ヴイストン社製 VS-RC003HV のみで
近藤科学社製
KRS シリーズのサーボを脱力する

Release 3.10

大阪電気通信大学
自由工房 HRP　　関　悠伍

目次

１．はじめに

本資料は、近藤科学会社製の KRS シリーズのサーボ（以下、KO サーボと略します）をヴイストン社製ロボットコントロールボード VS-RC003HV で脱力させる方法を解説します。

KO サーボのうち、一部のものは、VS-RC003HV で使用するにあたり、通常の方法では脱力させることができず、ROBO-ONE、ロボファイトなどのロボットバトル大会をはじめ、ロボットを取り扱う上で危険を伴います。

この問題は、FET（無接点リレー）などを搭載した外部の回路を通して主電源を遮断したりすることで解決できますが、専門の知識を伴う上に、搭載する基盤が増え、ロボットの肥大化につながってしまいます。近藤科学社製の RCB シリーズのボードを用いることでも解決することができますが、RCB シリーズに乗り換えを考えた中、慣れ、機能面等の関係から、やはり VS-RC003HV を使用したいという方もいらっしゃると思います。

VS-RC003HV は、少し難易度が高いものの、設定の自由度が非常に高く、多種多様な取り扱いが可能です。そのため、多少強引な方法となりますが、設定をうまく変更することで KO サーボを脱力させることができます。

本資料では、脱力させる基本的な方法のみ記載しております。本資料に未記載の取り扱いを行う場合は、関連する資料をご確認のうえ、お試しくださいますようよろしくお願いします。

また、KOサーボ以外で脱力することのできないサーボに関しては、前述のように、主電源を遮断するための脱力回路が必要です。そちらに関しましては、別途公開されている脱力回路の資料をご参照ください。

２． 手順

ここからは、VS-RC003HV を取り扱うためのソフトウェアである「RobovieMaker2」の操作方法、KO サーボを取り扱うためのソフトウェアである、ICS3.5 シリアルマネージャーの操作方法をある程度理解しているものとして、説明を進めます。操作方法が不明の場合は、お手数ですが各種マニュアルを見ながら、本資料を読んでいただきますようよろしくお願いします。

2-1 ICS バージョンの確認

KO サーボの ICS のバージョンを確認します。**ICS（バージョンなし）および ICS2.0 規格に準じている KO サーボは従来の方法で脱力できます。ICS3.5 規格及び ICS3.6 規格に準じている KO サーボを脱力する場合、本資料の方法を使用します。また、本資料では ICS3.6 規格は ICS3.5 規格に含まれるもの**として解説します。

以下の表を参考に、使用している KO サーボの ICS バージョンを確認してください。

また、以下の表に含まれていない KO サーボでも、ICS 規格に準じているものであれば脱力することができます。

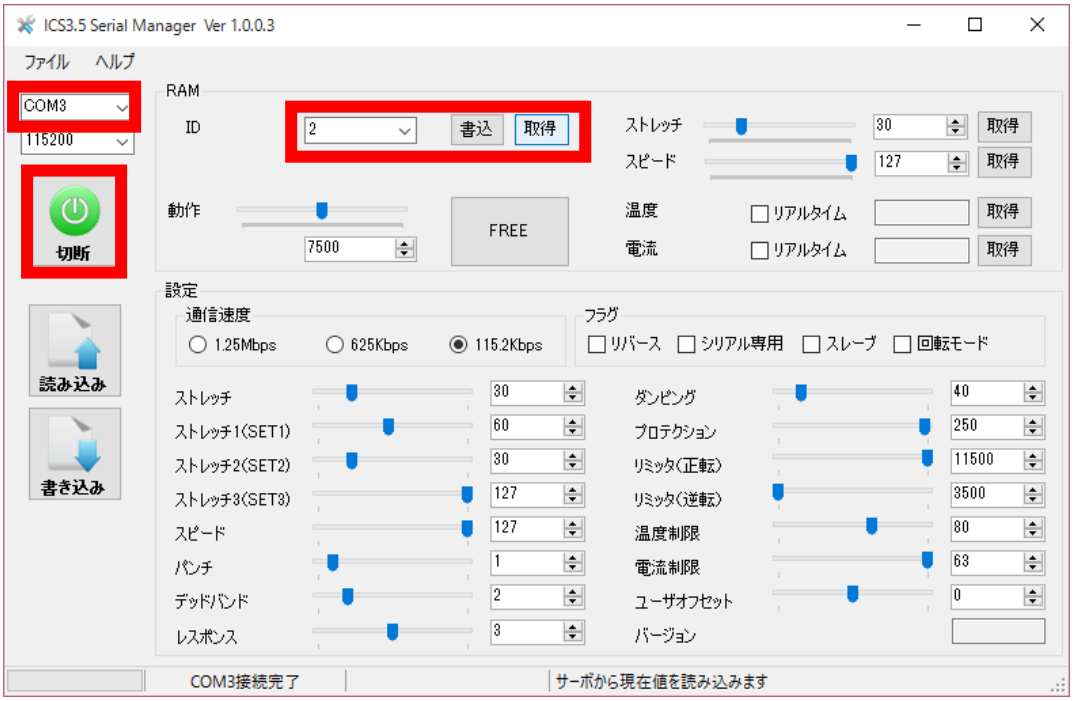
バージョン	サーボ型番	設定を行うのに 使用するソフトウェア
ICS (バージョンなし)	KRS-786 KRS-788HV KRS-2350HV KRS-4024HV	サーボマネージャー for2346REDVersion 又は Heart to Heart 3(J)
ICS2.0	KRS-4013HV KRS-4014HV	ICS2.0 シリアルマネージャー
ICS3.0	KRS-2552HV KRS-6003HV	※PWM 信号に非対応のため 取扱不可
ICS3.5	KRS-403xHV KRS-2542HV KRS-2572HV KRS-2552RHV KRS-6003RHV KRS-3204 KRS-3304	ICS3.5 シリアルマネージャー
ICS3.6 (ICS3.5 に含まれる)	KRS-3301 KRS-6003R2HV	ICS3.5 シリアルマネージャー

2-2 ICS3.5 規格のシリアル専用を解除する

ISC3.5 には「シリアル専用」という設定があるので、それを解除します。

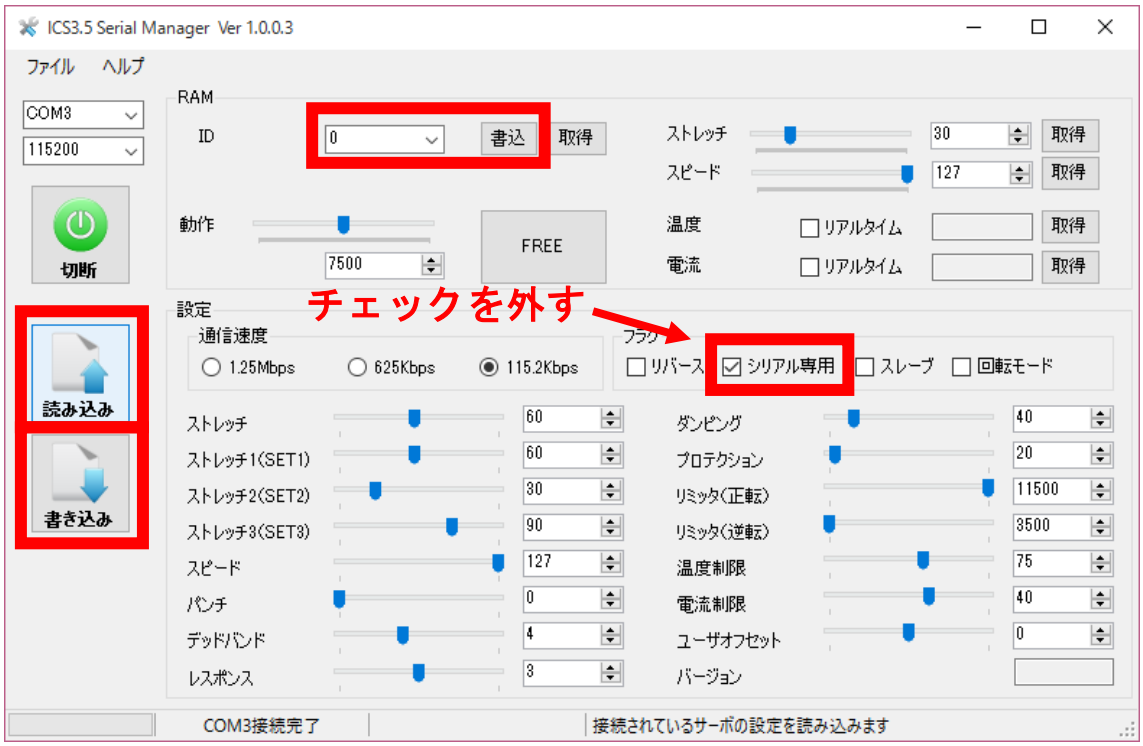
※既にPWM信号で動作している(VS-RC003HVで回転させることができている)場合はこの行程は必要ありません。一部のKOサーボは出荷時にシリアル専用が解除されているものがります。(KRS-3304,KRS-3301,KRS-3204等)

はじめに、ICS3.5 シリアルマネージャーを起動します。任意の COM を選択して接続ボタンをクリックします。赤い電源ボタンの絵が緑色になったら接続できています。次に ID で“取得”を選択して KO サーボと接続してください。接続できれば現在の ID番号が表示されます。



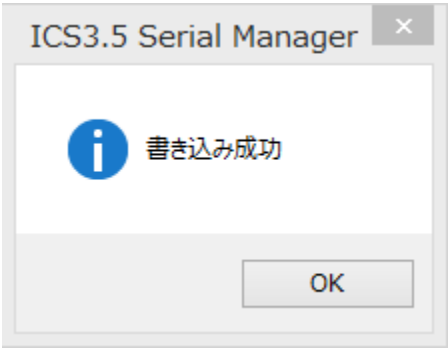
次に ID で 0 を選択して書き込みをクリックします。

さらに画面左にある読み込みボタンをクリックすると、サーボの現在の設定が反映されます。このときに、フラグの「シリアル専用」にチェックが入っている場合はチェックを外して画面左の書き込みを押してください。



書き込みが完了するとポップアップウィンドウが出ます。

以上の操作が終われば設定変更完了です。



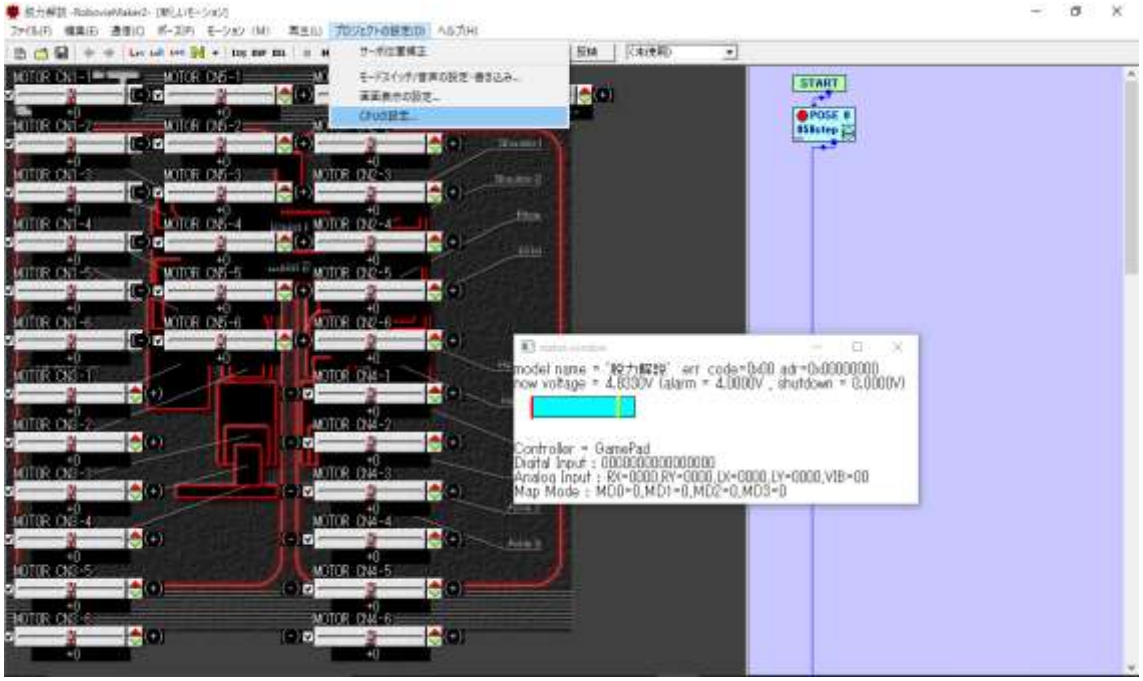
2-3 VS-RC003HV の CPU 設定を変更する

ここらかは実際にサーボをボードに接続した状態での作業になります。設定を複雑に変更するため、電源を接続させて動作させる場合は十分に注意してください。

今回は**サーボモータへ送信される信号の係数を操作しますので、設定途中でサーボにトルクを入れると思いもよらない角度へ動作してしまう場合があります。注意してください。**

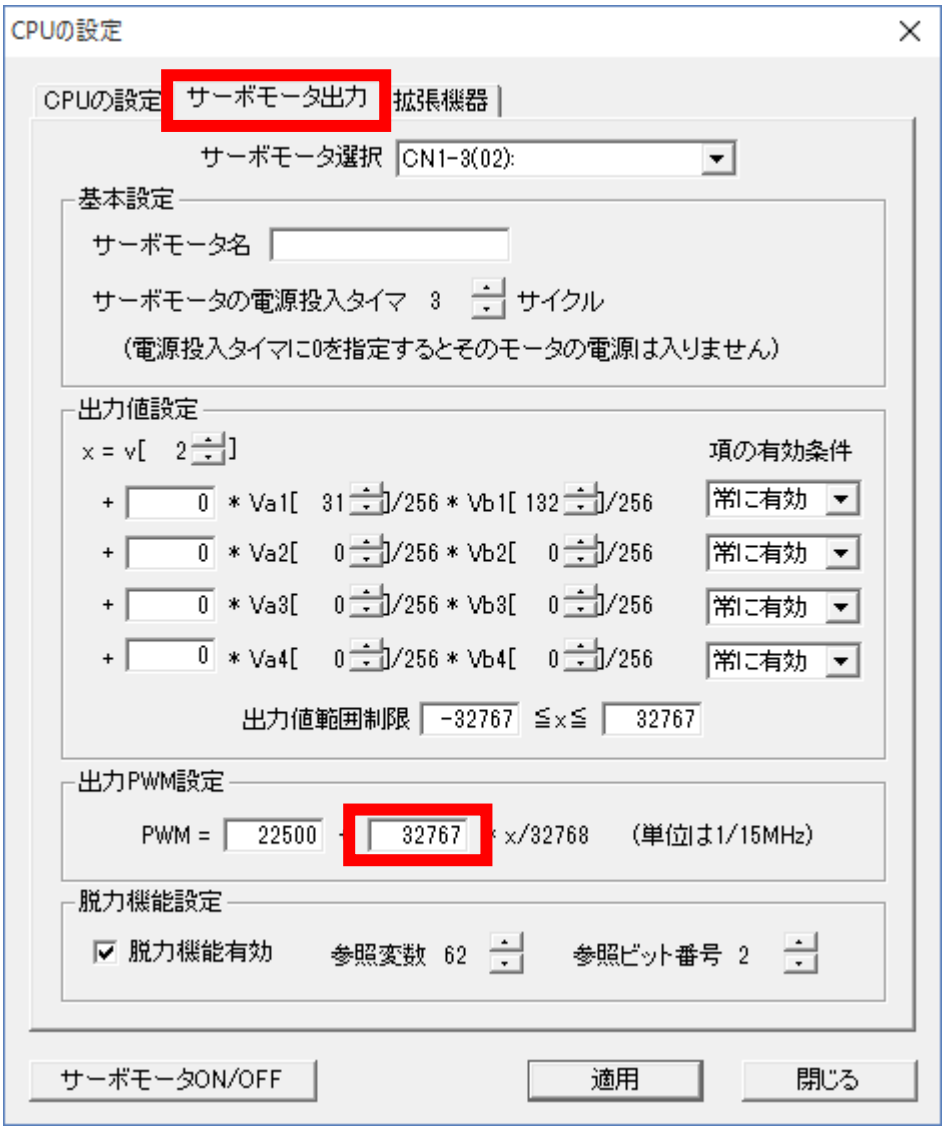
また、新規でロボットを作成される場合と、既存のロボットに脱力機能を追加する場合とで、操作する手順が変わります。**新規でロボットを制作された場合は、サーボモータの位置補正のみを行った状態にしてください。また、ch1-1(00)はできるだけ使用しないでください。**

初めに、RobovieMaker2 を起動し、ボードと接続してください。このとき、サーボに電源は入れないでください。次に、メニューバーのプロジェクトの設定より、CPU の設定を選択してください。



CPU の設定をクリックすると別画面が表示されますので、サーボモータ設定のタブを開きます。サーボモータ選択を KO サーボの接続されている ch に選択します。

ch を変更したら、出力 PWM 設定の部分を変更します。
15000 となっている部分を 32767 に変更します。
設定が完了したら適用をクリックします。



以上の作業を KO サーボが接続されている ch すべてに繰り返し設定します。
この操作を行うと、ポーズスライダなどでの変化量が変わります。
次への行程へは、以下の指示に従って進んでください。

・ロボットが新規作成である⇒2-5 脱力点を計算する
・ロボットが既存のものである⇒2-4 いままでのモーションを使えるようにする

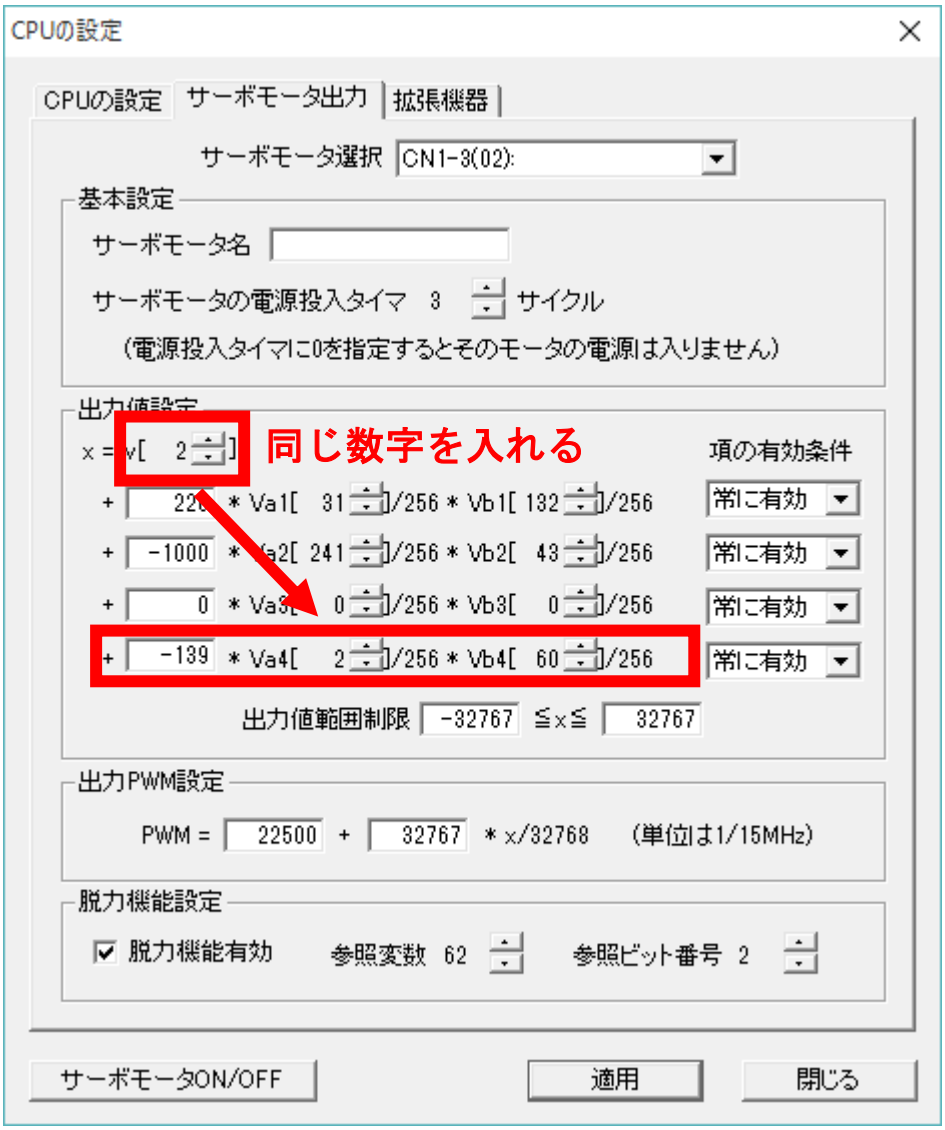
2-4 いままでのモーションを使えるようにする

サーボの動作する変化量が変更されているため、このままでは、これまでに作成してきたモーションが使用できません。そのために、いままでに作ってきたモーションがそのまま使えるように、出力値設定を変更します。

出力値設定の一番下の式に、倍率を元に戻す設定をします。（例として一番下を使っています。一番下を何かで使用している場合は別の段でも構いません。）

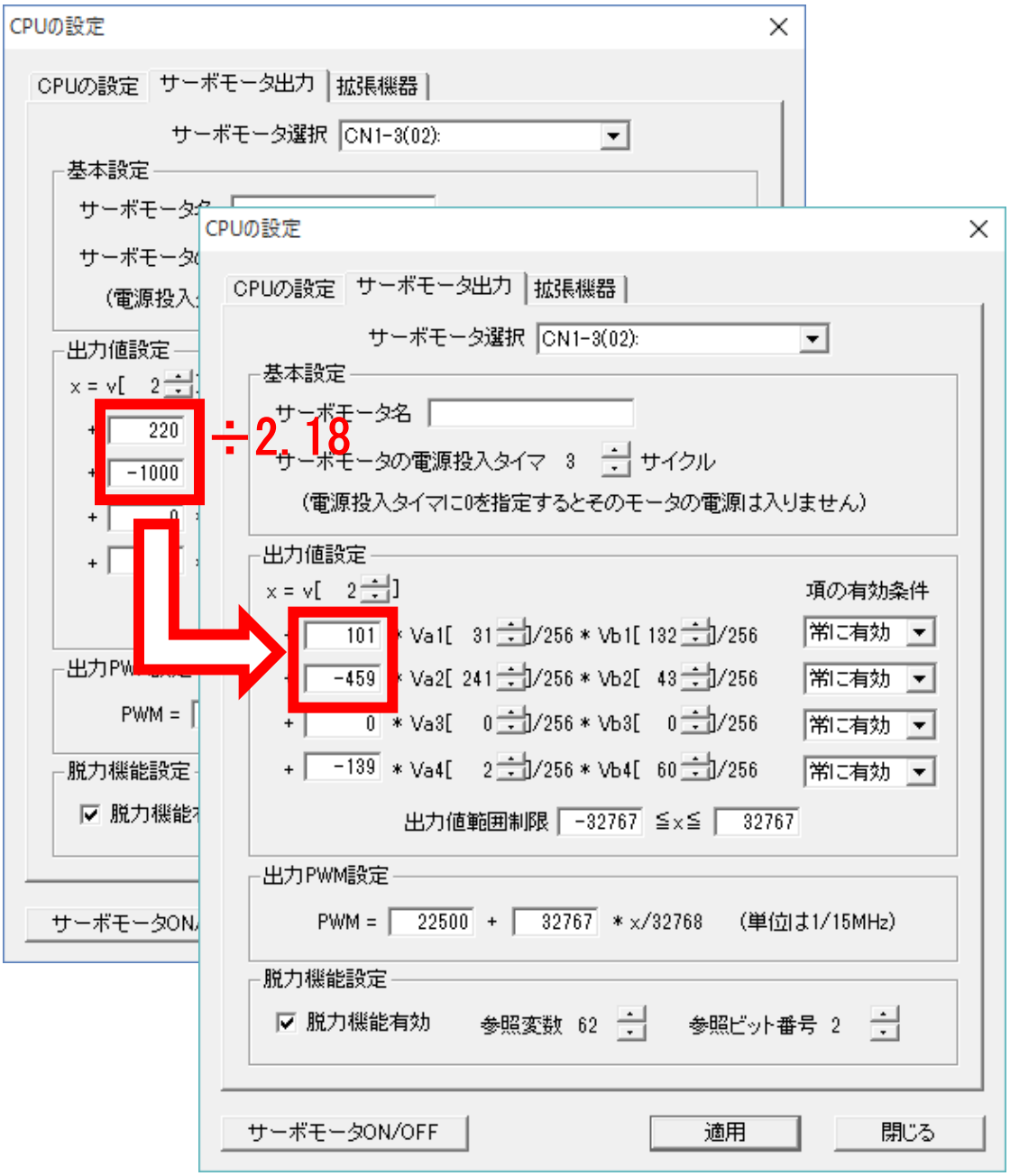
一番下の式の左のテキストボックスに-139 を入力します。

また、Va4[　]を Va4[　(V[　]の中と同じ番号)]に、Vb4[　]を Vb4[60]に設定します。ここでは例として変数 60（ポーズスライダ 60）を使用していますが、何かに使用しているならば、別のポーズスライダでも構いません。



また、ほかの段に何か出力値設定を設けている場合は、各段一番左のテキストボックスに入っている数字を 2.18 で割った値の近似値となる整数を入力します。

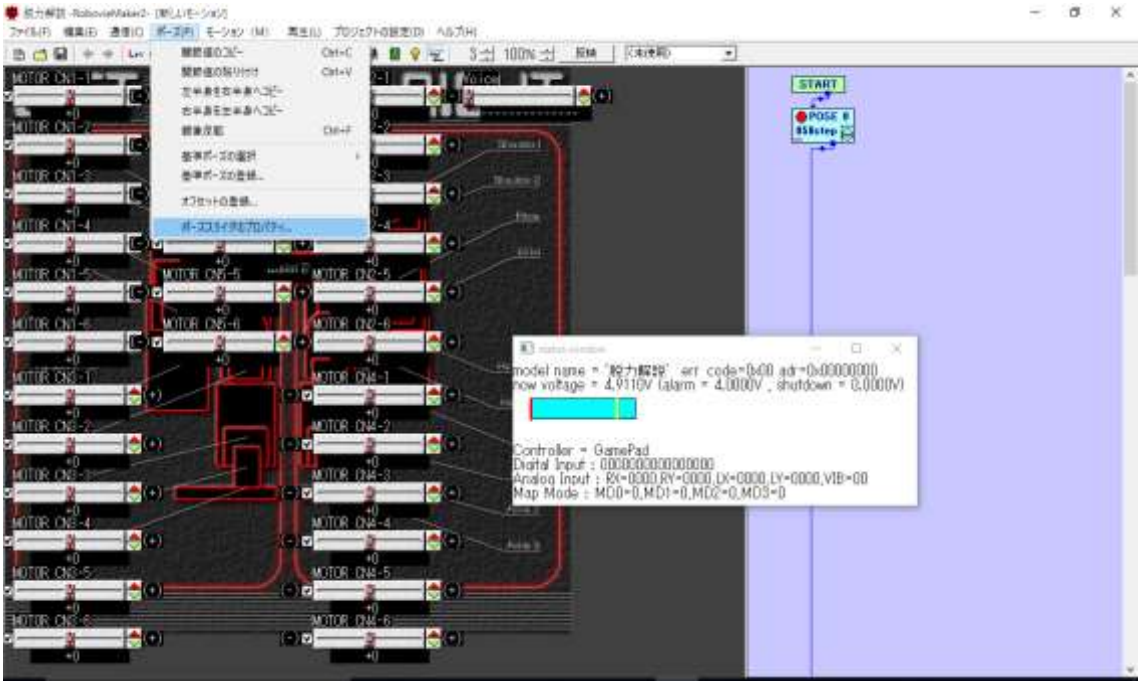
例において、一番上の段では、元が 220 なので、220÷2.18=100.91、これの近似値の整数である 101 を、上から 2 段目では、元が-1000 なので、-1000÷2.18=-458.71、これの近似値の整数である-459 を入力しています。



ただし、ch1-1(00)に限っては、ポーズスライダの変数が 0 のため、この方法は使えません。手間となりますが、ほかの ch に接続して設定を行ってください。

以上を KO サーボが接続されている ch すべてに適応します。

終了したら、適応を押していったん CPU の設定の画面を閉じます。
とじたら、メニューバーのポーズより、ポーズスライダのプロパティを選択します。



ポーズスライダのプロパティを選択したら、別画面が表示されるので、設定するポーズスライダを 60 に選択します。（先ほど 60 以外の変数を指定した場合は、選択した変数のポーズスライダを選択してください。）

設定で、以下の部分を変更してください。

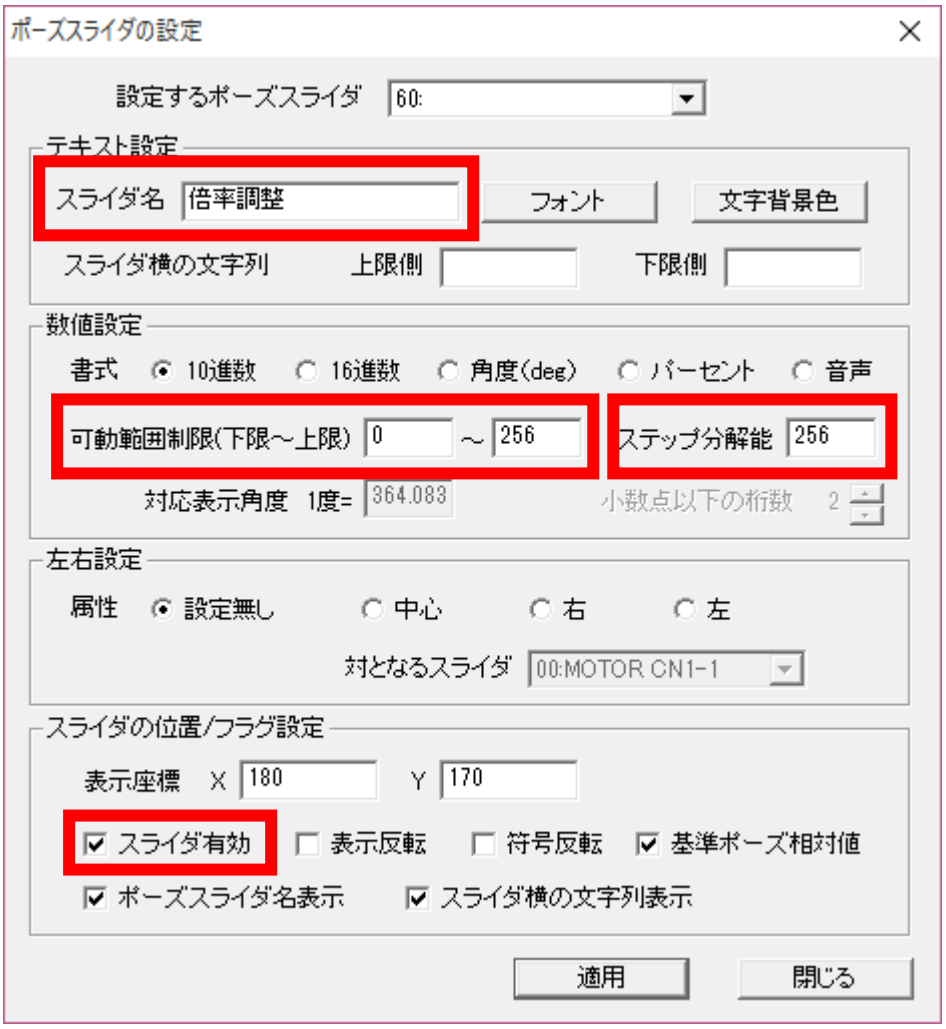
・スライダ名を「倍率調整」に変更します。（自由な名前で構いません。）

・可動範囲制限（上限～下限）を 0～256 に設定します。

・ステップ分解能を 256 に変更します。

・スライダ有効にチェックをいれます。

設定が終わったら適用を押してウィンドウを閉じてください。新しいスライダが表示されます。追加されたスライダは自由な位置に移動してかまいません。



次に、いままで作成したモーションを開きます。いままで作成してきたモーションでは、倍率設定のスライダが使用されていないので、0 になっています。これをすべて 256 に変更して保存します。**今後、この設定を行う前に作成したモーションを編集する際は、この倍率設定のスライダは必ず 256 にして調整を行ってください。**

新しいモーションを作成する場合、倍率設定のスライダを 256 にしなくてもモーション作成は可能ですが、今までの変化量で作成したい場合は 256 にしてください。

2-5 脱力点を計算する

CPU の設定を開いている場合はいったん閉じます。

ここで変更された内容を一度 ROM に書き込んでおいてください。

メニューバーのプロジェクトの設定より、モードスイッチ/音声マップの設定・CPU への書き込みを選択して書き込みます。

この状態で、すでに脱力信号を送れる状態になっています。KO サーボを脱力させるには、ポーズスライダを特定の値に調整します。

しかし、サーボの位置補正の場所によって脱力するポーズスライダの位置が異なります。**本資料では、この“脱力するポーズスライダの位置”を脱力点と呼ぶこととします。**

もう一度、CPU の設定を開き、サーボモータ設定の画面まで移動してください。前回と同様に、サーボモータ選択を KO サーボの接続されている ch に選択します。

ここで、出力 PWM 設定の一番左のテキストボックスの数字を以下の計算式に入れてください。以下の計算式の答えは、別紙などに記入するなどして、あとでわかるようにしておいてください。

脱力点　＝　７５０　－　テキストボックスの数字

例えば、テキストボックスの数字が 22500 だとすると 750-22500=-21750 となります。この-21750 が脱力点です。ポーズスライダを実際に-21750 あたりにするとこの ch の KO サーボは脱力します。

もし、ポーズスライダが左端に到達してしまい、脱力点に設定できない場合は、ポーズスライダの動作範囲が限定されている可能性があります。ポーズスライダの設定を開き、可動範囲制限の下限が-32767 になっているかどうかを確認してください。

それでも左端に達してしまい、脱力点に設定できない場合（結果が-32768 以下になった場合）は、以下の操作を行ってください。

１．ICS マネージャを使用してサーボにリバース設定をする

２．サーボの位置補正をもう一度行う。（リバースによって補正位置がずれます。）

３．もう一度脱力点の計算を行う。（テキストボックスの数字が変わります）

（詳しくは 3-3 をお読みください。）

また、脱力点に達しているのに脱力しない場合は、計算した値が間違っているか、CPU の設定で出力値範囲制限が設定されている場合がありますので、そちらをご確認ください。

以上の手順を KO サーボが接続されている ch すべてに行ってください。

2-6 脱力するか確かめる

実際に脱力するかを確かめます。CPU の設定を閉じてください。

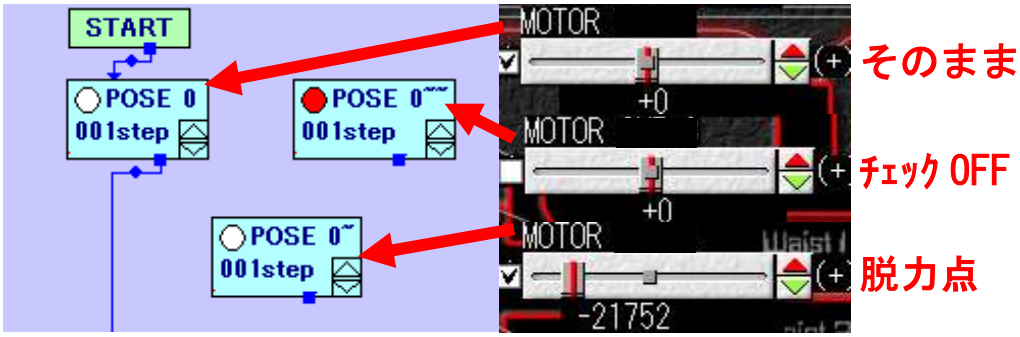
モーションを新規作成します。**このときサーボに電源は入れないでください。**

ポーズブロックの追加を 2 回行ってブロックを 3 つにします。**すべて、速度を 001step にしてください。これを行わないと誤動作の原因となります。**

1 つ目の POSE はなにも変更しません。

2 つ目の POSE はポーズスライダ左のチェックをすべて OFF にします。

3 つ目の POSE は各 ch のポーズスライダを、先ほど算出した脱力点の近似値に設定します。ステップ分解能が大きすぎて近似値から離れてしまう場合は ctrl を押しながらスピンボタンを押すと微調整ができるので近似値に調整してください。（3 つ目の POSE で **ICS3.5 以外のサーボのチェックはすべて OFF にしてください。**）



また 2-4 を行った場合は、脱力点の POSE のみで、“倍率調整” のポーズスライダを 0 にしておいてください。それ以外の 2 つは前回通り 256 にしておきます。

以上の 3 つの POSE ができたら、“そのまま” の POSE を選択してください。ここでサーボに電源を入れます。急に動作してもサーボがロックしないように、メニューバーの通信のサーボモータ ON より、サーボの電源を ON にしてください。

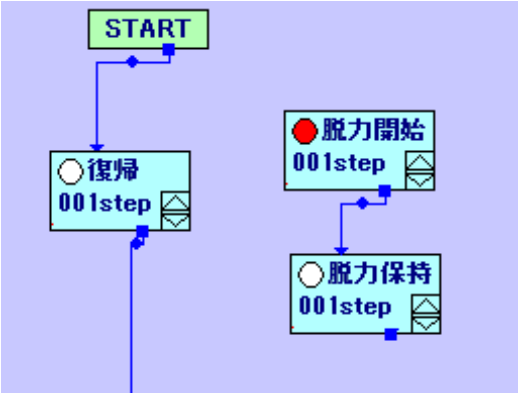
ここまでは普段と同じです。通常通りサーボに電源が入ると思います。

サーボに電源が入った状態で、“脱力点” の POSE を選択し、すぐに“チェック OFF” の POSE を選択してください。“脱力点” の POSE を選択した状態で数十秒以上待機しないでください。誤動作が発生します。

POSE を切り替えたら、接続しているサーボすべてが脱力していることを確認してください。うまく脱力できていれば成功です。また、“そのまま” の POSE を再度選択したときに、サーボが復帰することを確認してください。

2-7 脱力モーションのプリセットを作る

さきほど作成した POSE の名前をわかりやすくしておきます。POSE をダブルクリックするとプロパティが表示され、名称が変更できるので、わかりやすい名前に変更してください。例では、“そのまま”の POSE を“復帰”、“チェック OFF”の POSE を“脱力保持”、脱力点の POSE を“脱力開始”としました。



以上のモーションができたら保存します。

例として、保存名は「脱力プリセット」としました。

以後、脱力動作を行いたいときは保存したこのモーションをインポートすることで、容易に脱力できるようになります。

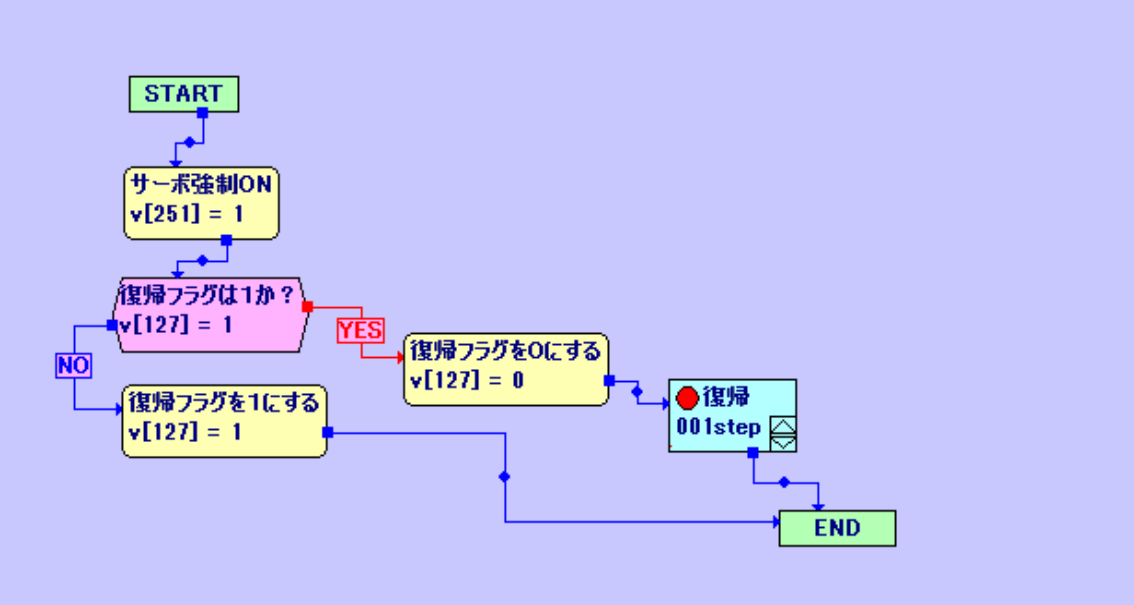
この POSE はすべて 001step で使用してください。

2-8 START＋SELECT で脱力できるように設定する

START＋SELECT を押すと脱力し、再度 START＋SELECT を入力するとアイドリングモーションに復帰するように設定を行います。

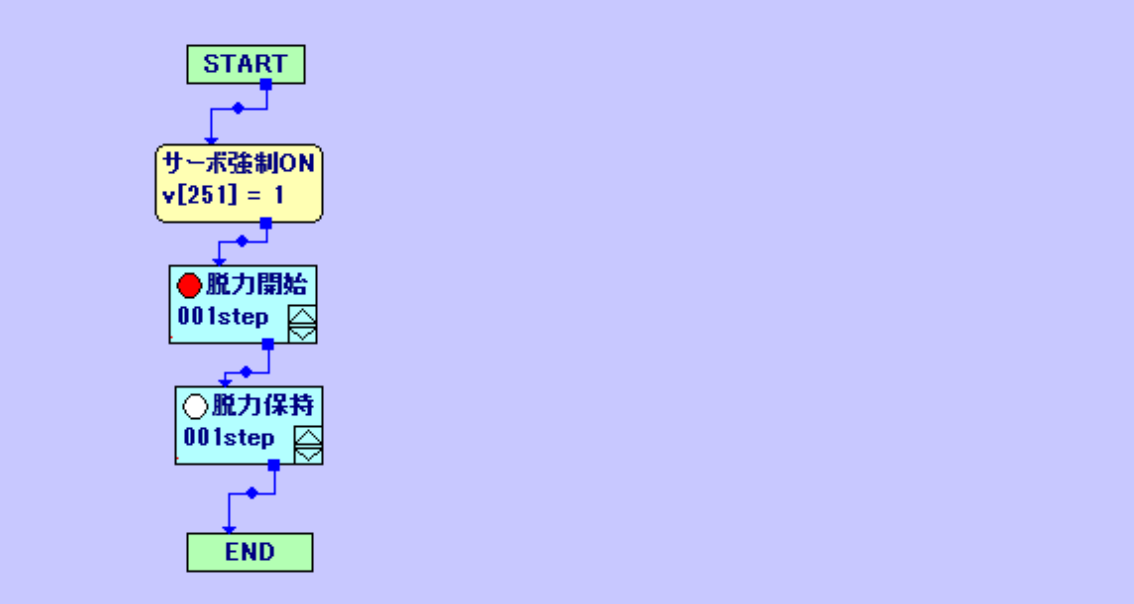
以下のモーションを作成し、保存してください。（変数 127 が使用されているときは他のユーザー変数でも構いません。）

例として、保存名は「動作書き換え」としました。



また、以下のモーションを別で作成し、保存してください。

例として、保存名は「脱力」としました。



以上の 2 個のモーションを操作マップに登録します。（例として、操作マップ v2 での登録方法で解説を行います。）

以下の条件でモーションを登録してください。

「動作書き換え」→参照変数 251 が“0”と「閾値＝」となったときに再生

「脱力」→参照変数 127 が“1”と「閾値＝」となったときに再生

また、両方のモーションで割込キャンセルと Idling 割込キャンセルを○有効に設定して下してください。

（※この2つのモーションの受入キャンセルは○有効にしないでください。）

さらに「動作書き換え」を並び順変更で 1 番上に、「脱力」を上から 2 番目に設定します。

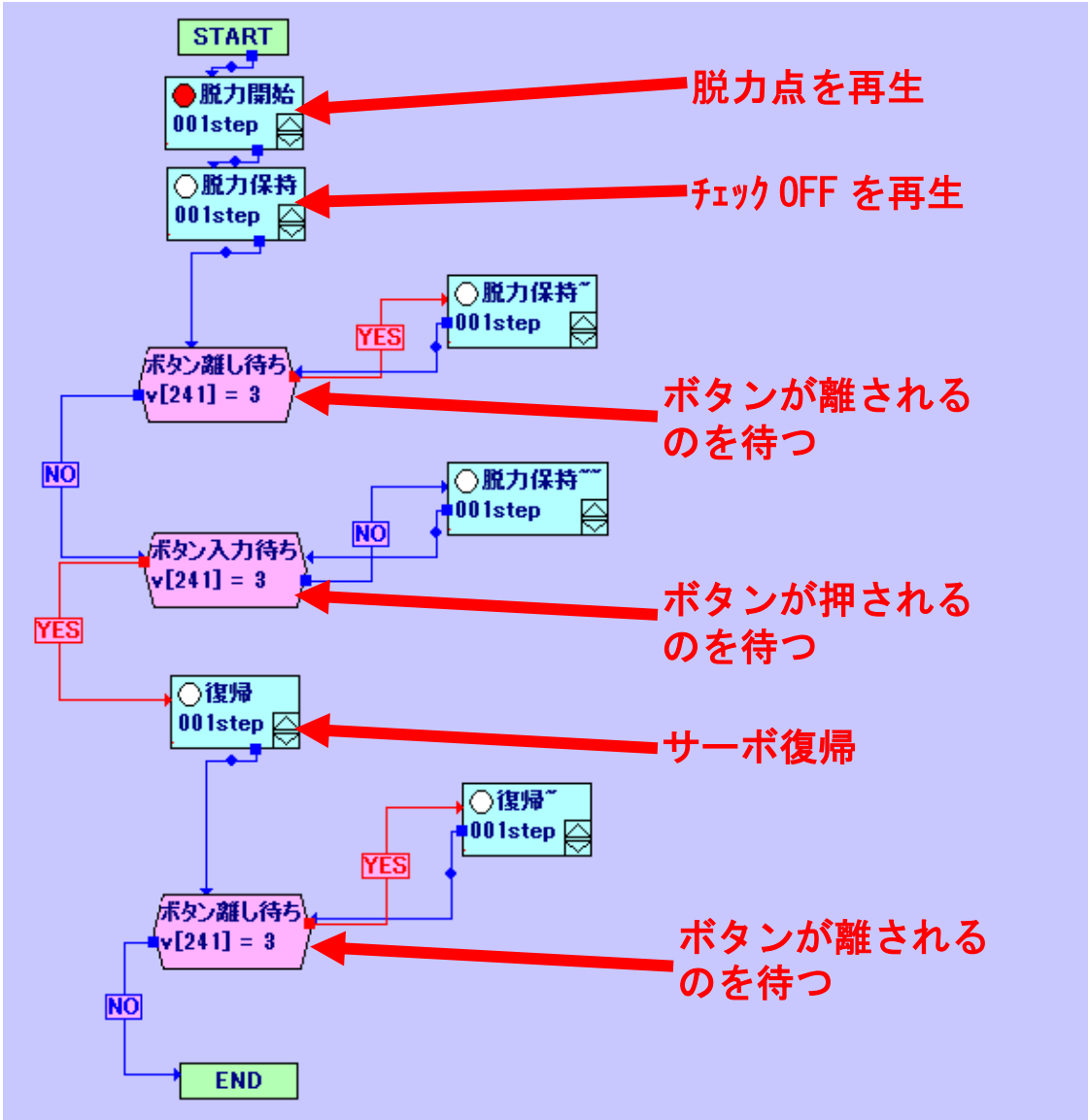
以上の設定を使用しているマップすべてに適用してください。設定が完了したらマップを保存し、VS-RC003HV へ書き込みを行います。

書き込みが完了したら、パソコンとの接続を終了し、実際にコントローラの START＋SELECT の入力で脱力・脱力から復帰できるかどうかを確認してください。

うまく動作すれば設定は終了です。

2-9 START＋SELECT 以外での脱力モーションの例

以下に START＋SELECT 以外のモーションで脱力するモーションの例を説明します。これはゲームパッドの R2 と L2 が同時押しされる（変数 241 が 3 に変化する）と脱力し、いったん離してもう一度 R2 と L2 を入力すると復帰する、というモーションです。この脱力モーションをメインに使用して脱力を行う場合も、誤動作防止のため 2-8 のモーション設定は必ず行ってください。



以上で脱力モーションは完成です。

以上のモーションは、操作マップ V2 に登録する際、START＋SELECT 以外に登録するようにしてください。START＋SELECT では使用できません。

３． 知識

この項では、KO サーボを脱力するにあたって、実際にどのようなことをしているかを具体的に説明します。この項は、読まなくても支障はありませんが、どのようなことをしているかが気になる、興味があるという方は是非ご参照ください。

3-1 ICS について

ICS というのは「Interactive Communication System」の略で、近藤科学社独自のデータ通信規格です。最近主流となっている KRS シリーズはすべてこれに準じて動作します。この ICS にはバージョンが存在し、各バージョンによって操作できる内容が異なり、一般のサーボモータでは変更できない部分を操作することができます。

この ICS はバージョン 3.0 を除き、PWM で動作させる場合、通常のラジオコントロール帯域（ニュートラル（±0 度の位置）1500μsec、700μsec~2300μsec）で動作するほか、通常では使用しない帯域を使用することで、リアルタイムでのサーボの特性をあらかじめ設定しておくことのできる 3 種類から変更したり、現在のサーボの角度位置を取得したりすることができます。

今回は、この「現在のサーボの角度位置を取得」する機能の 1 つである教示機能を使って脱力しています。この教示という機能は、近藤科学社製の RCB シリーズのボードを取り扱うソフトウェアである Heart to Heart において、ロボットを脱力させた後に、特定のポーズをとらせ、その状態で各サーボモータの位置情報を取得してモーション作成を行う際に使用します。今回はこの脱力する部分を使用します。

50μsec±5μsec の信号を送るとサーボがパワーリダクション（脱力）します。また、この状態から 700μsec~2300μsec の信号を送信すると信号に対応する位置に復帰します。**本資料では、この 50μsec±5μsec を脱力パルスと呼びます。**

各 ICS 規格に関してのコマンドリファレンスマニュアルは、近藤科学社ホームページで公開されています。詳しくはそちらのほうをご参照ください

3-2 RobovieMaker2 の CPU 設定

RobovieMaker2 では、CPU 設定において各サーボモータに様々な値をリアルタイムで加算するなど、送信する信号を詳しく設定することができます。これは、VS-RC003HV で制御できるサーボ数 30ch すべてに個別に設定できます。

また、ここでは出力 PWM 設定という項目もあり、ここでは実際に出力される PWM のデューティ比にかかわる設定を自由に行うことができます。

実際に出力される信号は

$$\text{出力されるPWM信号} = (\text{初期位置}) + \frac{(\text{任意の定数})}{32768} \times \text{出力値設定の値}x$$

となります。単位は 1/15MHz で、これは答えを 15 で割ることで、μsec として考えることができます。“任意の定数”は初期で 15000 に設定されており、“初期位置”は初期設定で 22500 となっています。“初期設定”はサーボの現在位置の補正を行った際に変化します。“任意の定数”はこの CPU 設定の画面で自由に指定します。ゆえに PC と接続しているときにか変更できません。“出力値設定の値 x ”はサーボのポーズスライダや加算している変数の値で、ロボットがモーションで動作しているときも PC でコントロールしているときも変数の指定によっては変更され続けます。

例として、初期状態での状態を式に表すと、

$$\text{出力されるPWM信号} = 22500 + \frac{15000}{32768} \times \text{出力値設定の値}x$$

となり、仮に出力値設定の値 x が 0 になっているならば出力される値は 22500、つまりこれを１５で割ると、ラジオコントロールにおいてサーボのニュートラル（±0 度の位置）である 1500μsec が送信されていることになります。

3-3 VS-RC003HV より脱力パルスを送るには

では、KO サーボを VS-RC003HV で脱力させるにはどうすればよいかですが、ここでもう一度、前節の式を考えます。

$$\text{出力されるPWM信号} = (\text{初期位置}) + \frac{(\text{任意の定数})}{32768} \times \text{出力値設定の値}x$$

この式で出力される PWM 信号の部分を操作して 50μsec、すなわち 750 にすることができれば KO サーボは脱力します。“初期位置”はサーボのラジオコントロール帯域である 700μsec~2300μsec、つまり 10500～34500 である可能性が考えられます。これはサーボの初期位置なので、各関節によって異なります。任意の定数は 15000 なので、これを式に考えると

サーボが逆転限界値（700μsec）に補正されている場合、

$$750 = 10500 + \frac{15000}{32768} \times \text{出力値設定の値}x$$

サーボが正転限界値（2300μsec）に補正されている場合、

$$750 = 34500 + \frac{15000}{32768} \times \text{出力値設定の値}x$$

となり、出力値設定の値 x が、逆転限界値において約-21300、正転限界値において約-73728 となれば脱力します。

しかし、VS-RC003HV では、各変数には-32767～32767 までの数字しか扱うことができません。ゆえに正転限界値において出力値設定の値 x を-73728 に設定することは不可能であるわけです。これを回避するために、“任意の定数”を変更します。

出力値設定の値 x を限界まで小さくしたとすると、

$$750 = 34500 + \frac{\text{任意の定数}}{32768} \times -32767$$

となり、任意の定数は約 33751 以上であればよいとされます。しかし、変数は 32767 までなので、この状態でもサーボの補正位置によっては KO サーボを脱力することができません。具体的な角度としては、可動範囲が 270 度の KO サーボにおいて+124 度～135 度の位置に補正されていると脱力できません。

この問題を回避するためには、ICS の機能でサーボにリバース設定をかけて、正負を逆転させます。ISC でサーボにリバースを設定した場合は、受け取った信号と逆にサーボが動作するものの、脱力するパルスが 50μsec であることに変わりはないので、+124 度～+135 度の位置に補正するときの数値が、-124 度～-135 度の数値となり、脱力できる範囲になります（実際には、+124 度～+135 度の位置に補正することはまりないので問題ないと思われます）。2-5 でリバースをかける場合の設定があるのはこのためです。

したがって、任意の定数は 32767 に設定します。このようにして送信できるパルスのデューティ比の範囲を変更して KO サーボの脱力を可能にします。

3-4 脱力 POSE の後にチェックボックスをはずす理由

2-6 で脱力点を送信したあとにチェックボックスを外した POSE を入れている理由を解説します。

モーションを作成している中でお気づきになられた方もいらっしゃると思いますが、脱力点の POSE を数秒間保持していると、サーボが誤動作を発生するようになります。

本来、KO サーボが近藤科学社製 RCB シリーズのボードから脱力パルスを受け取るのはほんの一瞬であり、脱力パルスが長時間にわたって連続で送信されてくることが想定されていません。これが原因で、KO サーボが受け取った脱力パルスが、別の信号と解釈され処理されることにより、意図しない動作を発生させます。この動作は、脱力していると思いロボットに触れた際などに危険を及ぼす可能性があります。

チェックボックスを外すということは具体的に何をしているかというと、パルスを OFF にしています。一般の二足歩行ロボット用のデジタルサーボはパルスが OFF になると脱力します。KO サーボも ICS（バージョンなし）と、ICS2.0 ではチェックボックスを外すだけで問題なく脱力します。しかし、ICS3.5 及びでは、パルスが OFF になっても現在の状態を保持し続け、変化を起こしません。

今回の誤動作の解決は、この“現在の状態を保持し続け、変化を起こさない”というのを逆に利用します。瞬間的に脱力点の POSE を送信し、すぐさまパルスが OFF になる POSE へ切り替えることで、誤動作を防止します。また、この POSE を作ることで、KO サーボ以外のサーボの脱力も忘れることなく同時に行うことができます。

以上がチェックボックスをはずした POSE 入れている理由です。

3-5 いままでのモーションを使用可能にする設定

2-4 について説明します。前述のとおり、サーボの動作する変化量が変更されているため、これまでに作成してきたモーションが使用できません。厳密には、出力 PWM 設定で 15000 だったところを 32767 に変更しているので、動作量は約 2.18 倍になっています。

これはポーズスライダの値を出力値設定で戻すことによって回避しています。

出力値設定の式を取り出すと、

$$\text{出力値設定の値}x = \text{ポーズスライダの値} + (-139) \times \frac{\text{ポーズスライダの値}}{256} \times \frac{\text{倍率調整スライダの値}}{256}$$

となっています。この-139 という数字は-1×256×(1.18/2.18)の近似値を示しています。-1 はポーズスライダの値を反転させる役割、256 は分数の“ポーズスライダの値”の下にある 256 を除去する役割、(1.18/2.18) は 2.18 倍されているものを 1 倍にもどすような数値にポーズスライダの値を調整する役割があります。倍率調整スライダの値はステップ分解能の設定で 0 か 256 しか入らないようになっています。ゆえに、一番右の分数は 0 か 1 になります。

いままでのモーションを再生するときは、この一番右の分数を 1 にすることで、この式が加算され、ポーズスライダによって動作する量が疑似的に今までと同じなるので、以前まで動作していたモーションが使用可能になります。

脱力するときのみ、この分数を 0 にして、ポーズスライダの倍率調整をなくし、送れるパルスの帯域を増やします。

これらの操作を行っているため、ポーズスライダの値が格納される変数が呼び出せない ch1-1（変数 0）は使用することを避けているというわけです。（出力値設定おいて、変数 0 を入力すると、その場所には定数 0 が代入される仕様となっているため。）どうしても使用したいという場合は、ほかのポーズスライダの数字を設定することで使用することもできます。

また、ほかの段に何か出力値設定を設けている場合に、各段一番左のテキストボックスに入っている数字を 2.18 で割ったのも、おこなった計算は違いますが、上記と同じ理由で、サーボの動く動作量を疑似的に元に戻すためです。

3-6 サーボを個別で脱力するには

いままでのモーションを使用可能にした場合（2-4 の手順を行った場合）が該当します。今までのモーションを使用可能にするために、ポーズスライダの値を疑似的に元に戻していました。しかし、この設定を ON・OFF しているポーズスライダ 60（変数 60）はすべての KO サーボに共通なので、一部のサーボにトルクを入れたまま、一部のサーボのみ脱力することができません。

これを解決する方法は何種類かありますが、ここでは 2 つ紹介します。

1 つ目は脱力点を送る瞬間のみ、トルクを保持したままの KO サーボのポーズスライダのチェックを外すということです。脱力点では脱力する KO サーボに脱力点を送信し、脱力しない KO サーボのチェックを外します。次に、チェックをすべて OFF にしていた POSE で、脱力していないサーボのチェックを入れることで、脱力していない KO サーボを動作させることができます。

2 つ目は、ポーズスライダ 60（変数 60）をすべての KO サーボに与えるのではなく、すべての KO サーボに個別に変数を与えるという方法です。ここで与える変数は、ユーザー変数でも自由に使えるポーズスライダの変数でも構いませんが、モーションの編集のしやすさを考慮するとポーズスライダのほうがいいでしょう。しかし、ポーズスライダの本数が倍になってしまうという難点もありますので、ここで使う変数については各環境に合わせて選んでください。

ほかにも数種類の方法がありますが、ここでは省略します。

3-7 未設定で START＋SELECT を押すとどうなるか

本資料の方法を適用せずに START＋SELECT を入力するとどのような問題が発生するかを解説します。VS-RC003HV はコントローラの START＋SELECT を押すことでサーボモータが OFF になります。この操作を行うと、再度 START＋SELECT が入力されるまで、一切のサーボへの信号の出力を停止します。つまり、これを ICS3.5 規格のサーボを含んだ状態で行うと、「サーボそのものは脱力していないのに、ボードが脱力状態に入っている」という矛盾状態になり、一見無線が全く効かなくなったかのような状態になります。とくに、全身 ICS3.5 規格の KO サーボのみで製作したロボットは、通常状態とこの矛盾状態の見分けがつきません。START＋SELECT を押せば復帰しますが、ロボットバトル大会などの試合中にこの状態に陥ると、この症状に気付かず、10 カウント以内に復帰できずにスタンディングダウンやノックアウトになってしまう場合があります。本資料の方法を適用する前に、「急に無線コントロールが効かなくなった」という経験がある方は、慣れなどの関係で、START＋SELECT を無意識に入力してしまっていた場合があります。

3-8 誤動作の無効化と脱力モーションの再生

3-7 による誤動作を回避するには、START＋SELECT のボタン入力を無効にすればよいということになります。VS-RC003HV は、START+SELECT が入力されると変数 251 が 0 の場合は 1 に、1 の場合は 0 に変化します。この変数 251 はサーボモータの ON・OFF を管理しています。ここが 0 の場合、サーボモータに PWM 信号は送信されず、操作マップに設定されているモーション再生のためのコントローラの入力が無効になります（3-7 で前述した“一切のサーボへの信号の出力を停止“）。従って、コントローラで START＋SELECT に脱力モーションを登録し、それを再生しようと思っても、変数 251 の変化が優先されるため、モーションは再生されません。

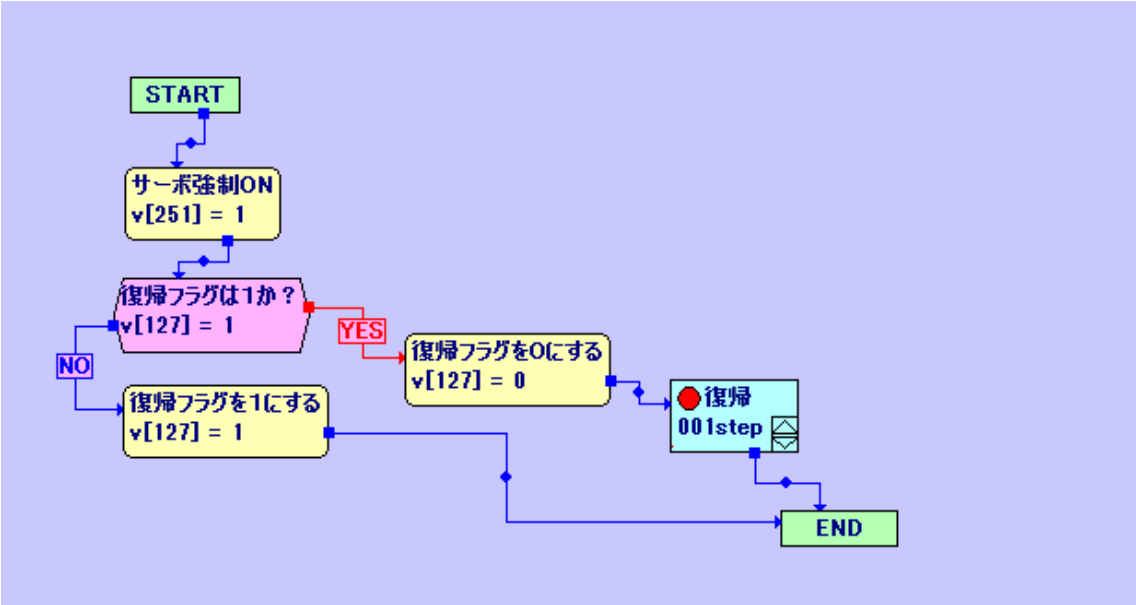
本資料の方法では、ロボットが動作中に START＋SELECT を受けとると、変数 251 が 0 に変化し、動作書き換えモーションが強制的に再生されます（コントローラ入力によるモーションの再生は無効であるが VS-RC003HV 自身が自らの変数変化でモーションを再生することは可能である）。動作書き換えモーション内で VS-RC003HV 自身が変数 251 を 1 に上書きすることで、自動的に脱力状態から復帰します。これにより、変数 251 の変化による 3-7 のような誤動作は事実上無効にできたということになります。

さらに、変数 127 を復帰フラグとして設け、ロボットがどの状態で、STRAT＋SELECT が入力されたかを管理し、状況に応じて脱力モーションの再生・脱力からの復帰処理を行っています。

以上の２つを重ねて動作させることにより、事実上 ICS3.5 規格のサーボを含んだ状態でも START＋SELECT での脱力を実現しています。

3-9 動作書き換えモーションの動作原理

誤動作書き換えモーションの動作について解説します。



まず、電源投入時にユーザー変数 251 は 0 に設定されるため、いきなり「動作書き換え」モーションが再生されます。また、通常の状態（アイドリングモーション再生中や普通のモーション再生中）では START＋SELECT が入力されると変数 251 が 0 になり「動作書き換え」モーションが再生されます。

モーションが再生され、≪サーボ強制 ON≫で、変数 251 が強制的に 1 になります（電源投入時は、この時点でまだアイドリングモーションすら再生されていないため、一瞬サーボにトルクが入るというような動作は発生しません）。

次に、電源投入時にユーザー変数 127 は 0 に設定されているため、≪復帰フラグは 1 か？≫は NO となり≪復帰フラグを 1 にする≫へ進みます。復帰フラグである変数 127 が 1 になると、操作マップ v2 の設定により自動で「脱力」モーションが再生され続けます。

ここで START＋SELECT を入力すると変数 251 が 0 になり、変数 127 は 1 のままですが、操作マップ v2 で優先度が高い「動作書き換え」モーションが割り込みで再生されます。

≪サーボ強制 ON≫で再び変数 251 を 1 に戻し、≪復帰フラグは 1 か？≫で YES にすすみ、復帰フラグを 0 にします。その後、アイドリングモーションが再生されサーボが ON になります。（“変数 251 が 0”の条件で再生されたモーションの中にあるポーズブロックは無効になるため≪復帰≫は再生されません。脱力を「動作書き換え」モーションで行わず「脱力」として独立させているのもこれが理由です。）

４．おわりに

VS-RC003HV で KO サーボを脱力するこの方法ですが、現在も試用中のため、今後新たな課題が浮上する可能性があります。また、本資料では、VS-RC003HV と近藤科学製サーボモータのみを使用しての脱力方法のみを記載となります。外部入力などを扱った場合の動作は保証できません。

もし、異常・不明な点・間違っていると思われる点を発見されましたら、お手数ですが以下の連絡先までご連絡くださいますようよろしくお願いします。本資料に関する情報を頂けましたら幸いです。よろしくお願いします。

2017 年 10 月 11 日
大阪電気通信大学
自由工房　ヒト型ロボットプロジェクト
関　悠伍
Mail：ee15a039@oecu.jp

Release 3.10　Online

5．参考文献

・RobovieMaker2 説明書
http://www.vstone.co.jp/products/vs_rc003hv/download/RobovieMaker2_docs.zip
・VS-RC003HV 取扱説明書
http://www.vstone.co.jp/products/vs_rc003hv/download/VS-RC003_manual.pdf
・VS-RC003HV 変数表
http://www.vstone.co.jp/products/vs_rc003hv/download/VS-RC003_valuelist.pdf
・VS-RC003HV シリアル通信資料
https://www.vstone.co.jp/products/vs_rc003hv/download/VSRC003_serial_manual.pdf
・ICS3.5　コマンドリファレンス
http://kondo-robot.com/w/wp-content/uploads/ICS3.5CommandReference1.pdf

6．更新履歴

Release 3.10	誤字脱字の修正、わかりにくい内容を補足
Release 3.00	START+SELECT で脱力できるように改定 大幅な資料の更新
Release 2.50	ポーズスライダの範囲が限定されている場合についての補足を追加 誤動作回避モーションに関する情報を追加
Release 2.40	ICS3.6 規格に関する内容を追加
Release 2.30	3-4 脱力 POSE の後にチェックボックスをはずす理由の内容を改
Release 2.20	定誤動作回避モーションに関する項目を追加
Release 2.10	START＋SELECT 入力に関する項目を追加
Release 2.00	大幅な資料の更新
(Release 1.21)	誤字脱字等の修正
(Release 1.20)	いままでモーションを使用可能にする項目を追加
(Release 1.10)	CPU の設定の項目を更新
Release 1.00	2015/08/13 初回限定公開版