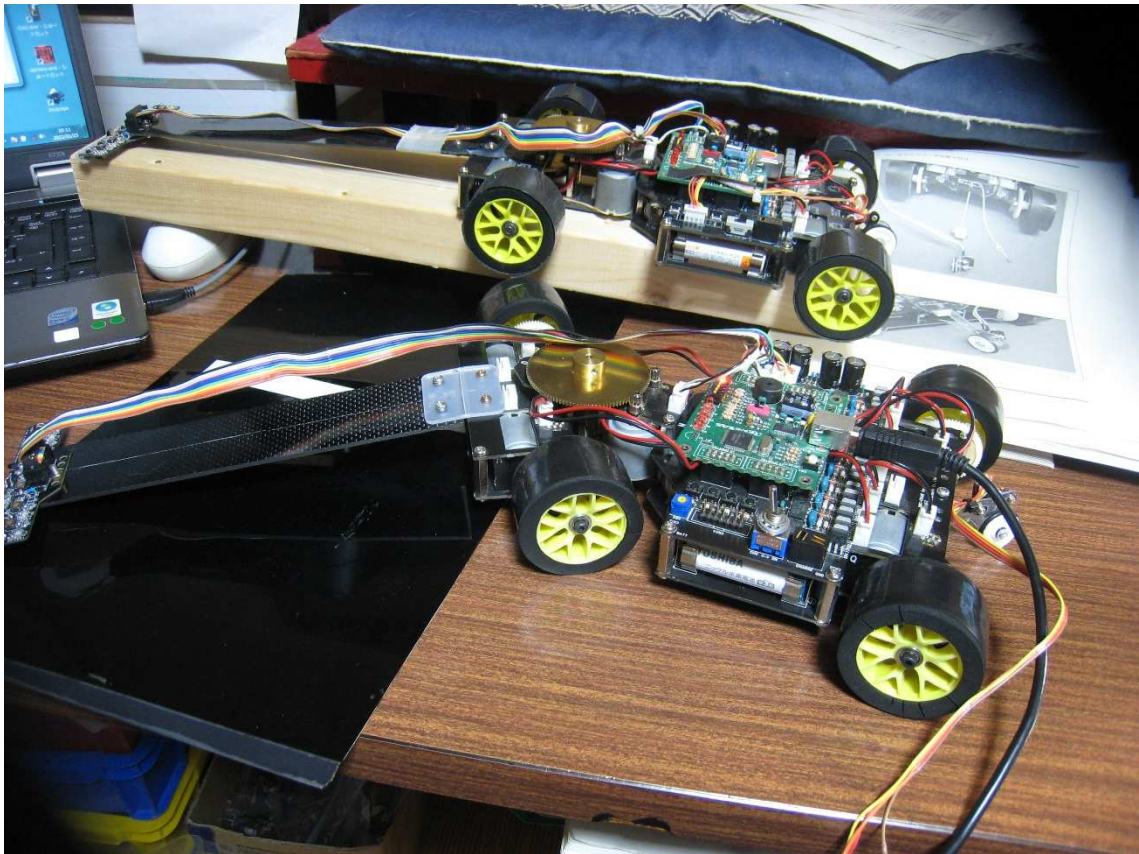


R 科応援

寒河江工高 電子機械科(R科)は長年マイコンカーを教材として取り組んでおり、主に Camera Class と Basic Class マシンを製作しているとのこと。毎年2月校外での独自大会を開催しており、寒河江市報掲載されるなど大いに盛り上がっています。今年度はマイコンカー山形県事務局も担当していることから、A Class 製作を希望する生徒もあり、今回はR科の応援で1次完成したマシンの動作確認をしました。

マシンは情報技術科同様、R8c/35a マシン(基板マイコンカー)です。

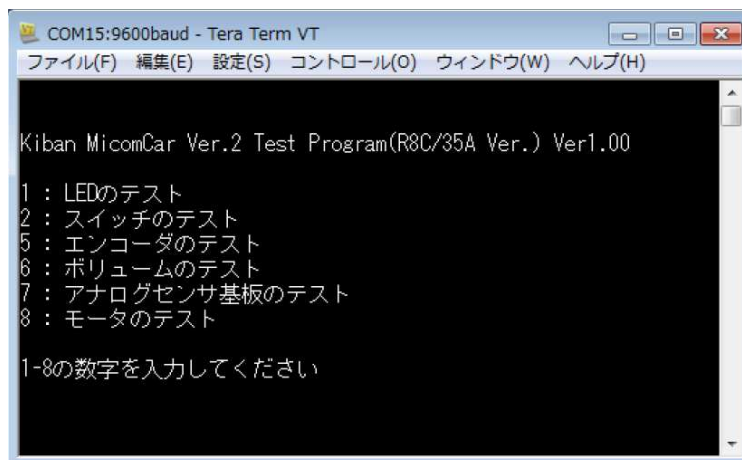


動作結果

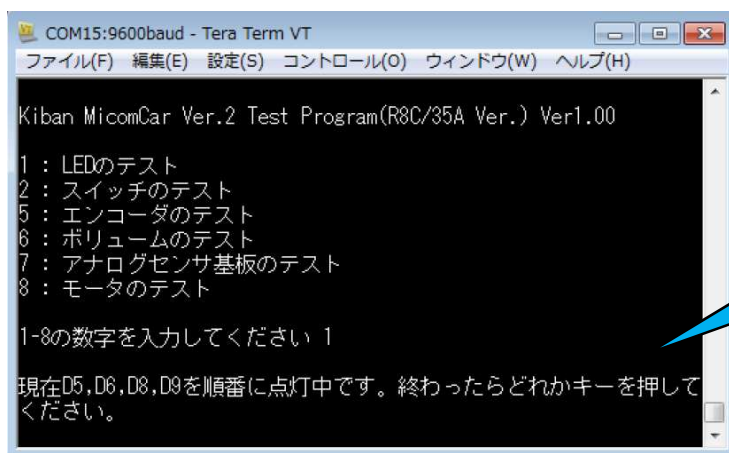
・不具合点

R8c/35a 増設 LED 不備	(抜け)
ボード改造不備	(D7 に 4.7k Ω 抵抗取り付け)
左前輪取り付け不備	(ジュラコンスペーサ抜け)
前輪取り付け方法不備	(ダブルナットで締める)
モータ正転,逆転反対	(端子交換)
センサアーム取り付け不備	(ナイロンナット使用へ)
ロータリエンコーダ不備	(ハンダ不良?)

• 動作確認

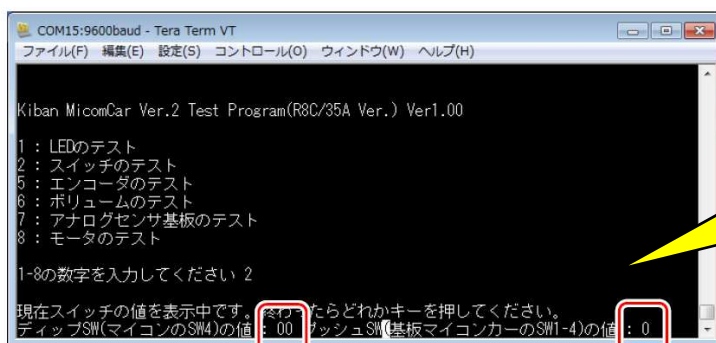


(1)LED のテスト



OK

(2)スイッチのテスト

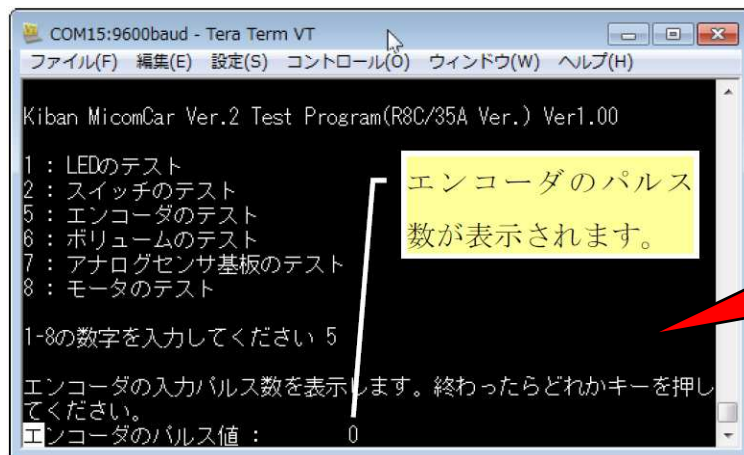


SW4,SW3,
SW1 OK
SW2 ×

SW4の値が16進数
で表示されます。

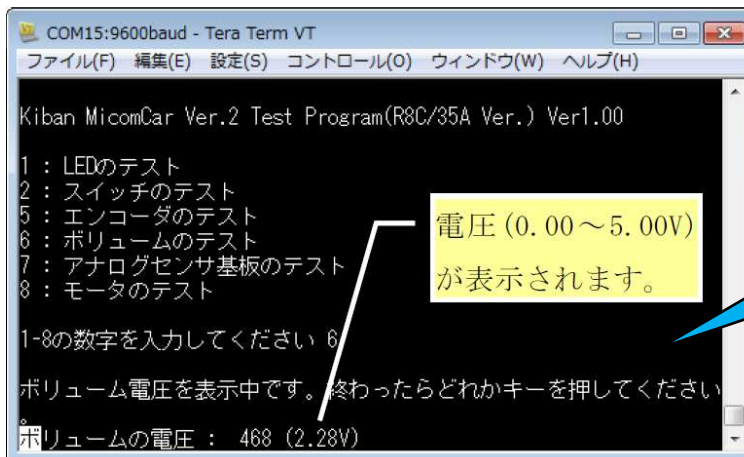
SW1～SW4 を押すと 1、離すと 0 と表示されます。

5.ロータリエンコーダ



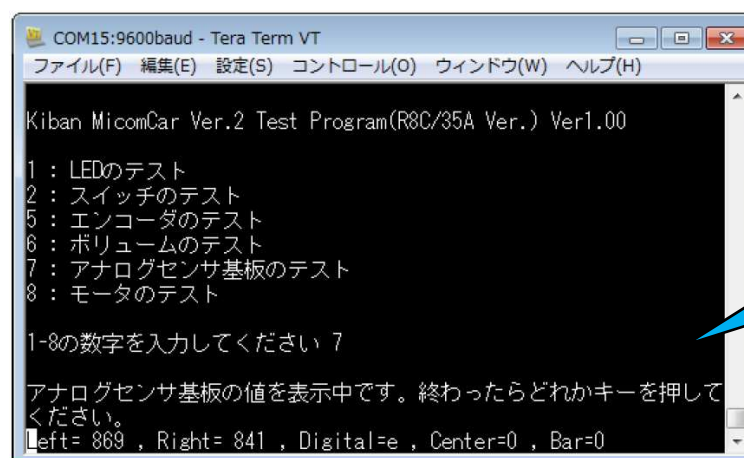
不具合発生

6.ポリユーム



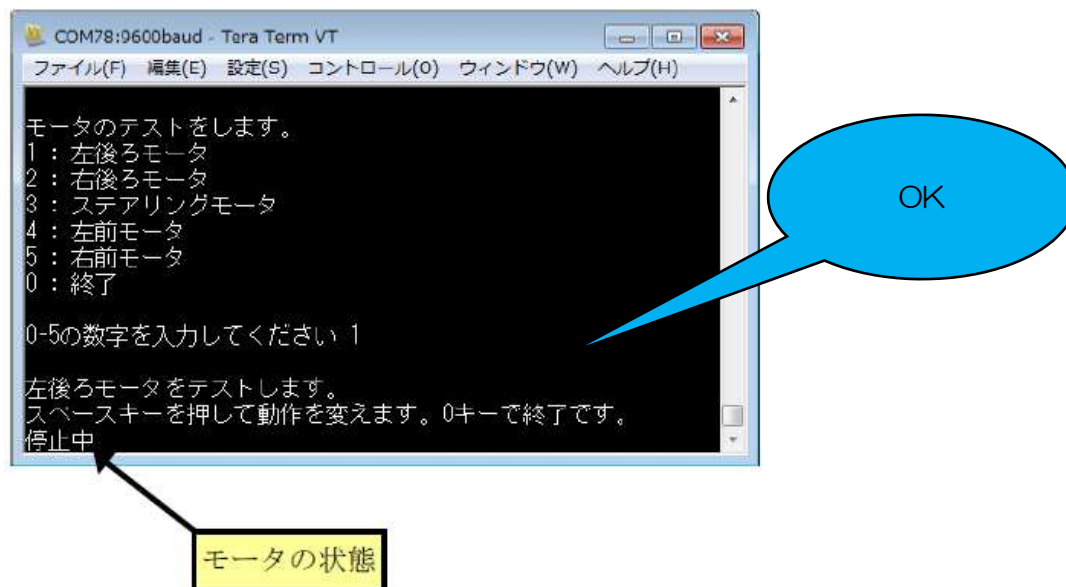
OK

7.アナログセンサ基板 確認OK



OK

8.モータテスト 確認OK



現状まとめ

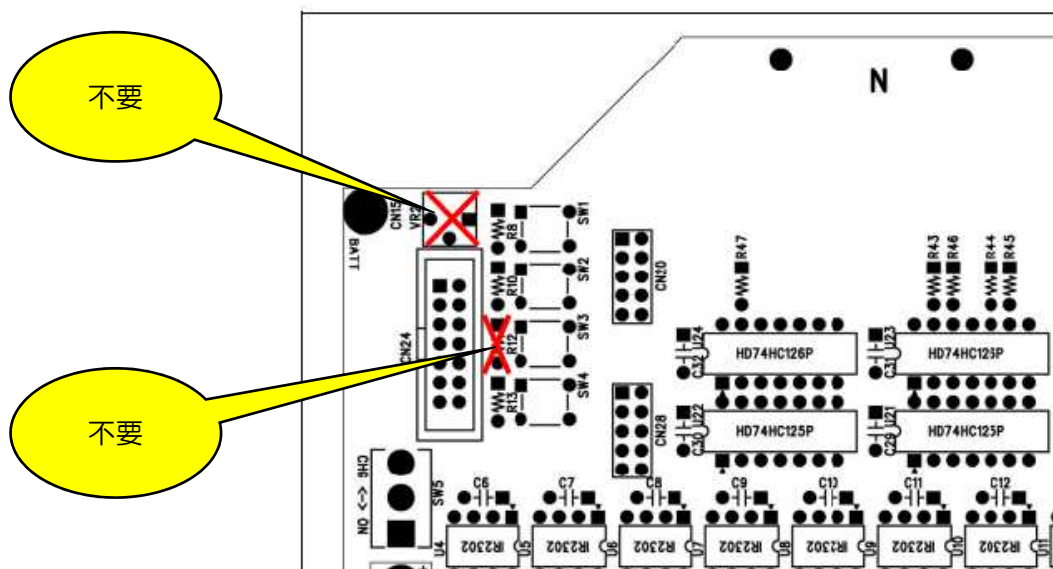
ロータリエンコーダ, SW2 以外は問題ないようです。

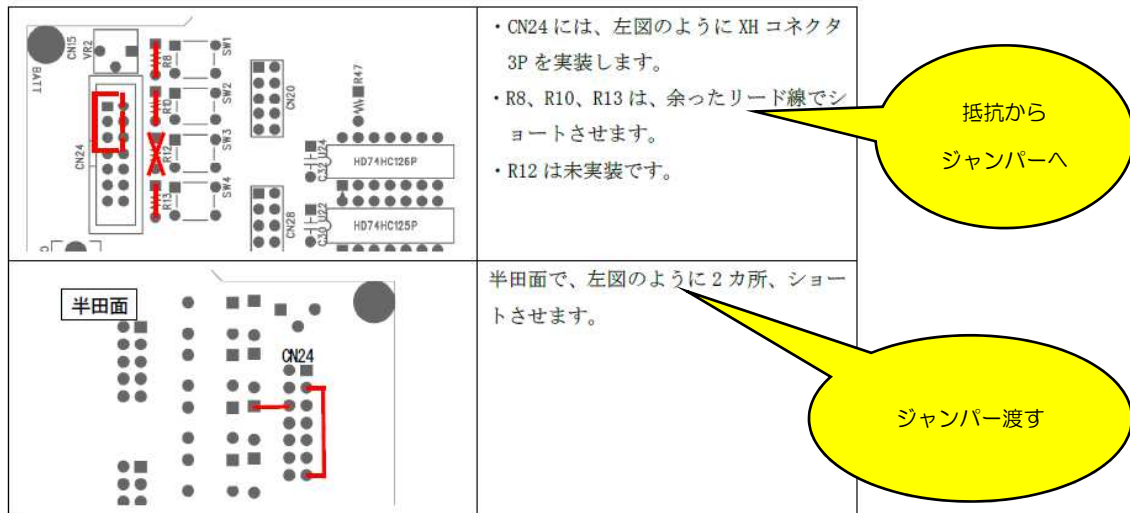
不具合対策

マザーボード製作での不備あり

「基板マイコンカー製作キット Ver.2 組み立てマニュアル」35～36P.

5.1 N基板（上側リアユニット）の組み立て





この変更で、動作確認全てOKとなりました。

メモ R8C/35a 書き込み手順

1. USB 接続
2. スライド SW を書き込み側に倒す(USB コネクタ側)
3. CPU 電源 ON

この段階で CPU ボードの赤 LED が 2 個点灯し通信可能となります。

• サンプルプログラムでトレースする前に

(1) アナログセンサの値確認

(2) ボリュウムの値確認 「基板マイコンカー製作キット Ver.2 プログラム解説マニュアル」86P.

ハンドルまっすぐ時、ボリュームの値が 2.23V 付近になるよう軸を調整して下さい。

基板マイコンカー製作キット Ver.2 プログラム解説マニュアル

5. プログラムの解説

5.2.26 ステアリングモータ角度の取得

```

782 :  /***/
783 :  /* サーボ角度取得 */
784 :  /* 引数 なし */
785 :  /* 戻り値 入れ替え後の値 */
786 :  /***/
787 :  int getServoAngle( void )
788 :  {
789 :      return( ad5 - iAngle0 );
790 :  }

```

ステアリングモータの角度は、AN5(P0_2)端子に接続されているポテンシオメータ(ボリューム)の値で分かります。
iAngle0 は、0 度のときの A/D 変換値を入れておきます。例えば、角度が 0 度のとき A/D 変換値が 456 なら、
iAngle0 変数に 456 を代入すると、戻り値は下記ようになります。

```

戻り値  = A/D 変換値 - 0 度のときの A/D 変換値
        = 456 - 456
        = 0

```

基板マイコンカーVer.2 は、実測で左右 40 度ずつハンドルが切れました。中心と左右最大にハンドルを切ったときの電圧を測ります。テストでポテンショメータの電圧(CN3 の 1 ピンと 3 ピン間)を計った結果、

左いっぱい…2.85V 中心…2.23V 右いっぱい…1.61V

となりました(下左図)。5.00V(電源電圧)が 1023 なので、それぞれの電圧を A/D 値に変換すると、

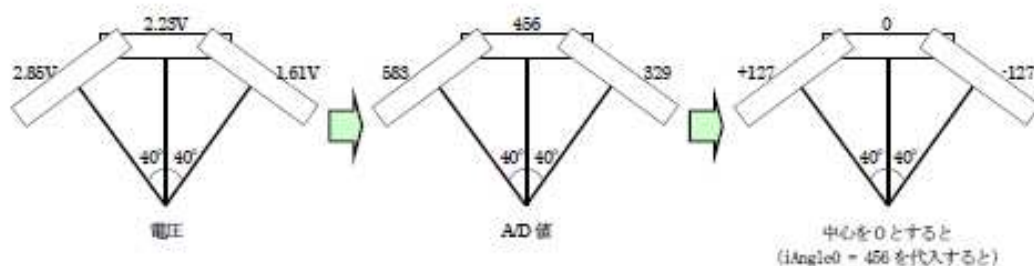
左いっぱい… $2.85/5 \times 1023 = 683$ 中心… $2.23/5 \times 1023 = 456$ 右いっぱい… $1.61/5 \times 1023 = 329$

となります(下中図)。

787 行の iAngle0 変数には、0 度ときの A/D 値を入れておきます。iAngle0 変数に 456 の値を入れると、

左いっぱい…+127 中心…0 右いっぱい…-127

となります(下右図)。



(3) Kd,Kp の値 「基板マイコンカー製作キット Ver.2 プログラム解説マニュアル」 89P.

5.2.28 ステアリングモータ制御

```

848 : /******
849 : /* サーボモータ制御
850 : /* 引数 なし
851 : /* 戻り値 グローバル変数 iServoPwm に代入
852 : /******
853 : void servoControl( void )
854 : {
855 :     int i, iRet, iP, iD;
856 :     int kp, kd;
857 :
858 :     i = getAnalogSensor(); /* センサ値取得
859 :     kp = 1; /* kpを増やしていき、ブルブルが
860 :     kd = 0; /* 大きくなったら kdを増やしてく
861 :             /* ださい
862 :     /* サーボモータ用 PWM 値計算 */
863 :     iP = kp * i; /* 比例
864 :     iD = kd * (iSensorBefore - i); /* 微分(目安は P の 5~10 倍)
865 :     iRet = iP - iD;
866 :     iRet /= 64;
867 :
868 :     /* PWM の上限の設定 */
869 :     if( iRet > 50 ) iRet = 50; /* マイコンカーが安定したら
870 :     if( iRet < -50 ) iRet = -50; /* 上限を 95 くらいにしてください
871 :     iServoPwm = iRet;
872 :
873 :     iSensorBefore = i; /* 次回はこの値が 1ms 前の値となる
874 : }

```

ここが
まくいけば
必ず走れます。

この関数で、コースの中心線からセンサがどれだけずれているかを検出して、ステアリングモータの PWM 値を計算します。ステアリングモータ制御の要(かなめ)の部分です。

(4) その他 「基板マイコンカー製作キット Ver.2 プログラム解説マニュアル」 111P.

配布サンプルプログラムには、servoControl2(),車線変更ルーチンも追加されています。
ここからは各自のマシンにあわせ調整して下さい。

8. プログラムの調整ポイント

本プログラムは、ゆっくり(1m/s 程度)で走行できるようになっています(レーンチェンジ部分は除く)。速度を上げたときの調整ポイントを解説します。

行	内容	説明
89~99	内輪の PWM 値	今回のプログラムは後輪駆動として、内輪差を計算しています。基板マイコンカー Ver.2 は 4 輪駆動なので、「角度計算(4WD 時)」を使用し、4 輪の PWM 値を計算して motor 関数で設定すると、4 輪が角度に合った回転数になります。
155~170	ステアリング角度と駆動モータの PWM 値	今回のプログラムは、角度が①35 以上 ②5~35 ③-35 以下 ④-5~-35 ⑤それ以外(-5~5) で駆動モータの PWM を調整しています。もっと細かく調整してみましょう。 また、速度が速いときの 5~10 度、遅いときの 5~10 度、というように速度も使って PWM 値を調整してみましょう。
193	クロスライン検出時のスピード	今回のプログラムでは 1m/s にしています。それ以上速くても直角を曲がれるか実験して、曲がれるならその速度にしてみましょう。 また、1m/s 以上の速度なら PWM を 0% にしていますが、クロスラインから直角までの距離が 50cm だと、減速しきれません。PWM 値をマイナスにして逆転ブレーキをかけてみましょう。
217 241	右クランク検出時のステアリングを曲げる PWM 値、 左クランク検出時のステアリングを曲げる PWM 値	右クランク検出時、右にステアリングを曲げる PWM 値を設定します。今回は 50% です。弱いと曲げるスピードが遅くなり、外側にくらんで旋回してしまいます。また 40 度になるとステアリング部分が車体にぶつかって、ステアリングモータがロックしてしまいます。ぶつかる直前にステアリングモータの PWM を弱めましょう。 左クランク検出時も同様です。
218,219 242,243	右クランククリア時の PWM 値、 左クランククリア時の PWM 値	計算値では、左前:60%、右前:35%、左後:45%、右後:24%ですが、実際は違う値の方が右クランクが曲がりやすいかもしれません。いろいろな PWM を試してみましょう。例えば、内輪の後輪の PWM を小さくした方が曲がりやすいかもしれません。 左クランククリア時も同様です。
869,870	ステアリングモータに加える PWM の上限設定	ステアリングモータに加える PWM の上限を設定しています。サンプルプログラムは、50%以上にならないようにしています。この数値が小さいと、せっかくの PD 制御もこの部分で制限されてしまうのでステアリングモータの反応が遅くなってしまいます。大きすぎると方がー、ステアリング部分が本体にぶつかってロックしてしまったときにステアリングモータに大電流が流れてしまいます。最初は 50%として、タイトレースが安定したら 90%程度にしてください。 例) 869 : if(iRet > 90) iRet = 90; 870 : if(iRet < -90) iRet = -90;