

# 制御用小型コンピュータによる模型制御

## 1 目的

制御用小型コンピュータによる模型制御を通じて、コンピュータによる機器制御の基礎を学習する。実際に模型を動かしながらハードウェアとソフトウェアの関係を学び、情報系のものづくり技術を体得する。

## 2 概要

H8/3664は制御用の小型コンピュータであり、内部に命令用メモリ(32KB)、データ用メモリ(2KB)、CPU、入出力ポート6系統等が収められている。パソコンで開発したプログラムを、シリアル通信によりH8/3664に転送し、そのプログラム動作で、入出力ポートを自由に制御することができる。

今回の実験では、左右2つのモータと反射型の光センサーをつけた模型を、H8/3664で制御し、白い床の上に描かれた黒い線の上を走行するプログラムを作成する。各実験者が作成したプログラムを走行させ、コースの周回に要した時間を計測する。

### 2.1 実験機材

実験で使用する機材は、以下の通り。

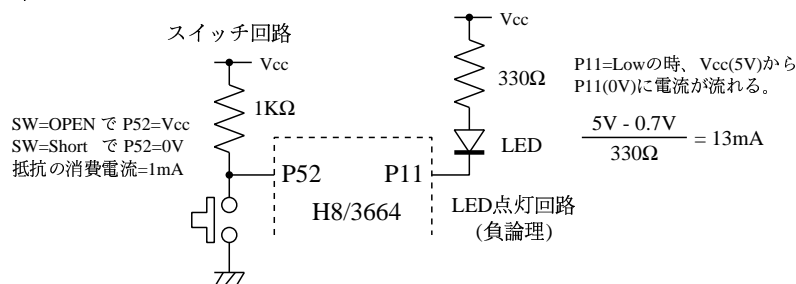
1. H8/3664CPUボード
2. ブレッドボードとジャンパ線
3. 反射型光センサ WPS-01
4. モータ制御IC(TA7291P)
5. 左右2モータによる模型シャーシ

## 2.2 ルール

1. スタート時、車体はスタートラインを越えないこと。車体全部がゴールラインを越えた時点でゴールとする。
2. コース周回中は、黒い線の上に車体の一部が乗っている事。コースをはずれた場合、時間の計測は中止する。
3. 黒い線は、センサーの間隔に応じて幅 60mm 以内で変更して良い。線幅の中心は変えないこと。
4. 時間は右まわり・左まわりそれぞれを測定し、その最大値と合計値を記録とする。右まわり・左まわりの走行の間にプログラムや配線の変更は NG とする。
5. 基本的には準備した実験の部品を使って作るが、性能向上のために個人で用意した部品に交換しても良い。

## 3 基本となる回路とプログラムの説明

H8/3664 の動作確認をするための基本プログラムとして、スイッチから入力した High/Low に応じて LED を点滅させるプログラム<sup>1</sup> を補足資料 A に示す。



配付資料で不足している情報については、実験資料として、

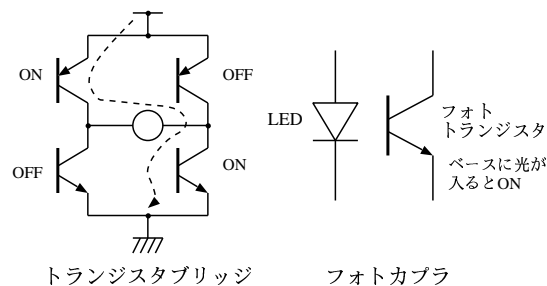
<http://www.ei.fukui-nct.ac.jp/t-saitoh/exp/h8/>  
にて公開しているので、参照すること。

<sup>1</sup>サンプルでは、ビット演算のプログラム例とするため、P11,P52 といった統一性の無い端子を用いている。プログラムを作る際には、判りやすい端子を使用すること。

## 4 実験で用いる周辺ハードの説明

### 4.1 モータ回転方向制御 IC

模型用の直流モータの回転制御では、次の注意が必要となる。(1) モータ回転開始時にコイルが低い抵抗状態で大量の電流が流れ、電源の電圧低下が発生し誤動作が起こる。(2) モータ整流子の接触時に発生する誘導起電力によるノイズによる論理回路の破壊や誤動作を防ぐ必要がある。このため論理回路とモータ用の電源や回路の分離が重要となり、一般的にフォトカブラが使われる。



回転方向の制御では一般的に図に示すようなトランジスタブリッジが使われる。

これらの回路は全体では複雑になるため、実験ではこれらを集積化したモータ制御 IC(TA7291P) を使用する。(補足資料 Bを参照)

### 4.2 反射型光センサー

実験で用いる光センサーには、モータ制御と同じようにフォトカブラを使用する。反射型のフォトカブラでは、LED の光を対象物にあて、光の反射量で変化するフォトトランジスタの電流量を用いる。実験で使用する WPS-01 (補足資料 B参照) は、センサー光を黒い物にあてると High, 白い物で Low となる。センサーの設置状況で閾値が変化するのでセンサー付属の抵抗を調整すること。

H8 には A/D コンバータ (アナログ値をデジタル値に変換する機能) が内蔵されているので、反射光の量を 2 値情報でなく微妙に変化する明るさの情報として読みとることもできる。

## レポート作成での補足

1. レポート作成時には、ルール・補足資料は記載しなくても良い。
2. レポートには、H8, センサー, モータ周辺の配線を記載せよ。
3. タイヤの位置やセンサーの位置や間隔も記載せよ。
4. プログラムの主要部分のリストを添付せよ。

## 5 実験方法

1. 章3で示した回路を配線し、スイッチ操作に応じてLEDの点滅が変化するか確認し、プログラムの意味を理解せよ。
2. 前記回路の入力を、章4で示した、反射型光センサに入れ換え、床の白床・黒線に正しく反応する様に、センサを調整せよ。
3. ライントレース動作をするために必要となる機械構成を、自分独自のアイデアを交えながら作成後、各センサ・モータが正しく動作することを確認せよ。
4. 前記ハードに対応したラインレース動作のプログラムを実行し、ルールに示した方法で計測を実施せよ。

## 6 考察

1. センサーの位置や間隔、モータ電圧、プログラム中の処理時間に影響するような様々なパラメータを変化させた時に、模型の動きにどういった影響が発生したか、考察せよ。
2. モータやセンサーといった非論理回路と、論理回路が混在する場合に発生する問題点を文献を調査し、その対策をこの模型制御に取り入れるにはどうすれば良いか説明せよ。

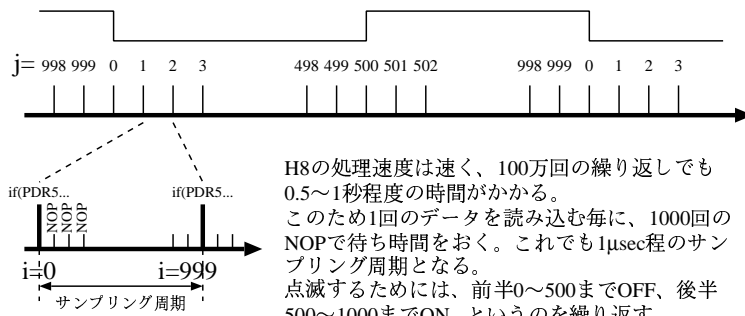
## A 基本プログラム

```

#include "3664.h"
/* P11をLEDに接続、P52をSWに接続 */
#define P1OUT 0x2 /* P11:0000,0010 */
#define P5IN 0x4 /* P52:0000,0100 */

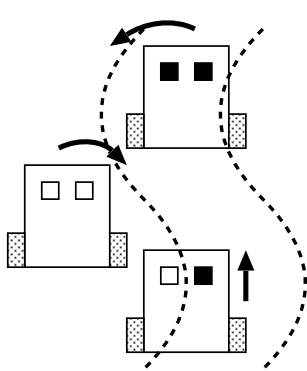
/* P52のスイッチがONの間、P11のLEDが点滅するプログラム */
int main()
{
    int i, j; /* モータのPWM制御にも使えるサンプル */
    /* データシート 9.1 ポート1を参照 */
    PMR1 = 0x00; /* 汎用入出力ポートに設定 */
    PCR1 = 0xff; /* ポート1を全ビット出力用に使用 */
    PDR1 = 0x00; /* ポート1をクリア */
    /* データシート 9.3 ポート5を参照 */
    PMR5 = 0x00; /* 汎用入出力ポートに設定 */
    PCR5 = 0x00; /* ポート5を全ビット入力用に使用 */
    j = 0;
    while( 1 ) {
        if ( (PDR5 & P5IN) == 0 ) { /* &周りの()は必須 */
            /* P5INがONの時 */
            if ( j >= 500 )
                PDR1 = P1OUT; /* jが500~999ならP1OUTをON */
            else
                PDR1 = 0; /* jが0~499ならP1OUTをOFF */
        } else {
            /* P5INがOFFの時 */
            PDR1 = 0;
        }
        /* 1msecほどの処理 */
        for( i = 0; i < 1000; i++ )
            NOP();
        /* 0~999の値を繰り返す */
        if( ++j > 1000 )
            j = 0;
    }
}

```

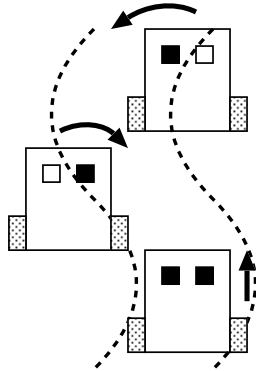




## D ライントレースの考え方

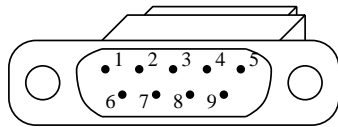


黒と白の境界をまたぐ場合  
線の太さは右の方式よりは  
細くても大丈夫。



黒の線の内側を走る場合  
線の太さは2つのセンサの  
幅より少し太めの必要がある。

## E シリアルコネクタのピン配置



配線に注意せよ！ シリアルケーブル  
と接続する端子を正面にして、左上か  
ら1番ピン。

## F 補足資料

以下に、実験の開発環境に依存する情報だけを、補足資料として抜粋した。

### F.1 雛型プログラムのコピー

電子情報演習室の環境は Z: ドライブはファイル共有で自分専用のディレクトリとなっている。Z: ドライブ配下に演習用のディレクトリを作り、雛型となるプログラム (以下のディレクトリ参照) を

¥¥10.120.11.60 ¥home ¥ei ¥t-saitoh ¥h8 ¥h8sample ¥h8car

まとめてその自分のディレクトリにコピーせよ。

編集するプログラムは、**h8car.c** である。

### F.2 H8 プログラムのコンパイル手順

デスクトップの Cygwin というショートカットもしくは、スタートメニューのプログラムの中にある “Cygwin → Bash Shell” を起動し、コマンドラインから命令を入力できる状態にする。

\$ cd Z:/ 演習用ディレクトリ /h8car

\$ ls -al                   ディレクトリ内容の確認

\$ make                   必要なコンパイル作業を自動実行

### F.3 H8 プログラムの転送

1. スタートメニューのプログラムの中にある H8write を起動
2. CPU の種類を H8/3664F を選択
3. CPU ボードの JP2,JP3 を Short にする。
4. CPU ボードのリセットを押す。
5. Z:/ 演習用ディレクトリ /h8car/h8car.mot を、ドラッグ&ドロップで H8write のウィンドウに入れる。
6. “OK” の確認によりプログラムが転送される。
7. CPU ボードの JP3 を抜き、リセットを押すとプログラムが起動する。