

取扱説明書

目次

1. 製品概要	1
1-1. はじめに	1
1-2. 注意事項	1
1-3. 製品の内容	2
2. 組み立て	3
2-1. ギヤボックスの組み立て	3
2-2. ギヤボックスの基板への取り付け	3
2-3. モータの配線	4
2-4. タイヤの組み立て・取り付け	4
2-5. 保護カバーの取り付け	4
2-6. シリアルケーブルの作成 (PUPPY 単体購入の方)	4
2-7. 完成・各部の名称	4
3. 開発環境の構築	5
4. 動作確認	6
4-1. サンプルプログラムの書込み	6
a.BB64E3687F,36057F,36064F,36077F の場合	6
b.BB64A7124F,7125F の場合	7
c.BB52A24xSNF,25xSNF の場合	8
d.BB32A26xSNF,27xSNF の場合	9
e.BB52A35xANF,35xCNF,35xDNF の場合	10
f.BB64A2AxSNF,2BxSNF の場合	11
g.BB80A2CxSNF,2DxSNF の場合	12
h.RY3048F-ONE の場合	13
i.HSB16C29S64NE の場合	13
4-2. サンプルプログラムの実行	14
5. 動作原理	16
5-1. PUPPY の運動方程式	16
5-1-1. 運動エネルギー	16
5-1-2. ポテンシャルエネルギー	18
5-1-3. ラグランジュ関数とラグランジュの運動方程式	18
5-2. PUPPY の制御モデル	19
5-3. 倒立走行制御系の設計	19
5-3-1. 離散時間制御モデル	20
5-3-2. 倒立走行制御系の構成	21
5-4. 倒立走行制御の実現	23
5-4-1. 状態量の取得	23
5-4-2. 制御周期の確立	23
5-4-3. PUPPY のパラメータ	23
6. PUPPY の回路	24
6-1. モータドライブ回路	24
6-2. 電流センサ回路	25
6-3. 傾斜角速度検出回路	25
6-4. モータ回転角検出回路	26
7. サンプルプログラムについて	27
7-1. H8/Tinyシリーズのサンプルプログラム	28
7-2. RY3048F-ONEのサンプルプログラム	30
7-3. SH Tinyシリーズのサンプルプログラム	30
7-4. R8C シリーズのサンプルプログラム	30
7-5. M16C Tinyシリーズのサンプルプログラム	30

1. 製品概要

1-1. はじめに

PUPPY は、(株)北斗電子が室蘭工業大学の技術指導および監修により開発した倒立制御学習用キットです。弊社 BaseBoard シリーズや、RY3048F-ONE と接続し、簡単なモータの速度制御、位置制御、トルク制御を学習できます。さらに角速度センサを搭載しておりますので倒立状態での制御も可能です。



1-2. 注意事項

- 本書を必ずよく読み、ご理解されたいうでご利用下さい。
- 本書は株式会社北斗電子製 PUPPY 本体の使用方法及び付属ソフトについて説明するものであり、ユーザシステムは対象ではありません。
- PUPPY は乾電池で駆動させることを前提として設計されており、**外部からの電源供給は故障の原因となりますのでおやめ下さい。**
- PUPPY は弊社評価用マイコンボード (BaseBoard シリーズ、RY3048F-ONE) を本体基板に接続することで動作いたします。対応する評価用マイコンボードをお持ちでない場合は別途ご購入の上ご利用下さい。
- PUPPY は制御の学習を目的として設計・製作された物です。学習以外の用途、弊社 BaseBoard シリーズ(動作確認済)、RY3048F-ONE を除く製品との接続について動作保障致しません。特に産業用途において商品への組み込み等を行った場合、保証及びサポートは行いません。
- PUPPY のデザイン・機能・仕様は性能や安全性の向上を目的に予告なく変更することがあります。本書の図は実物と異なる場合もあります。
- 本書及び製品は著作権及び工業所有権によって保護されており、全ての権利は弊社に帰属します。本書の無断複写・複製・転載はできません。
- 弊社は安全にご利用頂く為に検討・対策を行っておりますが、潜在的な危険・誤使用については全てを予見できません。本書に記載されている警告が全てではありませんので、お客様の責任で理解・判断し正しく安全にご利用下さい。
- PUPPY を輸出する場合は日本国の外国為替及び外国貿易法(輸出貿易管理令などの関連法令を含む)および適用の外国法令に基づく輸出(再輸出)許可等、必要な手続きをとった上で行って下さい。

限定保証

弊社は PUPPY が頒布されているご利用条件に従って製造されたもので、材料・仕上げに欠陥がないことを保証致します。PUPPY の保証期間は購入頂いた日から 1 年間です。

免責事項

- 火災・地震・第三者による行為その他の事故により PUPPY に不具合が生じた場合
- お客様の故意・過失・誤用・異常な条件でのご利用によって PUPPY に不具合が生じた場合
- PUPPY 及び付属品へのご利用方法に起因した損害が発生した場合
- お客様によって PUPPY 及び付属品へ改造・修理がなされた場合

弊社は特定の目的・用途に関する保証や特許侵害に対する保証等、本保証条件以外のは明示・黙示に拘わらず一切保証致しません。また、直接的・間接的損害金もしくは欠陥製品や製品の使用方法に起因する損失金・費用には一切責任がありません。損害の発生についてあらかじめ知らされていた場合でも保証致しません。ただし、明示的に保証責任または担保責任を負う場合でも、その理由のいかなを問わず、累積的な損害賠償責任は、弊社が受領した対価を上限とします。

PUPPY は「現状」で販売されているものであり、使用に際してはお客様がその結果に一切の責任を負うものとします。弊社は使用または使用不能から生ずる損害に関して一切責任を負いません。保証は最初の購入者であるお客様ご本人にのみ適用され、お客様が転売された第三者には適用されません。よって転売による第三者またはその為になすお客様からのいかなる請求についても責任を負いません。

PUPPY 本体と本体に付属する製品全般の価格、又は仕様(本書含む)は予告無く変更される場合があります。

1-3. 製品の内容

本キットには以下の物が含まれます。

●本体基板



●ギヤボックス(6速ギヤボックスセット 田宮模型)



●タイヤ(ナロータイヤセット 田宮模型)



●エンコーダディスク ASSY



●付属 CD



●保護カバー※

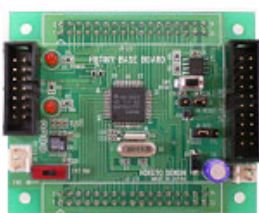


●PUPPY 回路図

※フィルムが貼ってあります。また、フィルムは写真と異なる場合もございます。

PUPPY セットご購入の方はさらに以下のものが付属します。

●マイコンボード



●書込み用 RS232C ケーブル



●マイコンボード回路図

●電源ケーブル

(PUPPY では使用しません)

- * 組み立て済みキットを御購入の方はギヤボックス・ロータリエンコーダ・タイヤが本体基板に組みつけられた状態になっています。
- * 単三乾電池 4 本は別途お求め下さい
- * 付属 CD に収録されているハードウェアマニュアルは開発時現在のもので、最新のものではありません。
最終的な設計に際しては、事前にルネサス エレクトロニクスもしくは特約店等へ最新の情報をご確認頂きますとともに、ホームページなどを通じて公開される情報に常にご注意下さい

2. 組み立て

※組み立て済みセットを御購入の方はこの章の工程は必要ありません。

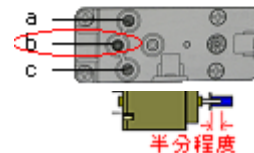
2-1. ギヤボックスの組み立て

ギヤボックスキット付属の説明書に従い組み立てます。このギヤボックスはギヤ比、出力軸の位置が選択できますが、PUPPY ではギヤ比を B タイプ(29.8:1)、出力軸の位置を b(中央)に選択して下さい。

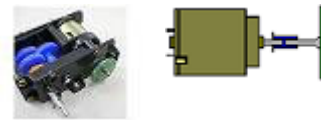
[注意事項]

- ギヤ、シャフトには十分にグリスを塗布して下さい
- ピニオンギヤをモータ軸に取り付ける工程では、モータ軸をピニオンギヤの穴半分程度まで挿入して下さい

タイプ	ギヤ比
A	11.6:1
B	29.8:1
C	76.5:1
D	196.7:1
E	505.9:1
F	1300.9:1

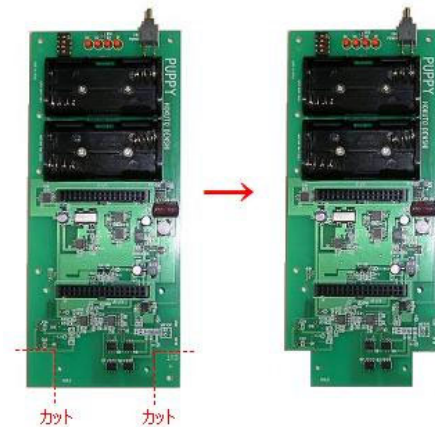


ギヤボックスが出来上がりましたらエンコーダディスク ASSY をギヤボックスの外側からピニオンギヤに差し込みます。

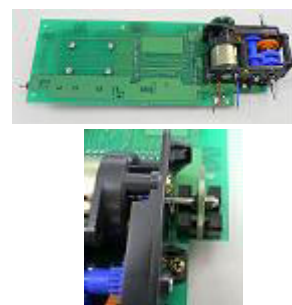


2-2. ギヤボックスの基板への取り付け

ギヤボックスを取り付ける前に本体基板の角をミシン目に沿ってニッパー等で切り落とします。バリが残った場合はヤスリ等できれいに取り除きます。



6速ギヤボックス HE に付属のφ3mm のタッピングビスとナットを使用してギヤボックスを基板に取り付けます。このときエンコーダディスクがフォトインタラプタ(コの字型の部品)のスリット中央に位置するように微調整し、モータを回したときエンコーダディスクとフォトインタラプタが接触しないようにします。



2-3. モータの配線

モータ配線を適当な長さにカットし、皮膜を剥いた後、モータの赤線を基板上側、青線を下側に差込み、基板裏面のパッドに半田付けします。

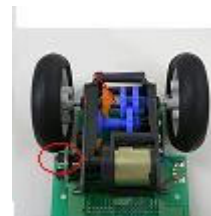


2-4. タイヤの組み立て・取り付け

付属のタイヤキットもギヤボックス同様に説明書のとおり組み立て、付属のナットでギヤボックスの出力シャフトに取り付けます。

〔注意事項〕

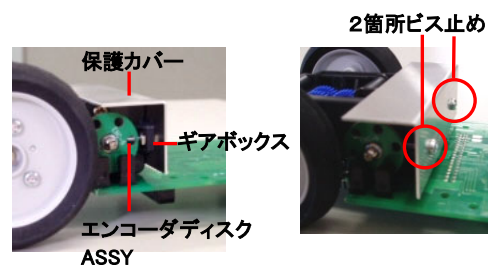
- ギヤボックスのシャフトに合ったジョイントをお選び下さい
- タイヤを取り付ける際、タイヤとフォトインタラプタが接触するような場合は出力シャフト固定位置を調整して下さい。



2-5. 保護カバーの取り付け

付属の保護カバーをギヤボックスを覆うように取り付けます。

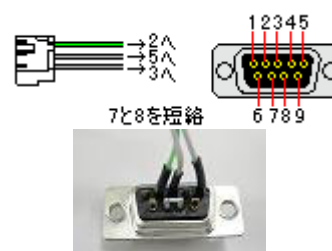
* ビスとナットは6速ギヤボックス HE に付属の物を使用します。



2-6. シリアルケーブルの作成 (PUPPY 単体購入の方)

マイコンボードに付属する RS232C ケーブルのコネクタの反対側に DSUB9 ピンコネクタを半田付けします。

- * すでにケーブルをお持ちの方はこの工程は必要ありません
- * DSUB9 ピンコネクタは付属しておりません。市販のものをお求め下さい。



2-7. 完成・各部の名称

マイコンボード、タイヤ、乾電池を図の様に取り付けて本体完成です。

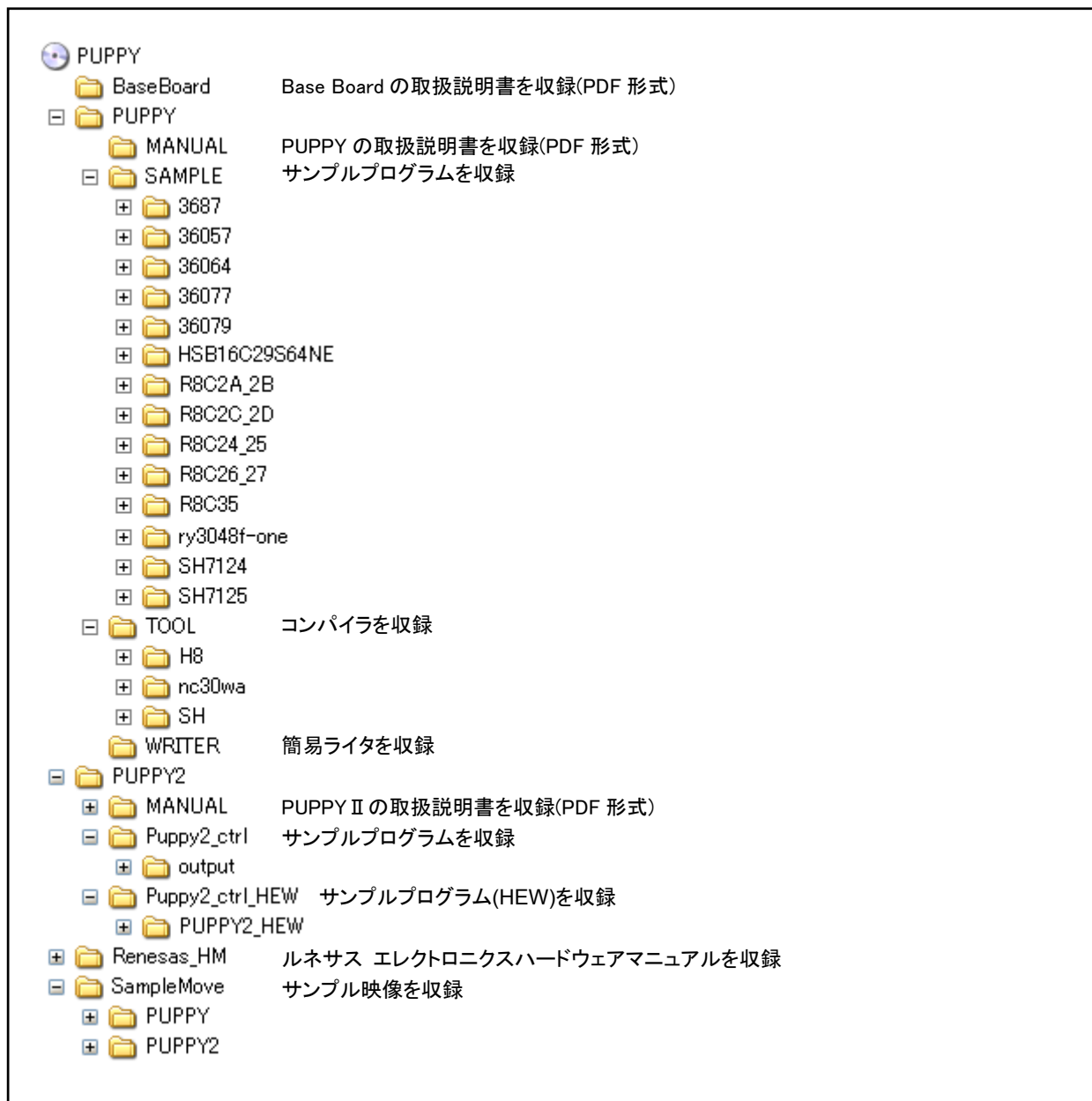
- * PUPPY 単体購入の方はマイコンボードは別途お求め下さい



3. 開発環境の構築

本キット付属CDの開発環境をご利用になる場合は、CD内の「PUPPY」フォルダを適宜にコピーしてお使い下さい。サンプルプログラムは `make.bat` を実行(ダブルクリック)してビルドします。本キット付属のCコンパイラ・アセンブラはオリジナルファイル形式HKTファイルを生成いたしますので作成されたユーザプログラム等の書き込みには `C:\¥PUPPY¥writer` の中にある書き込みソフトをご使用下さい。コンパイラ・アセンブラ・リンカの詳細はユーザズマニュアルをご覧ください。

収録ソフトウェアのフォルダ構成



4. 動作確認

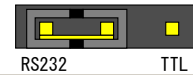
4-1. サンプルプログラムの書込み

倒立制御のサンプルプログラムをマイコンに書込みます。

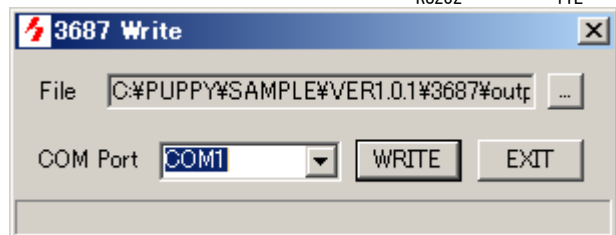
「PUPPY セット」ご購入の方はマイコンにあらかじめサンプルプログラムが書き込まれておりますのでこの手順は必要ありません。

a.BB64E3687F,36057F,36064F,36077F の場合

①BaseBoard 上の J6 ジャンパを[RS232]側にセットします。

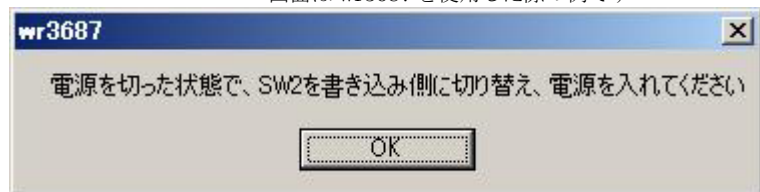


②wr3687 を起動すると右図のようなダイアログが現れますので、File に ¥PUPPY¥SAMPLE¥VER1.0.1¥3687¥output¥puppy.hkt を、COM Port に使用する COM ポートを選択します。

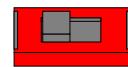


画面は wr3687 を使用した際の例です

③[WRITE]ボタンをクリックします。右図のようなメッセージが表示されます。

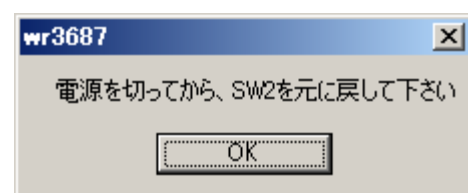


④シリアルケーブルで BaseBoard の J4(SHTinyBaseBoard では J6)とパソコンの COM ポートをつなぎ、ブート切り替えスイッチ(SW2)を書込み[FWE←]側に切り替え、PUPPY 本体の電源を入れます。BaseBoard 上の LED が 2 つ点灯します。



⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書込みを開始します。

⑥書込みが終了すると右図のようなメッセージが表示されます。

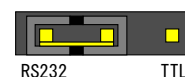


⑦PUPPY 本体の電源を切り、SW2 を元の位置に戻し、シリアルケーブルを取り外します。

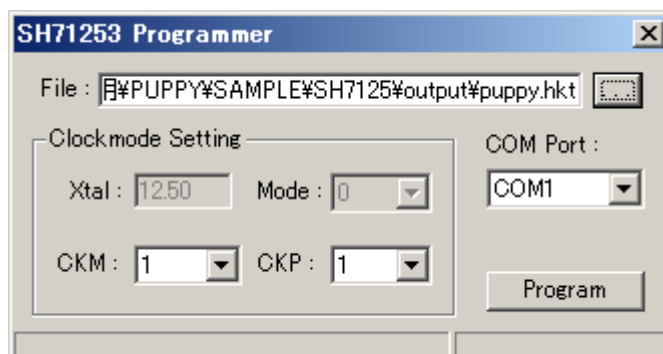
以上でプログラムの書込みは終了です。

b.BB64A7124F,7125F の場合

- ① BaseBoard 上の J4 ジャンパを[RS232]側にセットします。

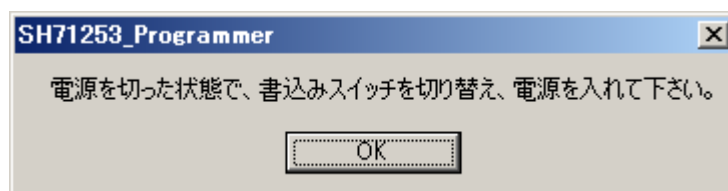


- ② SH712x3_Programmer を起動すると右図のようなダイアログが現れますので、File に ¥PUPPY¥SAMPLE¥SH7125¥output¥puppy.hkt を、COM Port に使用する COM ポートを選択します。

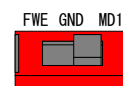


画面は SH71253_Programmer を使用した際の例です

- ③ [Program]ボタンをクリックします。右図のようなメッセージが表示されます。



- ④ シリアルケーブルで BaseBoard の J6 とパソコンの COM ポートをつなぎ、ブート切り替えスイッチ(SW2)を基板外側に切り替え、PUPPY 本体の電源を入れます。BaseBoard 上の POWER LED(D2)と BOOT LED(D7)が点灯します。



- ⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書き込みを開始します。

- ⑥ 書き込みが終了すると右図のようなメッセージが表示されます。



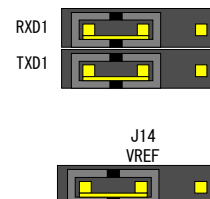
- ⑦ PUPPY 本体の電源を切り、SW2 を元の位置に戻し、シリアルケーブルを取り外します。

以上でプログラムの書き込みは終了です。

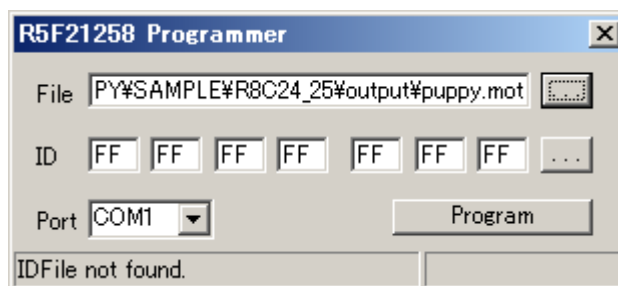
c.BB52A24xSNF,25xSNF の場合

x には内蔵 ROM のサイズの違いにより 4,6,8 のいずれかが入ります。

- ① BaseBoard 上の J6 ジャンパを RXD1,TXD1 側にセットします。
J14 ジャンパを基板左側にセットします。



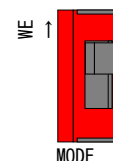
- ② R5F21258_Programmer を起動すると右図のようなダイアログが現れますので、File に¥PUPPY¥SAMPLE¥R8C24_25¥output¥puppy.mot を、COM Port に使用する COM ポートを選択します。



- ③ [Program]ボタンをクリックします。右図のようなメッセージが表示されます。



- ④ シリアルケーブルで BaseBoard の J6 とパソコンの COM ポートをつなぎ、ブート切り替えスイッチ(SW1)を WE→側に切り替え、PUPPY 本体の電源を入れます。



- ⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書き込みを開始します。

- ⑥ 書き込みが終了すると右図のようなメッセージが表示されます。

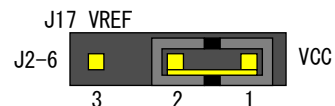
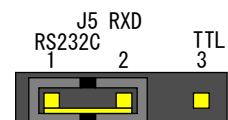


- ⑦ PUPPY 本体の電源を切り、SW2 を元の位置に戻し、シリアルケーブルを取り外します。

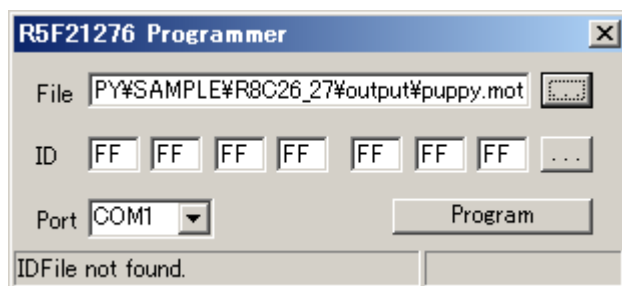
以上でプログラムの書き込みは終了です。

d.BB32A26xSNF,27xSNF の場合

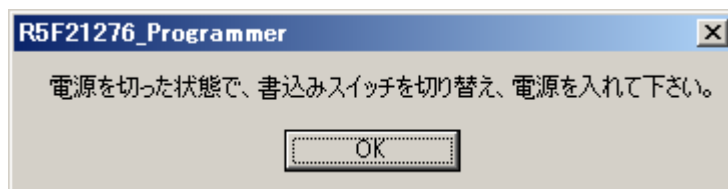
- ① BaseBoard 上の J5 ジャンパを RS232C 側にセットします。
J17 ジャンパを VCC 側にセットします。



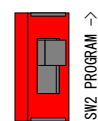
- ② R5F21276_Programmer を起動すると右図のようなダイアログが現れますので、File に¥PUPPY¥SAMPLE¥R8C26_27 ¥output¥puppy.mot を、COM Port に使用する COM ポートを選択します。



- ③ [Program]ボタンをクリックします。右図のようなメッセージが表示されます。



- ⑤ シリアルケーブルで BaseBoard の J8 とパソコンの COM ポートをつなぎ、ブート切り替えスイッチ(SW2)を PROGRAM→側に切り替え、PUPPY 本体の電源を入れます。



- ⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書き込みを開始します。

- ⑥ 書き込みが終了すると右図のようなメッセージが表示されます。



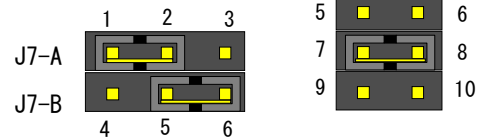
- ⑦ PUPPY 本体の電源を切り、SW2 を元の位置に戻し、シリアルケーブルを取り外します。

以上でプログラムの書き込みは終了です。

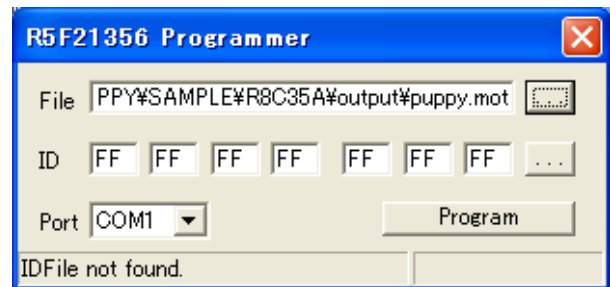
e.BB52A35xANF,35xCNF,35xDNF の場合

① BaseBoard 上の J7-A ジャンパを 1-2 ピン側、J7-B ジャンパを 5-6 ピン側にセットします。

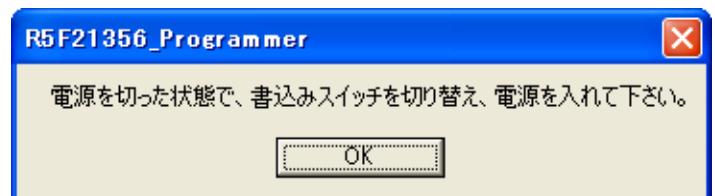
J13 および J14 ジャンパを 3-4 ピン、7-6 ピンにセットします。



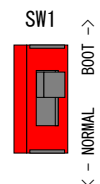
② R5F21356_Programmer を起動すると右図のようなダイアログが現れますので、File に ¥PUPPY¥SAMPLE¥R8C35 ¥output¥puppy.mot を、COM Port に使用する COM ポートを選択します。



③ [Program]ボタンをクリックします。右図のようなメッセージが表示されます。

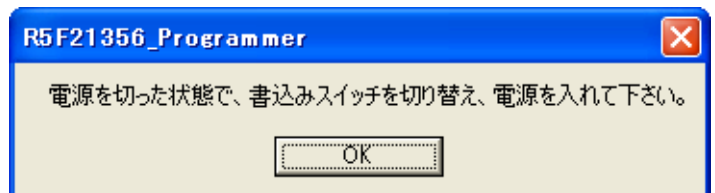


④ シリアルケーブルで BaseBoard の J6 とパソコンの COM ポートをつなぎ、MODE 切り替えスイッチ(SW1)を BOOT→側に切り替え、PUPPY 本体の電源を入れます。



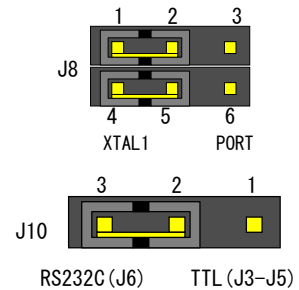
⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書込みを開始します。

⑥ 書込みが終了すると右図のようなメッセージが表示されます。

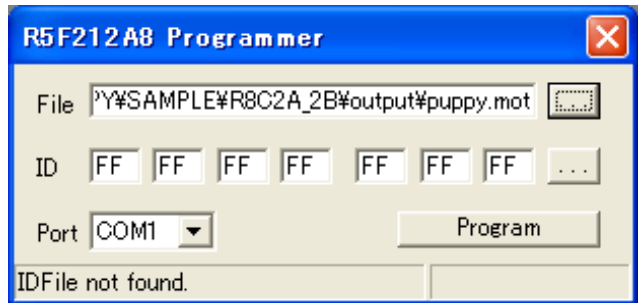


f.BB64A2AxSNF,2BxSNF の場合

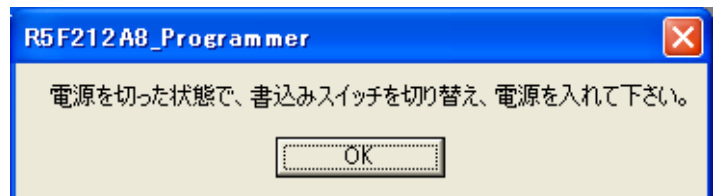
- ① BaseBoard 上の J8 ジャンパを XTAL1 側にセットします。
J10 ジャンパを RS232C(J6)側にセットします。



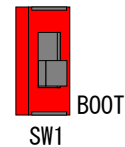
- ② R5F212A8_Programmer を起動すると右図のようなダイアログが現れますので、File に¥PUPPY¥SAMPLE¥R8C2A_2B¥output¥puppy.mot を、COM Port に使用する COM ポートを選択します。



- ③ [Program]ボタンをクリックします。右図のようなメッセージが表示されます。

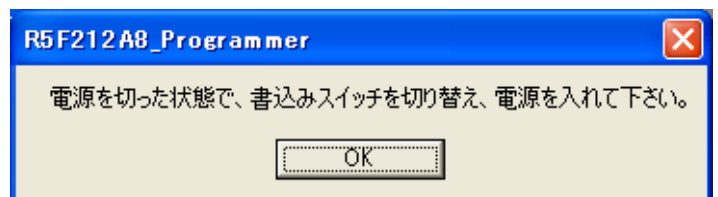


- ⑦ シリアルケーブルで BaseBoard の J6 とパソコンの COM ポートをつなぎ、ブート切り替えスイッチ(SW1)を BOOT 側に切り替え、PUPPY 本体の電源を入れます。



- ⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書込みを開始します。

- ⑥ 書込みが終了すると右図のようなメッセージが表示されます。

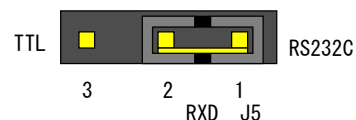


- ⑦ PUPPY 本体の電源を切り、SW1 を元の位置に戻し、シリアルケーブルを取り外します。

以上でプログラムの書込みは終了です。

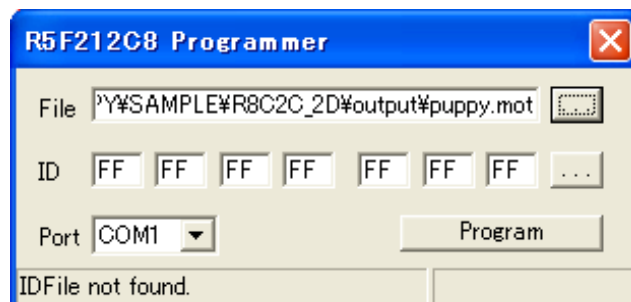
g.BB80A2CxSNF,2DxSNF の場合

- ① BaseBoard 上の J5 ジャンパを RS232C 側にセットします。

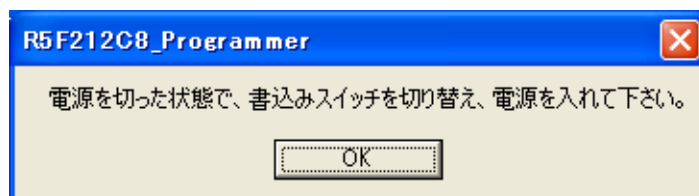


※基板シルク上は RXD J4 となっておりますが、正しくは J5 となります

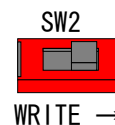
- ② R5F212C8_Programmer を起動すると右図のようなダイアログが現れますので、File に ¥PUPPY¥SAMPLE¥R8C2C_2D ¥output¥puppy.mot を、COM Port に使用する COM ポートを選択します。



- ③ [Program] ボタンをクリックします。右図のようなメッセージが表示されます。

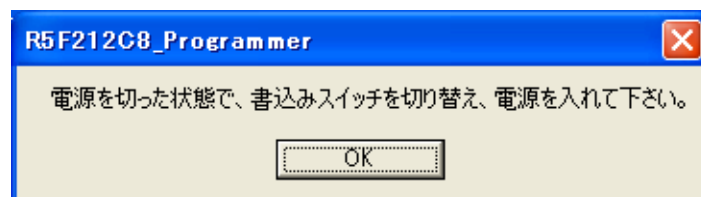


- ④ シリアルケーブルで BaseBoard の J6 とパソコンの COM ポートをつなぎ、ブート切り替えスイッチ(SW2)を WRITE → の方向に切り替え、PUPPY 本体の電源を入れます。



- ⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書き込みを開始します。

- ⑥ 書き込みが終了すると右図のようなメッセージが表示されます。



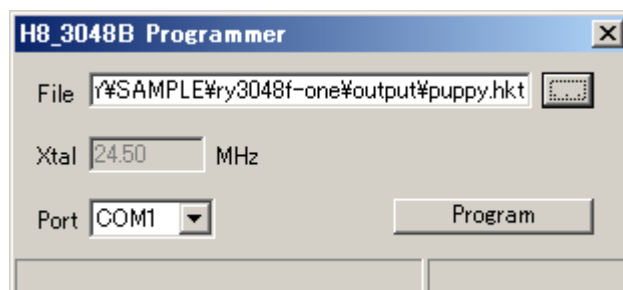
- ⑦ PUPPY 本体の電源を切り、SW2 を元の位置に戻し、シリアルケーブルを取り外します。

以上でプログラムの書き込みは終了です。

h.RY3048F-ONE の場合

- ① RY3048F-ONE アダプタをお持ちの方はアダプタ取扱説明書に従って RY3048F-ONE とアダプタ、PUPPY を接続します。RY3048F-ONE をお持ちでない方は後示の接続表のとおり PUPPY と RY3048F-ONE を接続してください。

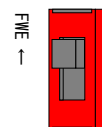
- ② H8_3048B_Programmer を起動すると右図のようなダイアログが現れますので、File に¥PUPPY¥SAMPLE¥ry3048f-one¥output¥puppy.hkt を、COM Port に使用する COM ポートを選択します。



- ③ [Program] ボタンをクリックします。右図のようなメッセージが表示されます。



- ④ シリアルケーブルでマイコンボードの J4 とパソコンの COM ポートをつなぎ、ブート切り替えスイッチを FWE→側に切り替え、PUPPY 本体の電源を入れます。



- ⑤ ②で表示されたメッセージの[OK]ボタンをクリックし、書き込みを開始します。

- ⑥ 書き込みが終了すると右図のようなメッセージが表示されます。



- ⑦ PUPPY 本体の電源を切り、SW2 を元の位置に戻し、シリアルケーブルを取り外します。

以上でプログラムの書き込みは終了です。

i.HSB16C29S64NE の場合

HSB16C29S64NE のサンプルプログラムの書き込みについては、複数の書き込み方法がございますので、詳しくは HSB16C29S64NE 付属の取扱説明書をご参照下さい。

4-2. サンプルプログラムの実行

サンプルプログラムは、PUPPY の倒立制御を実行するプログラムです。以下に動作させる手順を示します。

① 単三乾電池 4 本を電池ボックスにセットする

【 注意事項 】 単三型アルカリ電池 (LR6) をお使い下さい

充電可能なニッケル水素やニッカド電池を使う事ができますが、電池によっては不安定になったり、誤作動を起こす場合があります。この場合、アルカリ電池でご使用下さい。



② BaseBoard のスイッチ、ジャンパの確認 (SW2→基板内側)



③ PUPPY 上部のディップスイッチの設定

(1,2,→ON 側 3,4→数字側)



④ PUPPY 上部を持ち、車輪を接地させ、PUPPY が前後に倒れないような位置で支持し、電源を入れる。(しばらくその状態を保持して下さい)



* LED が 4 つ点灯し、0.5 秒おきに左から順番に消灯する



* すべての LED が消灯した後、一瞬すべての LED が点灯する



⑤ ゆっくりと手を離す

PUPPY はバランスをとりながら倒立制御を開始します。

操作手順を示した動画ファイル(倒立操作.wmv)が付属 CD の「MANUAL」フォルダの中にありますのでそちらもご参照下さい。

前ページ記載方法でなかなか倒立しない場合は・・・
壁などに立てかけて倒立制御を開始させることもできます。

上記①,②の操作後

③PUPPY 上部ディップスイッチの設定(1,3,4→数字側 2→ON 側)



④電源を切った状態で PUPPY を垂直な壁に立てかける
(手を離しても倒れないようにして下さい)



⑤電源を入れる



5. 動作原理

構造的に不安定な PUPPY を倒立させる制御は、ほうきの柄を手のひらの上に乗せてバランスをとる動作に似ています。概念的には PUPPY 本体の傾斜角、角速度、車輪の角速度等から車輪に適切なトルクを求め、それをモータに発生させることを繰り返せば倒立状態を継続することができます。ここではサンプルプログラムでの PUPPY のモデリング、制御系設計の方法について簡単に説明します。

5-1. PUPPY の運動方程式

モデリングにはラグランジュの運動方程式を使用します。この方法は系全体の運動エネルギー、ポテンシャルエネルギーを一般化座標と対応する一般化速度を使って表すと、機械的な計算を行うだけで系の運動方程式が導かれる利点があります。

以下のように変数とパラメータの記号を定義します。

θ : 本体の進行方向の傾斜角(鉛直軸から進行方向に傾斜したときが正)

ϕ : 本体から測った車輪の回転角(前進回転方向が正)

I : 本体の傾斜方向の慣性モーメント

L : 車軸から測った本体の重心の距離

M : 本体の質量

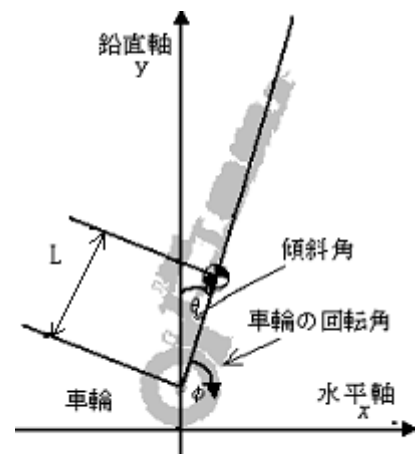
m : 車輪 2 つの質量

J : 車輪の慣性モーメント

r : 車輪の半径

D_ϕ : 車輪の回転に伴う速度摩擦係数

D_θ : 車軸回りに本体が回転するときの速度摩擦係数



PUPPY の側面模式図

5-1-1. 運動エネルギー

剛体の運動エネルギーは、重心回りの回転運動エネルギーと重心の並進運動エネルギーの和で与えられます。

車輪の回転運動エネルギーは車輪の軸回りの慣性モーメントが J なので

$$T_{wr} = \frac{1}{2} J (\dot{\phi} + \dot{\theta})^2 \quad \dots\dots(1-1)$$

ここで、回転角速度が $\dot{\phi}$ だけではなく $\dot{\theta}$ が加算されているのは、傾斜することによっても車輪が回転するためです。

傾斜方向の回転運動エネルギーは、車軸回りに θ だけ回転するときには本体の重心回りに同じだけ回転していることに注目すると

$$T_{Br} = \frac{1}{2} I \dot{\theta}^2 \quad \dots\dots(1-2)$$

となります。

並進運動エネルギーに関しては、車輪の並進運動エネルギーと本体の並進運動エネルギーをそれぞれ求めなければなりません。

車輪の並進運動エネルギーは、車輪 2 つの質量が m なので重心が車軸の中心にあると考えて求めます。車輪の動く速さを v とすると、車輪の並進運動エネルギーは

$$v = r(\dot{\theta} + \dot{\phi})$$

$$T_{wt} = \frac{1}{2} m v^2 = \frac{1}{2} m r^2 (\dot{\theta}^2 + 2\dot{\theta}\dot{\phi} + \dot{\phi}^2) \quad \dots\dots(1-3)$$

本体の並進運動エネルギーは

$$T_{Bt} = \frac{1}{2} M V^2$$

$$= \frac{1}{2} M (V_x^2 + V_y^2)$$

$$= \frac{1}{2} M \{ \dot{\phi}^2 r^2 + \dot{\theta}^2 (r^2 + 2rL \cos \theta + L^2) + 2\dot{\phi}\dot{\theta}(r^2 + rL \cos \theta) \}$$

$$\dots\dots(1-4)$$

ただし

$$\begin{cases} V_x = \frac{d}{dt} \{ r(\theta + \phi) + L \sin \theta \} \\ V_y = \frac{d}{dt} (L \cos \theta) \end{cases}$$

となります。

運動エネルギーの総和は式(1-1),式(1-2),式(1-3),式(1-4)を足しあわせたものとなります。

$$T = T_{wr} + T_{Br} + T_{wt} + T_{Bt} \quad \dots\dots(1-5)$$

5-1-2. ポテンシャルエネルギー

ポテンシャルエネルギー U は重心の位置エネルギーなので

$$U = MgL \cos \theta \quad \dots\dots(1-6)$$

となります。

5-1-3. ラグランジュ関数とラグランジュの運動方程式

運動方程式を書き下すために必要なラグランジュ関数 $L(\varphi, \dot{\varphi}, \theta, \dot{\theta})$ は、運動エネルギーとポテンシャルエネルギーから機械的に以下のように求められます。

$$L = T - U \quad \dots\dots(1-7)$$

ラグランジュの運動方程式は、車輪を駆動する軸トルクを τ とすると以下のように与えられます。

$$\begin{cases} \frac{d}{dt} \left(\frac{\partial}{\partial \dot{\varphi}} L \right) - \frac{\partial}{\partial \varphi} L + D_{\varphi} \dot{\varphi} = \tau \\ \frac{d}{dt} \left(\frac{\partial}{\partial \dot{\theta}} L \right) - \frac{\partial}{\partial \theta} L + D_{\theta} \dot{\theta} = 0 \end{cases} \quad \dots\dots(1-8)$$

式(1-8)に式(1-7)を代入して計算を進めていくと以下の結果が導き出されます。

$$\begin{cases} \left\{ (m+M)r^2 + J \right\} \ddot{\varphi} + \left\{ (m+M)r^2 + MrL \cos \theta + J \right\} \ddot{\theta} - \dot{\theta}^2 MrL \sin \theta + D_{\varphi} \dot{\varphi} = \tau \\ \left\{ (m+M)r^2 + MrL \cos \theta + J \right\} \ddot{\varphi} + \left\{ (m+M)r^2 + 2MrL \cos \theta + ML^2 + J + I \right\} \ddot{\theta} \\ - \dot{\theta}^2 MrL \sin \theta - MgL \sin \theta + D_{\theta} \dot{\theta} = 0 \end{cases} \quad \dots\dots(1-9)$$

式(1-9)の定数部分を変数変換して見通しを良くします。

$$\begin{cases} a = (m+M)r^2 + J \\ b = MrL \\ c = ML^2 + I \\ u = MgL \end{cases}$$

とおくと

$$\begin{cases} a \ddot{\varphi} + (a + b \cos \theta) \ddot{\theta} - \dot{\theta}^2 b \sin \theta + D_{\varphi} \dot{\varphi} = \tau \\ (a + b \cos \theta) \ddot{\varphi} + (a + 2b \cos \theta + c) \ddot{\theta} - \dot{\theta}^2 b \sin \theta - u \sin \theta + D_{\theta} \dot{\theta} = 0 \end{cases} \quad \dots\dots(1-10)$$

と表されます。

このラグランジュの運動方程式(1-10)を用いて制御の設計を行っています。

5-2. PUPPY の制御モデル

PUPPY が倒立状態にあるときには θ と $\dot{\theta}$ は小さな値をとり続けます。前章で導いた運動方程式

は、これらの変数に関して非線形な関数を含んでいましたが、倒立状態を考えますと $\theta = 0, \dot{\theta} = 0$

のまわりで線形近似した式に置き換えても問題が生じないと考えます。

このことから $\cos \theta \approx 1, \sin \theta \approx \theta$ と近似し、 $\dot{\theta}$ 等の非線形項をゼロとみなすと式 (1-10) は

$$\begin{cases} a\ddot{\phi} + (a+b)\ddot{\theta} + D_{\phi}\dot{\phi} = \tau \\ (a+b)\ddot{\phi} + (a+2b+c)\ddot{\theta} - u\theta + D_{\theta}\dot{\theta} = 0 \end{cases} \dots\dots(2-1)$$

と表されます。この式を変形すると、

$$\begin{aligned} \ddot{\theta} &= a_{21}\theta + a_{22}\dot{\theta} + a_{23}\dot{\phi} + b_2\tau \\ \ddot{\phi} &= a_{31}\theta + a_{32}\dot{\theta} + a_{33}\dot{\phi} + b_3\tau \end{aligned} \dots\dots(2-2)$$

ただし、

$$\begin{aligned} a_{21} &= \frac{au}{ac-b^2} & a_{22} &= \frac{-aD_{\theta}}{ac-b^2} & a_{23} &= \frac{(a+b)D_{\phi}}{ac-b^2} \\ a_{31} &= \frac{-(a+b)u}{ac-b^2} & a_{32} &= \frac{(a+b)D_{\theta}}{ac-b^2} & a_{33} &= \frac{-(a+2b+c)D_{\phi}}{ac-b^2} \\ b_2 &= \frac{-(a+b)}{ac-b^2} & b_3 &= \frac{(a+2b+c)}{ac-b^2} \end{aligned}$$

となりこの式を状態方程式の形にすると

$$\frac{d}{dt} \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \dot{\phi}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \dot{\phi}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ b_2 \\ b_3 \end{bmatrix} \tau(t) \dots\dots(2-3)$$

となります。この式 (2-3) が以降の制御設計の基礎となる状態方程式となります。

5-3. 倒立走行制御系の設計

PUPPY を倒立させる場合には、倒立する位置を制御するのか、あるいは倒立させて走行させるか(静止状態も含みます)を最初に決める必要があります。サンプルプログラムでは制御器の次数が低くて済む後者の制御を採用します。

5-3-1. 離散時間制御モデル

PUPPY はマイコンにより制御しますので連続時間形式で表される式(2-3)を離散時間形式の動特性モデルに変換しなければなりません。この変換には連続時間系の微分方程式(2-3)の解が利用できます。式(2-3)を簡単に

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad \dots\dots(3-1)$$

ただし

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 \\ b_2 \\ b_3 \end{bmatrix} \quad \mathbf{x}(t) = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \dot{\phi}(t) \end{bmatrix} \quad \mathbf{u}(t) = [\tau(t)]$$

のように表して $t = t_0$ における状態 $\mathbf{x}(t_0)$ を初期値とする解を求めると

$$\mathbf{x}(t) = e^{\mathbf{A}(t-t_0)}\mathbf{x}(t_0) + \int_{t_0}^t e^{\mathbf{A}(t-s)}\mathbf{B}\mathbf{u}(s)ds \quad \dots\dots(3-2)$$

となります。つぎに、

$$\begin{aligned} t_0 = kT, t = (k+1)T & : T \text{ は制御周期,} \\ \mathbf{u}(t) = \text{const.} & \quad \text{for} \quad t \in [kT, (k+1)T] \end{aligned}$$

とにおいて、式(3-2)を書き下すと

$$\begin{aligned} \mathbf{x}(k+1) &= e^{\mathbf{A}T}\mathbf{x}(k) + \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-s)}\mathbf{B}\mathbf{u}(k)ds \\ &= e^{\mathbf{A}T}\mathbf{x}(k) + \int_0^T e^{\mathbf{A}s}\mathbf{B}\mathbf{u}(k)ds \end{aligned} \quad \dots\dots(3-3)$$

となります。これが、周期 T で制御を加えた場合の PUPPY の離散時間制御モデルです。これを

$$\begin{aligned} \mathbf{A}_D &= e^{\mathbf{A}T} \\ \mathbf{B}_D &= \int_0^T e^{\mathbf{A}s}\mathbf{B}ds \end{aligned}$$

とおくと

$$\mathbf{x}(k+1) = \mathbf{A}_D\mathbf{x}(k) + \mathbf{B}_D\mathbf{u}(k) \quad \dots\dots(3-4)$$

あるいは

$$\begin{bmatrix} \theta(k+1) \\ \dot{\theta}(k+1) \\ \dot{\phi}(k+1) \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} \theta(k) \\ \dot{\theta}(k) \\ \dot{\phi}(k) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix} \tau(k) \quad \dots\dots(3-5)$$

となります。

5-3-2. 倒立走行制御系の構成

倒立走行制御は本体の重心回りに回転する角速度 $\dot{\theta}$ をゼロ、車輪の回転角速度 $\dot{\phi}$ を目標値に制御することを目的とし、以下のように表されます。

$$\begin{aligned}\dot{\theta}(k) &\rightarrow 0(k \rightarrow \infty) \\ \dot{\phi}(k) &\rightarrow \dot{\phi}_d(k \rightarrow \infty)\end{aligned}\quad \text{ただし } \dot{\phi}_d \text{ は制御指令値}$$

これらを実現する手段として誤差ベクトル $\mathbf{e}(k)$ を導入します。

$$\begin{aligned}\mathbf{e}(k) &= \dot{\phi}_d - \dot{\phi}(k) \\ &= \dot{\phi}_d - \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta(k) \\ \dot{\theta}(k) \\ \dot{\phi}(k) \end{bmatrix}\end{aligned}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \quad \text{とすると}$$

$$\mathbf{e}(k) = \dot{\phi}_d - \mathbf{C}\mathbf{x}(k) \quad \dots\dots\dots(3-6)$$

つぎに、式(3-4)、式(3-6)の両辺に前進差分演算子 Δ ：

$$\Delta\mathbf{h}(k) \equiv \mathbf{h}(k+1) - \mathbf{h}(k) \quad \dots\dots\dots(3-7)$$

を作用させると差分演算子 Δ の線形性により以下の式が導かれます。

$$\Delta\mathbf{x}(k+1) = \mathbf{A}_D\Delta\mathbf{x}(k) + \mathbf{B}_D\Delta\mathbf{u}(k) \quad \dots\dots\dots(3-8)$$

$$\Delta\mathbf{e}(k) = -\mathbf{C}\Delta\mathbf{x}(k) \quad \dots\dots\dots(3-9)$$

ここで、式(3-9)の左辺を定義式に則って元に戻した後、式(3-8)と一緒にして次のような拡大系を構成します。

$$\begin{bmatrix} \Delta\mathbf{x}(k+1) \\ \mathbf{e}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_D & \mathbf{0} \\ -\mathbf{C} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}(k) \\ \mathbf{e}(k) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_D \\ \mathbf{0} \end{bmatrix} \Delta\mathbf{u}(k) \quad \dots\dots\dots(3-10)$$

そして、最後にこの拡大形式(3-10)に対して漸近安定な状態フィードバック則を求めていきます。いま、漸近安定化する状態フィードバック則がリカッチの式を解くなどして式(3-11)のように求めることができたとします。

$$\begin{aligned}\Delta\mathbf{u}(k) &= \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_e \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}(k) \\ \mathbf{e}(k) \end{bmatrix} \\ &= \mathbf{F}_x\Delta\mathbf{x}(k) + \mathbf{F}_e\mathbf{e}(k)\end{aligned} \quad \dots\dots\dots(3-11)$$

実際に PUPPY に加える操作量 $\mathbf{u}(k)$ を求めるためには、式(3-11)の k を過去に向けて変えた関係式を作り、辺々総和をとります。

$$\begin{aligned}
& \Delta \mathbf{u}(k) + \Delta \mathbf{u}(k-1) + \cdots + \Delta \mathbf{u}(0) \\
&= \mathbf{F}_x \{ \Delta \mathbf{x}(k) + \Delta \mathbf{x}(k-1) + \cdots + \Delta \mathbf{x}(0) \} + \mathbf{F}_e \{ \mathbf{e}(k) + \mathbf{e}(k-1) + \cdots + \mathbf{e}(0) \} \\
&\quad \cdots \cdots \cdots (3-12)
\end{aligned}$$

式(3-12)の差分演算により目的とする操作量が得られます。

$$\mathbf{u}(k+1) = \mathbf{F}_x \mathbf{x}(k+1) + \mathbf{F}_e \sum_{i=0}^k \mathbf{e}(i) - \mathbf{F}_x \mathbf{x}(0) + \mathbf{u}(0) \quad \cdots \cdots \cdots (3-13)$$

以上が状態フィードバックによる PUPPY の倒立走行制御の基本設計プロセスです。しかしながら現実には、式(3-13)をそのまま用いるのは制御周期が状態観測と式(3-13)の計算に必要とされる時間に比較して十分長い場合しか理論どおりの性能を示しません。デジタル制御では制御と制御の間の時間は操作量を変化させることのできないオープンループ構造になってしまいますので、できるだけ制御周期を短くにとって絶えずフィードバックが働くようにしたいと望むのが普通でしょう。このような場合には、1 制御遅れの制御系を構成して対処することとします。1 制御遅れの制御とは操作量を 1 ステージ前の観測値のみから構成する制御で式(3-11)を動特性式(3-12)を使って、以下のように変形して導きます。

$$\begin{aligned}
\Delta \mathbf{u}(k) &= \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_e \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}(k) \\ \mathbf{e}(k) \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_e \end{bmatrix} \begin{bmatrix} \mathbf{A}_D & \mathbf{0} \\ -\mathbf{C} & \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}(k-1) \\ \mathbf{e}(k-1) \end{bmatrix} + \begin{bmatrix} \mathbf{F}_x & \mathbf{F}_e \end{bmatrix} \begin{bmatrix} \mathbf{B}_D \\ \mathbf{0} \end{bmatrix} \Delta \mathbf{u}(k-1) \\
&= \begin{bmatrix} \mathbf{F}_x \mathbf{A}_D & -\mathbf{F}_e \mathbf{C} & \mathbf{F}_e \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}(k-1) \\ \mathbf{e}(k-1) \end{bmatrix} + \mathbf{F}_x \mathbf{B}_D \Delta \mathbf{u}(k-1) \\
&= (\mathbf{F}_x \mathbf{A}_D - \mathbf{F}_e \mathbf{C}) \Delta \mathbf{x}(k-1) + \mathbf{F}_x \mathbf{e}(k-1) + \mathbf{F}_x \mathbf{B}_D \Delta \mathbf{u}(k-1) \\
&\quad \cdots \cdots \cdots (3-14)
\end{aligned}$$

後は、先ほどと同様な方法で実際に入力できる形の操作量が求められます。

$$\begin{aligned}
& \Delta \mathbf{u}(k) + \Delta \mathbf{u}(k-1) + \cdots + \Delta \mathbf{u}(0) \\
&= (\mathbf{F}_x \mathbf{A}_D - \mathbf{F}_e \mathbf{C}) \{ \Delta \mathbf{x}(k-1) + \Delta \mathbf{x}(k-2) + \cdots + \Delta \mathbf{x}(0) \} \\
&\quad + \mathbf{F}_e \{ \mathbf{e}(k-1) + \mathbf{e}(k-2) + \cdots + \mathbf{e}(0) \} \\
&\quad + \mathbf{F}_x \mathbf{B}_D \{ \Delta \mathbf{u}(k-1) + \Delta \mathbf{u}(k-2) + \cdots + \Delta \mathbf{u}(0) \} \\
&\quad \cdots \cdots \cdots (3-15)
\end{aligned}$$

操作量はつぎのようになります。

$$\begin{aligned}
\mathbf{u}(k+1) &= (\mathbf{F}_x \mathbf{A}_D - \mathbf{F}_e \mathbf{C}) \mathbf{x}(k) + \mathbf{F}_e \sum_{i=0}^{k-1} \mathbf{e}(i) + \mathbf{F}_x \mathbf{B}_D \mathbf{u}(k) \\
&\quad - (\mathbf{F}_x \mathbf{A}_D - \mathbf{F}_e \mathbf{C}) \mathbf{x}(0) - \mathbf{F}_x \mathbf{B}_D \mathbf{u}(0) + \mathbf{u}(1) \\
&\quad \cdots \cdots \cdots (3-16)
\end{aligned}$$

さらに式(3-16)の右辺の最後 3 項をゼロとみなしステージを 1 つ遅らせると

$$\mathbf{u}(k) = (\mathbf{F}_x \mathbf{A}_D - \mathbf{F}_e \mathbf{C}) \mathbf{x}(k-1) + \mathbf{F}_e \sum_{i=0}^{k-2} \mathbf{e}(i) + \mathbf{F}_x \mathbf{B}_D \mathbf{u}(k-1) \quad \cdots \cdots \cdots (3-17)$$

この式(3-17)が実際に PUPPY に加えられる制御入力 $\mathbf{u}(k)$ になります。書き下すと、

$$u(k) = F_1\theta(k-1) + F_2\dot{\theta}(k-1) + F_3\dot{\phi}(k-1) + F_4\sum_{i=0}^{k-2}e(i) + F_5u(k-1) \quad \dots\dots\dots(3-18)$$

となります。この式は、本体姿勢角、本体姿勢角速度、車輪角速度の一制御周期前の観測値それぞれ $\theta(k-1)$ 、 $\dot{\theta}(k-1)$ 、 $\dot{\phi}(k-1)$ 、一制御周期前までの制御偏差の積算値 $\sum_{i=0}^{k-2}e(i)$ 、並びに一制御周期前の制御入力 $u(k-1)$ にそれぞれ漸近安定なフィードバックゲイン $F_1 \sim F_5$ を掛けて足し合わせたものが現制御周期 k の制御入力 $u(k)$ となるということを表します。

5－4．倒立走行制御の実現

前項までで理論的な倒立走行の制御モデルを説明しました。（3-18）式で倒立走行制御が実現できますが、実際に PUPPY のサンプルプログラムではどのような構成でこの制御を実現しているのかを説明します。

5－4－1．状態量の取得

式(3-18)において制御入力を算出する際、本体姿勢角、本体姿勢角速度、車輪回転角速度がそれぞれ必要になります。

本体姿勢角速度は、圧電振動ジャイロセンサで計測し、時間積分して本体姿勢角を算出します。車輪角速度はロータリエンコーダを用いて車輪回転角を計測し、時間微分して算出します。それぞれの回路構成等は「PUPPY の回路」で説明します。

5－4－2．制御周期の確立

離散時間形式の制御を正しく行うには制御周期を確立し、制御周期ごとに状態量を計測し、それらの値から得られる制御入力を次の制御周期までの間、系に与え続けなければなりません。

PUPPY ではマイコンに内蔵のタイマを用いて制御周期を確立しています。詳しくは、「サンプルプログラムについて」で説明します。

5－4－3．PUPPY のパラメータ

倒立走行制御の設計を行う際に車輪の半径や質量等さまざまなパラメータを設定しなければなりません。ほとんどのものは簡単に計測可能です。

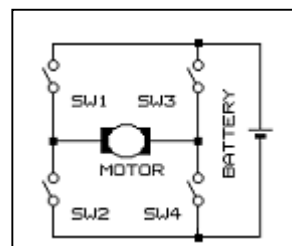
車輪半径	r	0.029 [m]
車輪(2 つ分)と車軸の質量	m	0.06 [kg]
本体の質量(乾電池含む)	M	0.27 [kg]
車輪と車軸の慣性モーメント	J	0.00115 [$kg \cdot m^2$]
本体の慣性モーメント	I	0.0036973 [$kg \cdot m^2$]
車軸から本体重心までの距離	L	0.105 [m]
車体の粘性摩擦抵抗	D_θ	2.00e-7 [Nms/rad]
車軸の粘性摩擦抵抗	D_ϕ	1.00e-4 [Nms/rad]

PUPPY のパラメータ例

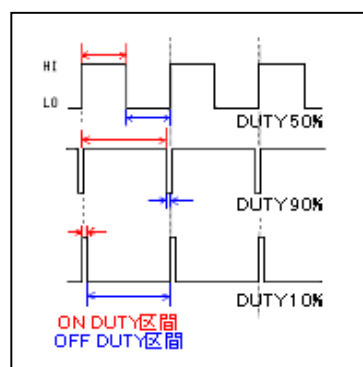
6. PUPPYの回路

6-1. モータドライブ回路

PUPPY のモータドライブ回路は H ブリッジ方式を採用しています。右にその模式図を示します。図の SW1 と SW4 を ON にすれば、モータを正転させることができ、SW3 と SW2 を ON にすればモータを逆転させることができます。また、この回路で SW1 を ON にしたまま SW4 を素早く ON と OFF とを繰り返したとき SW4 の ON 時間が OFF 時間に

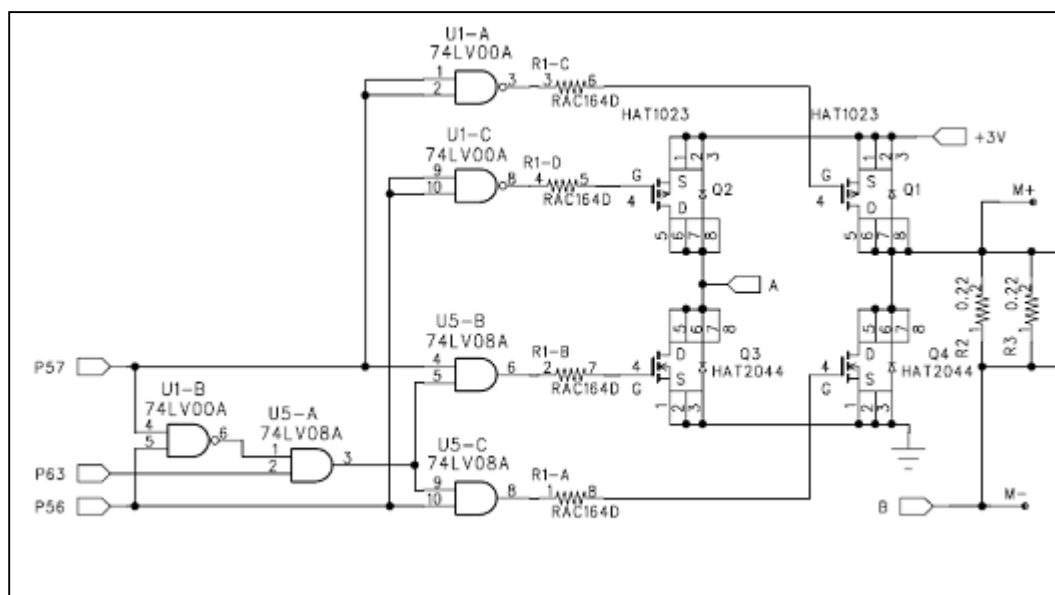


比べ長ければモータは速く回り、短ければモータは遅く回ります。逆回転にしたい場合は SW3 と SW2 で同様の操作をします。つまり H ブリッジ回路ではモータの回転方向、回転速度を自在に変化させることが出来るということです。



先ほどの図は単純にスイッチを使った例としましたが、PUPPY に搭載の回路ではスイッチの代わりに MOSFET を使用してマイコンでスイッチングの制御をしています。SW1、SW3 は N チャンネル MOSFET、SW2、SW4 は P チャンネル MOSFET で置き換え、さらに PWM という方式で SW2、SW4 の ON、OFF を切り替えます。PWM とは Pulse Width Modulation のことで右図のようにパルス幅を変化させることで SW2、SW4 の ON 時間を変化させモータの出力を調整できるようになっています。

PUPPY で実際に用いられている回路は下図のとおりです。



マイコンのポート P 57、P 56 端子はそれぞれ汎用出力ポートに設定してロジックレベルの HI、LO を出力し、P63/FTIOD0 端子からは PWM 波形を出力すると H ブリッジの前段階入っているロジック回路の組み合わせにより表のような動作をします。

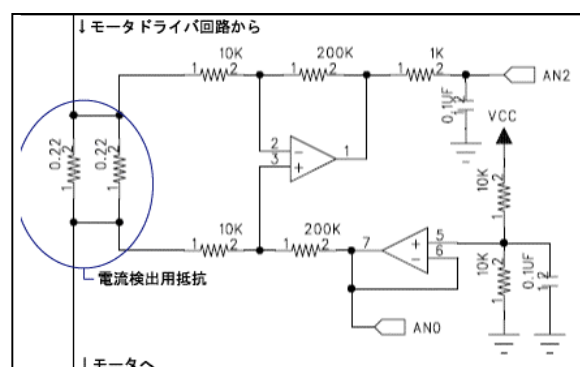
	P56	P57	P63
停止	L	L	PWM
CW	L	H	PWM
CCW	H	L	PWM
ブレーキ	H	H	PWM

6－2．電流センサ回路

倒立走行制御の制御入力はトルクで与えられ、モータが発生するトルクはモータに流す電流に比例します。PUPPY ではモータの制御に PWM 方式を採用していますが、PWM の DUTY 比とモータに流れる電流の間には比例関係のようなものではありません。モータにあるきまった DUTY 比で電圧を印加したとき、モータの出力軸が無負荷のときと、拘束されているときでは、モータに流れる電流は大きく異なります。

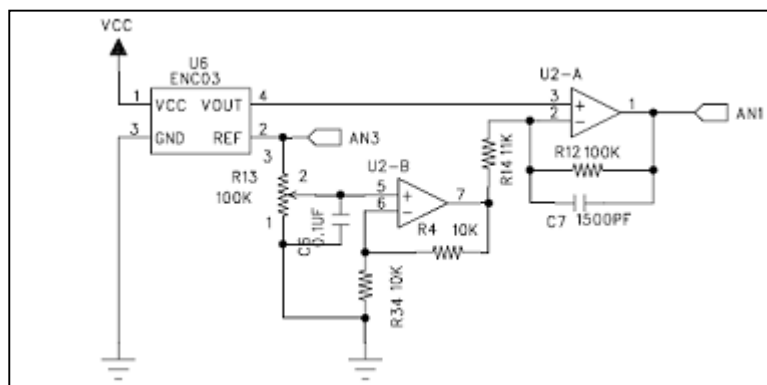
PUPPY では、モータをトルク制御するために、モータに流れる電流を観測しフィードバックする閉ループ系を構成し、倒立走行制御で与えられたトルク(電流値)に速やかに収束するようにしています。

右図は PUPPY 電流センサ回路です。モータに流れる電流は、電流検出用の抵抗を流れますが、そのときに抵抗両端で発生する電圧降下はそこを流れる電流に比例して大きくなるので、この電圧降下を増幅しマイコンの AD 変換器に入力すると電流を観測することが出来ます。



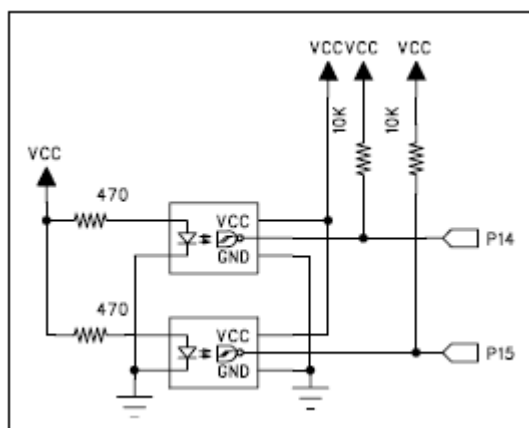
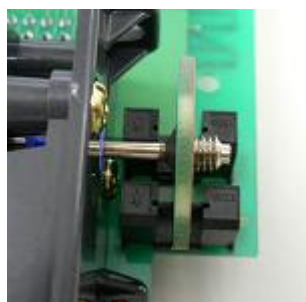
6－3．傾斜角速度検出回路

本体基板中央部に取り付けられている圧電振動ジャイロ(ENC-03M)は車体の傾斜角速度を検出するためのセンサです。傾斜角速度に比例した電圧が約 2.5V (VCC/2) を中心に出力されます。PUPPY ではこの出力電圧をさらに増幅してマイコンの AD 変換器に入力します。



6-4. モータ回転角検出回路

本キットではモータ軸にエンコーダディスクを取り付け、フォトインタラプタ(TLP1033)と組み合わせてフォトロータリエンコーダを構成します。TLP1033 は内部に波形処理回路が内蔵されておりますので外付けの回路は図のように単純なものとなります。出力はそれぞれ外部割込み端子に接続し、一方の割り込み処理中に他方の出力レベルを観測することで正回転、逆回転を判別できるようになっています。



7. サンプルプログラムについて

本キット付属のサンプルプログラムは PUPPY を倒立状態で車輪の速度制御をするプログラムです。DIP スイッチ(SW1)で数パターンの動作モードを切り替えることができます。



スイッチ4	1:位置制御モード	0:速度制御モード
スイッチ3	1:前方 0:後方	1:前進 0:後退
スイッチ2	00: 10cm 01: 20cm	00:2rad/sec 01: 4rad/sec
スイッチ1	10: 30cm 11: 40cm	10:6rad/sec 11: 8rad/sec

動作モードの例

*スイッチ4:1 スイッチ3:1 スイッチ2:0 スイッチ1:1のとき

→PUPPY は制御開始後 20cm 前方に移動しその場にとどまり続けようとしています。

*スイッチ4:0 スイッチ3:0 スイッチ2:1 スイッチ1:1のとき

→PUPPY は制御開始後車輪角速度 8rad/sec の速さで後退します。

サンプルプログラムの倒立制御では一定周期ごとと状態量(本体の姿勢角,車輪の回転角等)を測定し、その値からモータに発生させるべきトルクを計算し出力することで実現できます。プログラムで必要なものは一定周期の生成、状態量の測定およびモータに与えるトルクの計算、モータのトルク制御です。

一定周期の生成

サンプルプログラムでは倒立走行制御と、それを実現するために必要なモータ制御を同時に行っています。倒立走行制御は 10ms 毎、モータ制御は 200 μ s 毎に行っています。

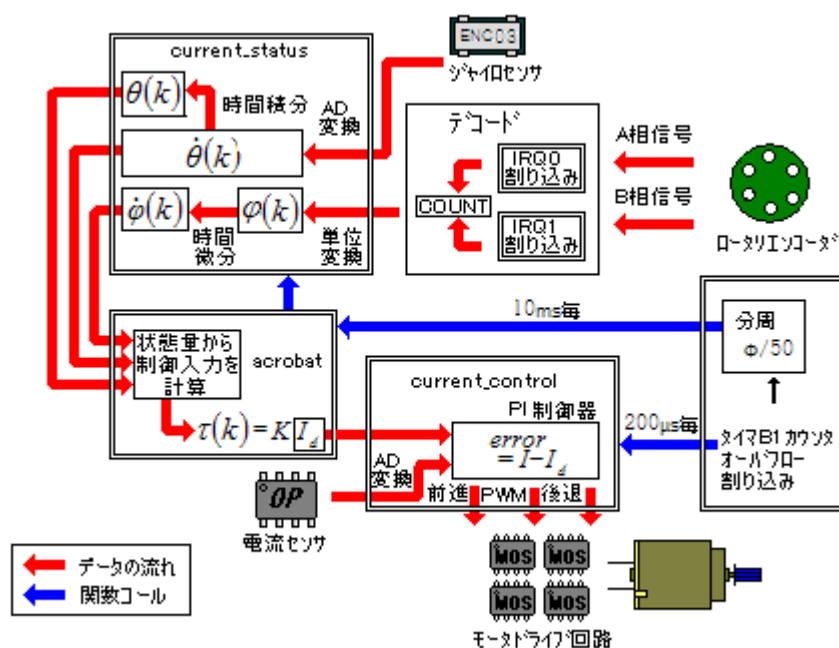
状態量の測定およびモータに与えるトルクの計算

職率走行制御を実施するために必要な状態量は、PUPPY 本体の傾斜角、角速度、車輪の回転角速度です。本体傾斜角速度はセンサの AD 変換値、傾斜角は角速度の積分値、車輪回転角速度はロータリエンコーダパルスによる割り込み処理からそれぞれ求めます。状態量を観測したならそれらを制御入力トルク計算式に代入するとモータに与えるべきトルクが求められます。

モータのトルク制御

制御周期ごとに更新されるトルク指令値をモータに印加します。モータに流れる電流値を観測し、目標とする電流値(=トルク)に速やかに収束するように PI 制御を用いてモータに印加する PWM を増減して電流を制御します。

7-1. H8/Tinyシリーズのサンプルプログラム

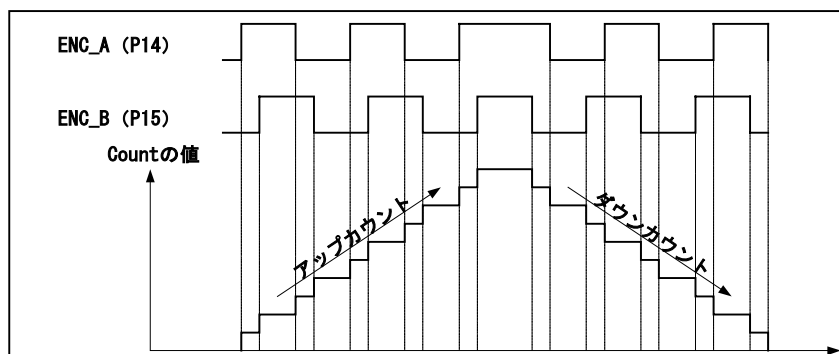


プログラム中の関数について説明します。

int_irq0

IRQ0 端子に接続されたフォトロータリエンコーダ A 相の出力パルスが割り込み要因になったときに呼び出される関数です。車輪の回転に応じてパルスカウンタ(Count)を増減します。図の要領で割り込み要因が立ち上がりエッジのとき B 相が LO のとき、もしくは割り込み要因が立ち下がりエッジのとき B 相が

HI で Count をインクリメント、逆の条件でデクリメントします。割り込み要因を反転させ次の割り込みに備えます。



int_irq1

IRQ1 端子に接続されたフォトロータリエンコーダ B 相の出力パルスが割り込み要因になったときに呼び出される関数です。INT_IRQ0(void)と同様な関数ですが Count の増減要因は逆になります。int_irq0,int_irq1 の2つの関数でロータリエンコーダの A 相,B 相信号をデコードし車輪回転角を検出します。

int_tb1

タイマ B1 のオーバーフロー割り込みで呼び出される関数です。タイマ B1 のカウンタオーバーフローは $200\mu\text{sec}$ ごとに発生し、関数内でモータ制御関数 `control_current` を呼び出します。さらに 50 回に 1 回の割合 (10msec 毎) で倒立制御フラグ (Cflag) をセットします。

current_sutatus

現在の PUPPY の状態量を観測する関数です。PUPPY の倒立走行制御には車体傾斜角 θ 、車体傾斜角速度 $\dot{\theta}$ 、車輪回転角速度 $\dot{\phi}$ が状態量として必要です。車体傾斜角速度については傾斜角速度検出回路の出力を AD 変換して単位変換します。車体傾斜角と車輪回転角速度の観測については少々工夫が必要です。車輪回転角速度は車輪回転角を表す Count 値車体傾斜角は、車体傾斜角速度を時間積分して求めます。車輪回転角速度 $\dot{\phi}$ は車輪回転角を表す Count を時間微分した後その結果を SI 単位系に変換し、あらかじめ制御の設計で用いた変数 $d\text{ThetaG}(\dot{\theta})$ 、 $d\text{Fai}(\dot{\phi})$ に格納しています。また、 ϕ を求めるために $d\text{ThetaG}(\dot{\theta})$ を台形法により時間積分し $\text{ThetaG}(\theta)$ を算出しています。この関数は制御周期ごとに呼び出されることを前提に作られていますのでそれ以外のタイミングで呼び出されると時間微分、積分が正確に行われませんので注意が必要です。

acrobat

倒立走行制御を行うための関数です。フィードバックゲインと PUPPY の現在の状態量を制御式に代入し、モータのトルク指令値を計算します。PUPPY 制御説明書で説明したとおり 1 ステージ遅れの制御を実施しますので、計算結果であるモータ電流指令値 I_d は、次にこの関数が呼び出されたときにモータ制御関数 `control_current` に渡されます。

control_current

モータトルク制御用関数です。電流センサ入力の AD 変換値を SI 単位系に変換し、現在のモータ電流値を表す変数 I に格納します。 I と I_d の差が制御偏差となり、PI 制御で制御偏差がすみやかにゼロに収束、つまり電流値が指令値に収束します。

gyro_offset

傾斜角速度検出回路の出力は角速度がゼロのとき、ENC-03M の内部基準電圧(約 1.2V)となります。本体基板の電源電圧 V_{cc} (安定化 5V)は単三乾電池 2 本(約 3V)を昇圧して作られており、電池の消耗具合によっては若干の変動が見られます。このことより車体傾斜角速度ゼロ時の傾斜角速度検出回路出力の AD 変換結果(GYRO_OFFSET)は必ずしも毎回同じとは限りません。そこでサンプルプログラムでは電源投入ごとに毎回 GYRO_OFFSET を計測することで、より安定した倒立走行制御を実現しています。

wait_m

引数で指定した時間(msec)だけ待機します。この関数はタイマ B1 のカウンタオーバーフロー割り込みごとに立つフラグ(Cflag)を利用していますので割り込みをマスクした状態で実行すると意図しない無限ループに陥りますので必ずタイマ B1 のカウンタオーバーフロー割り込み許可、CCR の割り込みマスクビットをクリアした状態で使用して下さい。

その他の関数につきましては適宜プログラム中のコメントをご覧ください。

7-2. RY3048F-ONEのサンプルプログラム

H8Tiny シリーズのサンプルプログラムではロータリエンコーダから出力される A 相、B 相信号を外部割り込み端子に入れて割り込み処理で信号をデコードしていますが RY3048F-ONE の場合 H8/3048F-ONE 内蔵タイマの ITU (16 ビットインテグレートドタイマユニット) に位相係数モードがあるため TCLKA、TCLKB 端子に A 相、B 相信号を入力するだけで自動的に信号をデコードできます。

制御周期の生成には ITU のオーバフロ割り込みを使用し、モータ制御周期をチャンネル 0、倒立制御周期をチャンネル 1 で生成します。

その他は H8 Tiny シリーズと同様です。

7-3. SH Tiny シリーズのサンプルプログラム

SH7125F の内蔵タイマにも MTU2 (マルチファンクションタイマパルスユニット 2) に位相係数モードがあるため TCLKA、TCLKB 端子に A 相、B 相信号を入力するだけで自動的に信号をデコードできます。

制御周期の生成には CMT (コンペアマッチタイマ) のコンペアマッチ割り込みを使用し、モータ制御周期をチャンネル 1、倒立制御周期をチャンネル 0 で生成します。

その他は H8 Tiny シリーズと同様です。

7-4. R8C シリーズのサンプルプログラム

R8C シリーズでは、ロータリエンコーダのカウント取得に INT 割り込み、制御周期の生成にはタイマ RA のカウントアンダフロー割り込みを使用します。

R8C24/25、2A/2B、2C/2D、35 シリーズマイコンではモータ制御用の PWM をタイマ RD で生成、26/27 シリーズではタイマ RB で生成しています。

7-5. M16C Tiny シリーズのサンプルプログラム

M16C Tiny シリーズでは、ロータリエンコーダのカウント取得に INT 割り込み、制御周期にはタイマ B (Bo)、モータ制御用の PWM をタイマ A (Ao) で生成しています。

RY3048F-ONE の接続には変換基板(PUPPY-RY3048FONE ADAPTER 別売)が便利です。
ワイヤ接続される方は下表のように接続して下さい。

PUPPY側			RY3048F-ONE側		
信号名	信号説明	ピンNo	ピンNo	端子名	端子機能
ENC_A	ロータリエンコーダA相	J2-3	⇔	J2-9	TCLKA ITUクロック入力
ENC_B	ロータリエンコーダB相	J2-4	⇔	J2-8	TCLKB ITUクロック入力
LED4	LED4点灯・消灯	J2-7	⇔	J2-2	PA7 汎用出力
LED3	LED3点灯・消灯	J2-8	⇔	J2-3	PA6 汎用出力
LED2	LED2点灯・消灯	J2-9	⇔	J2-4	PA5 汎用出力
LED1	LED1点灯・消灯	J2-10	⇔	J2-5	PA4 汎用出力
VREF	ジャイロセンサの基準電圧	J2-11	⇔	J1-6	AN3 アナログ入力
CURRENT	電流センサ回路出力電圧	J2-12	⇔	J1-7	AN2 アナログ入力
GYRO	ジャイロセンサ回路出力電圧	J2-13	⇔	J1-8	AN1 アナログ入力
AGND	電流センサ基準電圧	J2-14	⇔	J1-9	AN0 アナログ入力
BATT_M	モータ駆動用電池電圧	J2-15	⇔	J1-5	AN4 アナログ入力
BATT_C	CPU・周辺機能用電池電圧	J2-16	⇔	J1-4	AN5 アナログ入力
SW4	SW4の状態	J1-13	⇔	J3-2	PB7 汎用入力
SW3	SW3の状態	J1-14	⇔	J3-3	PB6 汎用入力
SW2	SW2の状態	J1-15	⇔	J3-4	PB5 汎用入力
SW1	SW1の状態	J1-16	⇔	J3-5	PB4 汎用入力
PWM	モータ駆動用PWM	J1-21	⇔	J3-7	TIOCA4 ITU波形出力
BACK	モータ逆回転	J1-26	⇔	J3-8	PB1 汎用出力
FORWARD	モータ正回転	J1-27	⇔	J3-9	PB0 汎用出力

PUPPY側		RY3048側	
信号名	ピンNo	ピンNo	信号名
GND	J1-1 J1-2 J1-33 J1-34	J1-10 J2-10 J3-10	GND
	J2-1 J2-2 J2-33 J2-34		
VCC	J1-31 J1-32 J2-31 J2-32	J1-1 J2-1 J3-1	VCC

対応マイコンボード

現在 PUPPY 対応のマイコンボードは下記のとおりとなります。

【H8Tiny Base Board シリーズ】

製品型名	実装マイコン型名	価格 (税別)
BB64E3687F	HD64F3687FP	¥5,000
BB64E3687GF	HD64F3687GFP	
BB64E36034F	HD64F36034FP	
BB64E36034GF	HD64F36034GFP	
BB64E36037F	HD64F36037FP	
BB64E36037GF	HD64F36037GFP	
BB64E36054F	HD64F36054FP	
BB64E36054GF	HD64F36054GFP	
BB64E36057F	HD64F36057FP	
BB64E36057GF	HD64F36057GFP	
BB64E36064GF	HD64F36064GFP	
BB64E36077GF	HD64F36077GFZ	
BB64E36079GF	HD64F36079GFZ	

【SH Tiny Base Board シリーズ】

製品型名	実装マイコン型名	価格 (税別)
BB64A71253F	R5F71253N64FP	¥5,000
BB48A71243F	R5F81243N48FP	

【R8C Base Board シリーズ】

製品型名	実装マイコン型名	価格 (税別)
BB52A24xSNF	R5F2124xSNFP	¥5,000
BB52A25xSNF	R5F2125xSNFP	
BB32A266SNF	R5F21266SNFP	
BB32A276SNF	R5F21276SNFP	
BB52A35yANF	R5F2135yANFP	
BB52A35wCNF	R5F2135wCNFP	
BB52A35wDNF	R5F2135wDNFP	
BB64A2AzSNF	R5F212AzSNFP	
BB64A2BzSNF	R5F212BzSNFP	
BB80A2CzSNF	R5F212CzSNFP	
BB80A2DzSNF	R5F212DzSNFP	

- ※ w には内蔵 ROM サイズの違いにより 4,5,6 のいずれかが入ります。
- ※ x には内蔵 ROM サイズの違いにより 4,5,6,7,8 のいずれかが入ります。
- ※ y には内蔵 ROM サイズの違いにより 4,5,6,7,8,A,C のいずれかが入ります。
- ※ z には内蔵 ROM サイズの違いにより 7,8,A,C のいずれかが入ります。

【RY3048F-ONE シリーズ】

製品型名	実装マイコン型名	価格 (税別)
RY3048F-ONE typeH	HD64F3048BF25	¥6,600

※ ワイヤ接続、または別売の変換基板(PUPPY RY3048F-ONE ADAPTER)での対応となります。

【SmartBB!!!36077 学習キット】

製品型名	実装マイコン型名	価格 (税別)
SmartBB!!!36077 学習キット (ボード型名:SMBB64E36077)	HD64F36077GFP	¥9,500 (キット価格) ※ボードのみの販売はございません

※ SMBB64E36077 側ボードの J1,J2 裏面に付属 34 ピンヘッダをご自身で、はんだ付けが必要です。

【HSB16C29S64NE】NEXCESS 演習ボード

製品型名	実装マイコン型名	価格 (税別)
HSB16C29S64NE	M30291FCHP	¥12,381
HSB16C29S64NE(基板のみ)※1		¥11,000
HSB16C29S64NE+BB STUDY I/O ※2		¥18,095

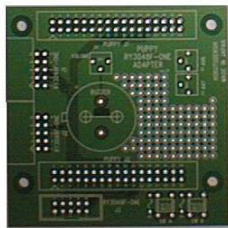
※1 製品内容は M16C/29 実装マイコンボードと回路図のみ

※2 BB SUTDY I/O のみの販売はございません

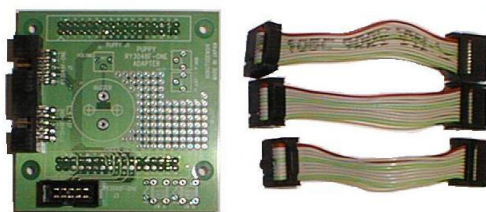
※ご注文時、税別¥10,000 未満の場合別途 発送手数料・送料として ¥1,000(税別)がかかります。また表示価格に消費税額が加算されます。

PUPPY RY3048F-ONE ADAPTER (別売)

基板のみ ¥2,500(税別)

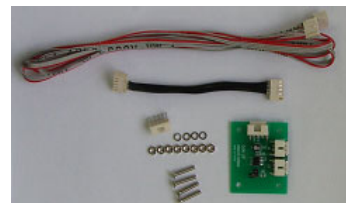


コネクタ実装済み ¥5,000(税別)



※ PUPPY 用 CAN ボード (別売) ¥2,000(税別)

CAN 機能搭載マイコンボードにて CAN 機能をご使用される際にご活用下さい



※ご注文時、税別¥10,000 未満の場合別途 発送手数料・送料として ¥1,000(税別)がかかります。また表示価格に消費税額が加算されます。

SH Tiny、R8C 等順次追加予定 (ファームウェアの変更のみで対応可能) PUPPY に関する最新情報は弊社ホームページでご確認下さい。

パピ－ PUPPY 無償サポート登録のご案内



平素は格別のご高配を賜り、厚く御礼申し上げます。また、この度は弊社 PUPPY をご購入頂き誠に有難うございます。製品変更時の無償サポート対象ユーザ様の登録につき、下記のとおりご案内申し上げます。ご多忙とは存じますが、ご協力の程宜しくお願い申し上げます。

記

E-Mail にてご連絡をお願い申し上げます。この製品のサポートに限り活用させていただきますので下記事項と具体的なご連絡先を頂ければと存じます。

- PUPPY シリアル番号 (製品箱・本体裏面に記載がございます)
- 御社名
- 部署名
- ご担当者様お名前
- 電話番号
- FAX 番号
- ご住所
- E-Mail (ソフトの送付時のみ使用させていただきます)

連絡先：株式会社北斗電子 サポート窓口

e-mail:support@hokutodenshi.co.jp URL:http://www.hokutodenshi.co.jp TEL 011-640-8800 FAX 011-640-8801

以上

パピ－
PUPPY 取扱説明書 © 2005-2014 北斗電子 Printed in Japan 2005 年 12 月 20 日初版発行 REV.3.2.1.0 (140520)

発行 株式会社 **北斗電子**

e-mail:support@hokutodenshi.co.jp(サポート用)、order@hokutodenshi.co.jp(ご注文用) URL:http://www.hokutodenshi.co.jp

TEL 011-640-8800 FAX 011-640-8801 〒060-0042 札幌市中央区大通西 16 丁目 3 番地 7