチ
弾を投下します。
- ⑧ クリアスイッチ
杉本くんが我に帰ったところまで時を戻せます。

ゲームのルール

自機を左右に移動させ敵の攻撃をおけながら敵を攻撃するというアクション(?)ゲームです。
敵の攻撃が当たると、時間が切れるとゲームオーバーとなります。
弾は3発しか持っていません。投下すると減り、敵を潰すと2発回復します。
敵の弾は潰せません。

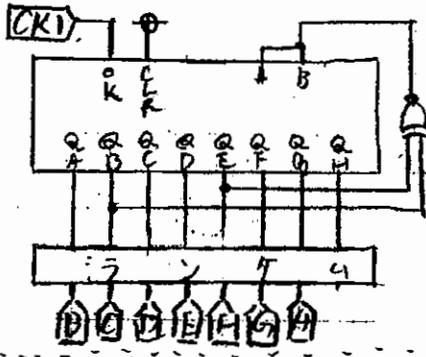
~回路図~

LEDはH共通。

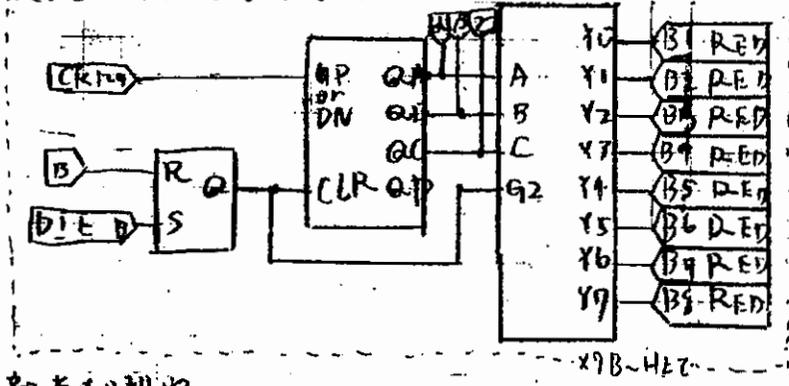
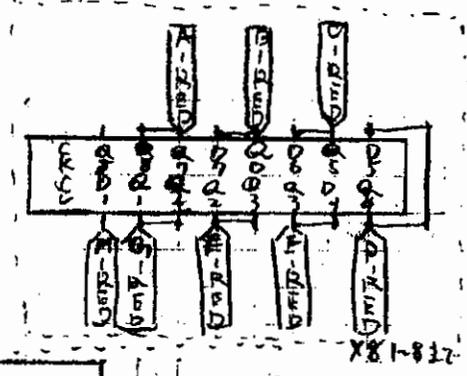


回路図 (続き) ~

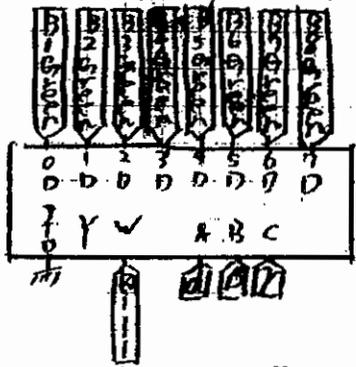
敵



敵機

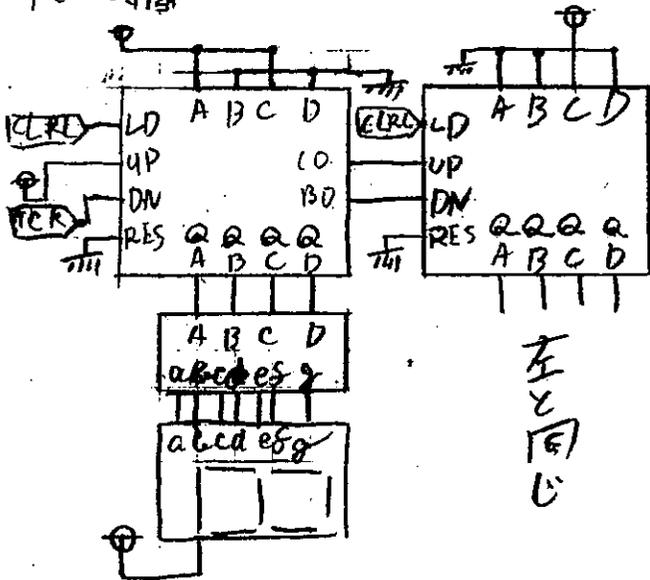


敵当り判定



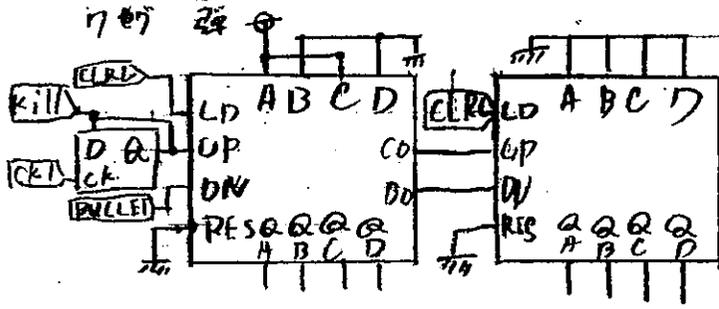
~回路図(続)~

7セグ時間



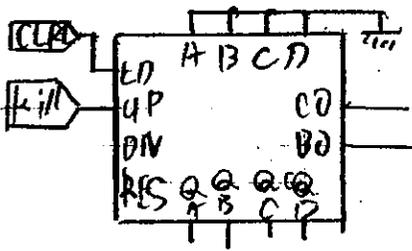
左と
同じ

7セグ



以下時間用の7セグ:同じ

7セグ スコア



この他
別の7セグと
同じ

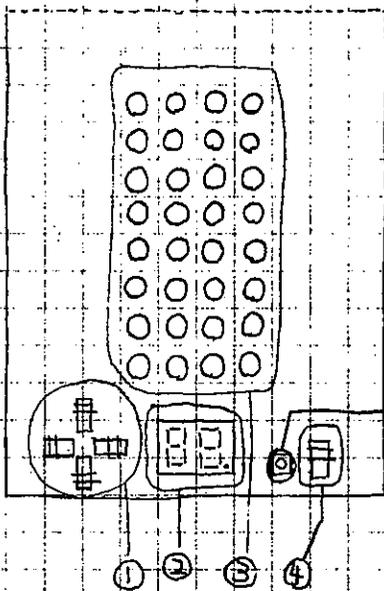
子口一ツを交わせ

制作者 M2 中島 裕人

回路設計者 H1 木村 けん

協力して下さった人 木村けんをはじめとする
物無の方マ

～ 外見 ～ (操作部と画面の他は省略)



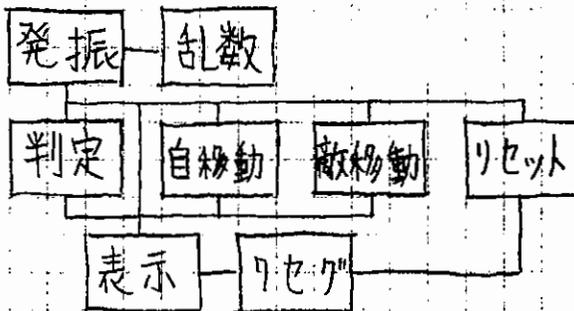
※比率は異なります

- ① 上下左右スイッチ
(自機を上下左右に動かします。
一番右(本)まで行くと、左(本)にワープできます。
自機は下の16マス(4×4)の内側で操作できます。
こわねないよう、対金が巻いてあります。
- ② 敵をたおした回数(得点)
- ③ 画面(LED)
- ④ 弾を出すスイッチ
- ⑤ リセット

～ルール～

- ・自機はスイッチを使い、下半分(4×4)を自由に移動できます。
- ・敵機も上半分を移動します。
- ・スイッチを使い、玉を銃つことが出来ます。
- ・自機に玉が当たると死亡、リセットして下さい。
- ・敵機に玉が当たると、点が入り、敵が強くなります。

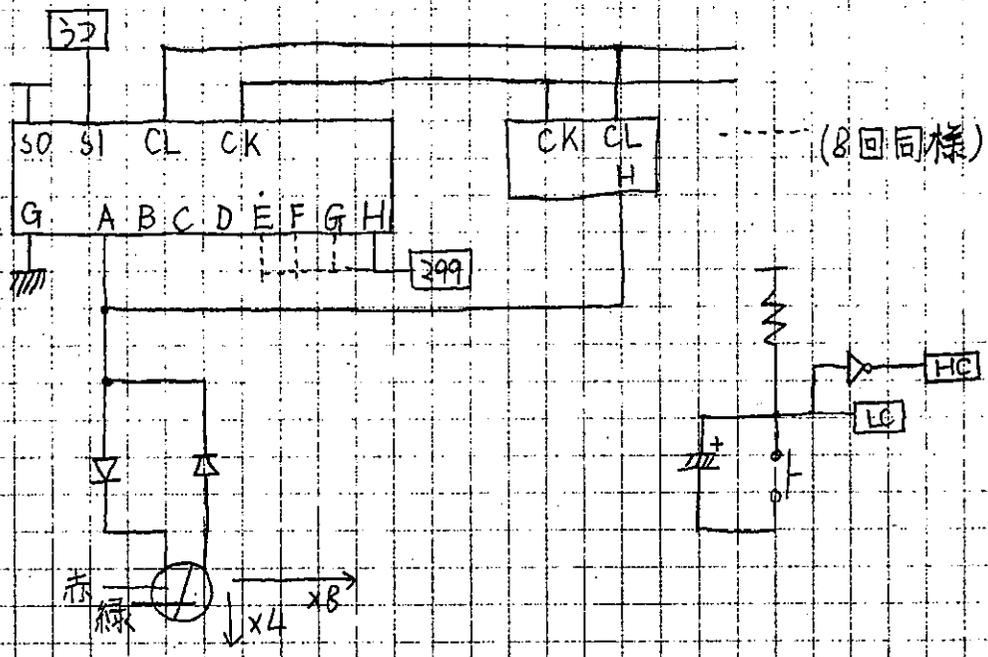
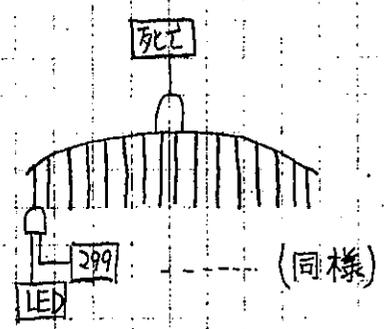
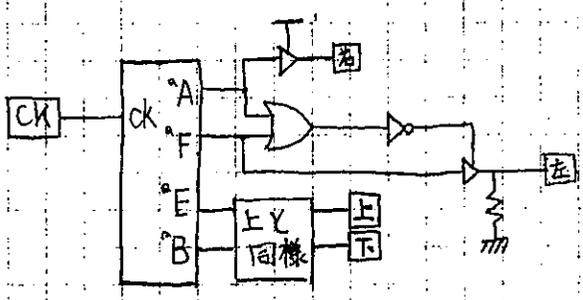
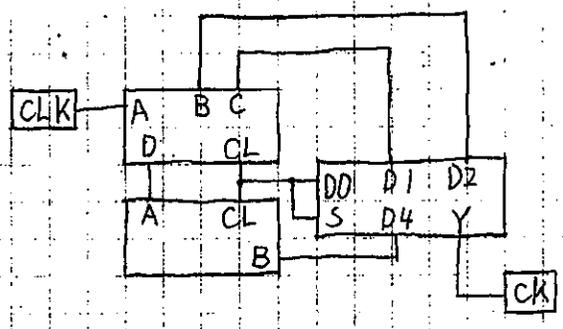
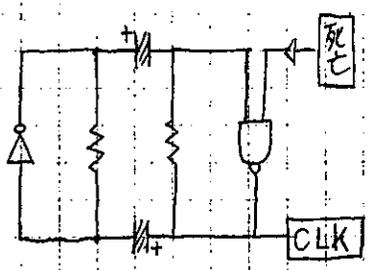
ブロック図

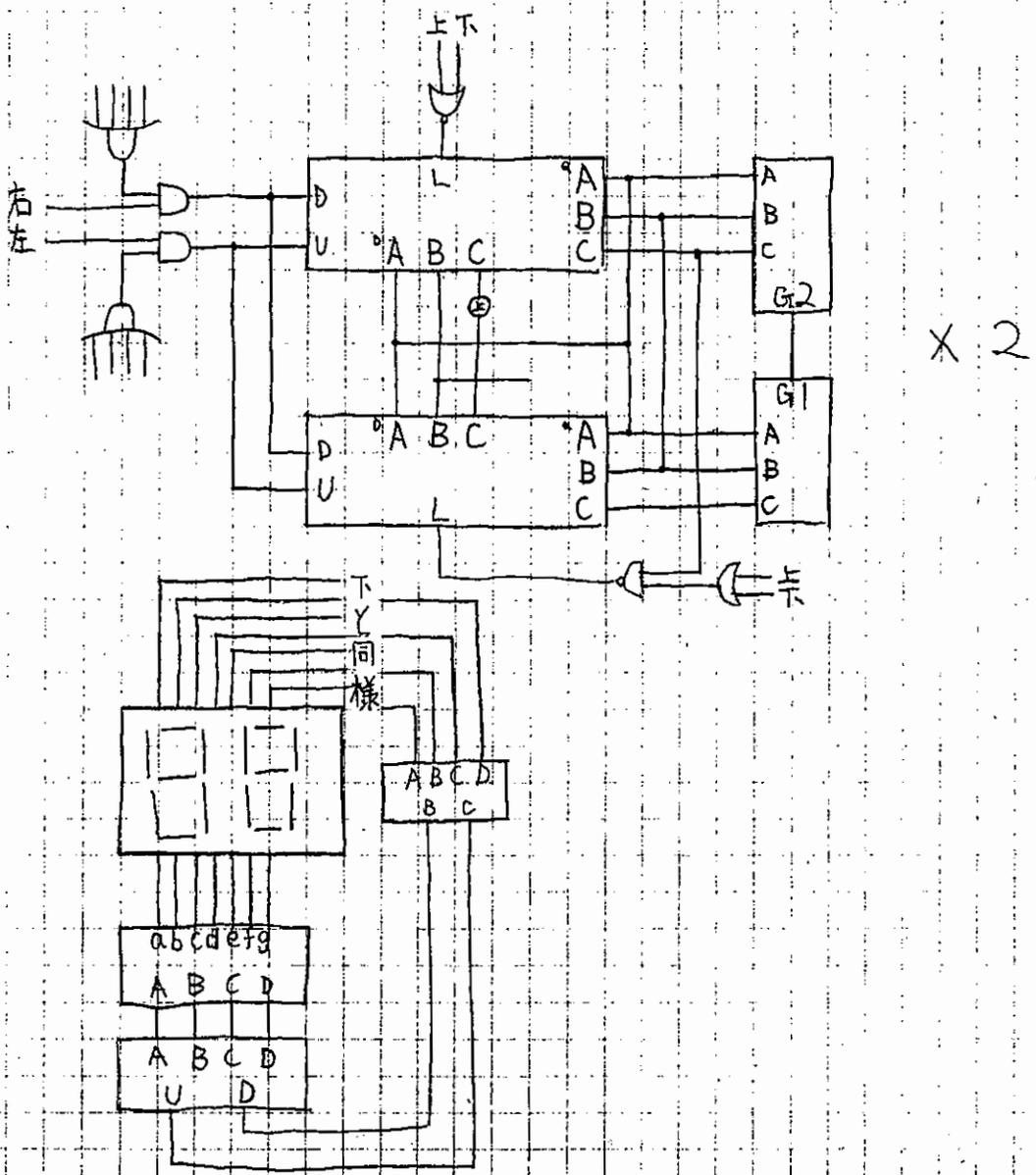
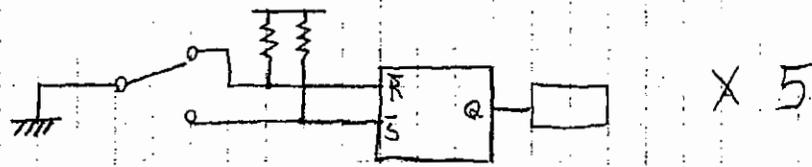


～ストーリー～

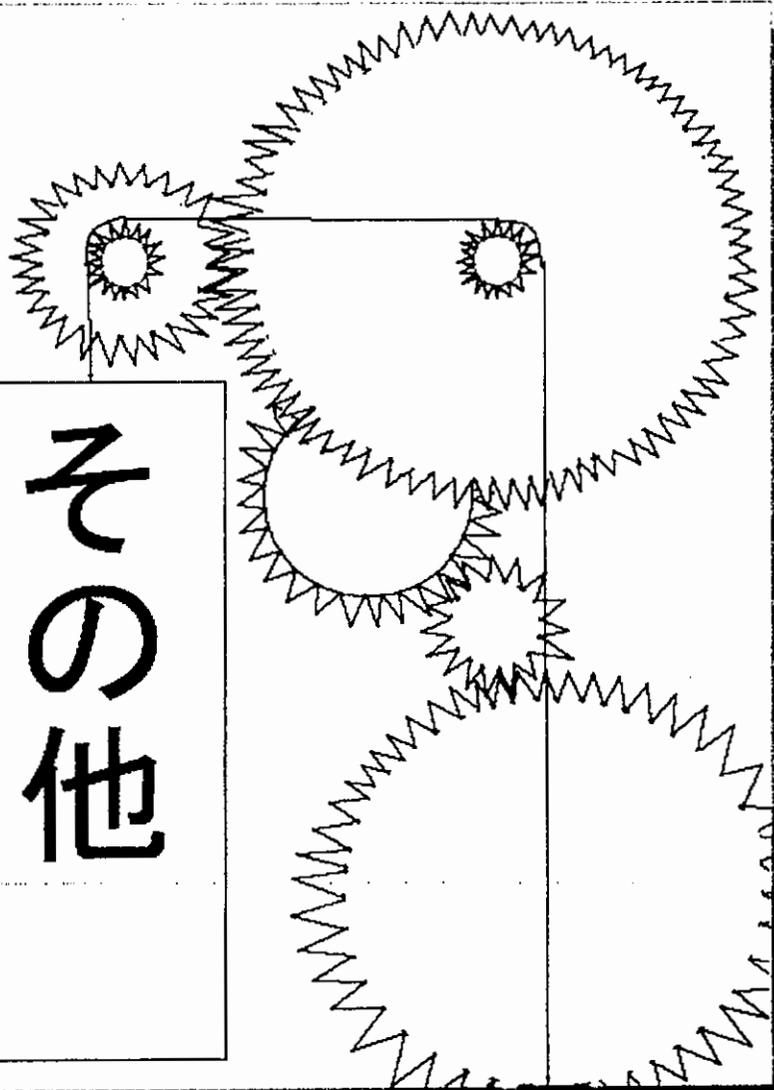
例会中、井上さんが「投げ」るチョコクを交わしましょう。

~ 回路図 ~





その他

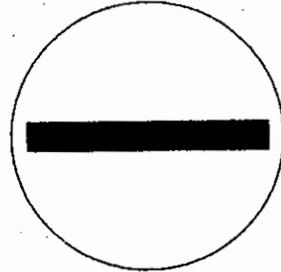


売り物

100YEN

500YEN

コイン投入口



返却



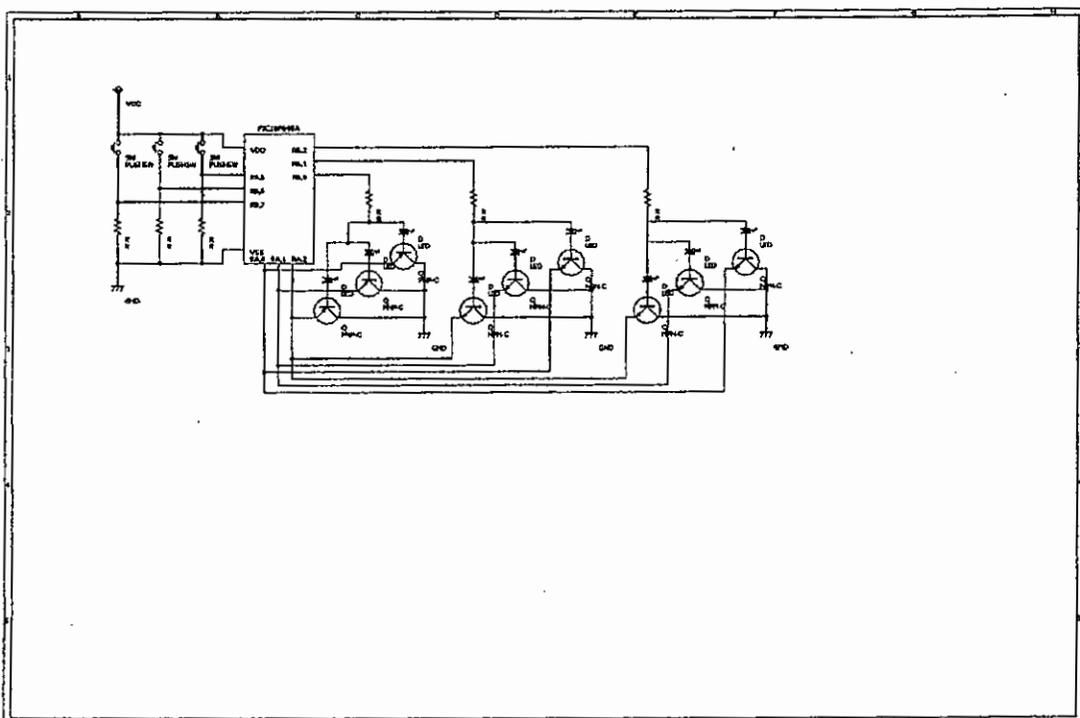
太古の廃人

製作者:M3-7 渡邊翔

このゲームは要するに、下図の右から流れてくる玉を一番左のところで重なった瞬間にスイッチを押し、キャッチするというゲームです。2 段あるので、上下のボタンで上下の列の玉をキャッチしてください。成功するとあたり判定が入りますが、キャッチしたときにとっていないと死亡になり、初めは五個ある時期が一つ減ります。全てなくなるとゲームオーバーです。ちなみに一定量以上とるとスピードが多少速くなるようになっています。ゲームオーバーのときはリセットスイッチを押しやり直してください。

==回路図==

* 一部 LED 部分を 5 個から 3 個に省略しています



回路についての簡単な説明をすれば、要するに3つのスイッチ(キャッチかける2、リセット)をみて、それでは後はひたすらダイナミック点灯で光らせるという単純なものです。プログラムのほうは長すぎるため、別途配布という形になっております。

* 注意:この回路、プログラムどおり作っても動かない可能性がございます。あらかじめご了承下さい。

麻布学園物理部無線班回路図集

2007年4月10日発行

編集責任者 : 井上 忠明
発行者 : 麻布学園物理部無線班
印刷責任者 : 竹島 啓純
印刷所 : 麻布学園第二応接室
表紙デザイン : 木村 朴
その他デザイン : 西村 陽樹
早川 潤
池上 涼平

乱丁・落丁はお取替えいたします。

また、この回路図集について質問等があれば麻布学園物理部無線班までお願いいたします。

掲載されている回路図やプログラムの通りに作っても動かない可能性があるので、参考程度に参照ください。

無断転載・複写はお断りします

And they're buying a stairway to heaven.



B U T U B U



M U S E H A N

㊦ 出版

本書は麻布学園物理部無線班の部員が1年間かけて
研究、製作したものについて記されています。
しかし、未完の物もある為、書いてある通りに作っても
上手くいかない場合があります。御了承ください。

定価：本体100円(税込み)