MotionProcessor 実践チュートリアル



無線コントロールユニットとの接続

モーションプロセッサーは多くの機能を持っていますが、本来のパフォーマンスを発揮するためには、スクリプトを使用する必要がでて きます。しかし、ある程度プログラミングの経験がないと、いきなり記述して動作させるのは難しいと思います。 特に質問が多かった、無線コントロールユニットとの組み合わせで使用する場合の操作および設定方法を順を追って説明します。

実際に無線コントロールシステムを使用する前に、単純にサーボをモーションプロセッサーを PC 上のソフトウエアから動作させる手順 を確認しておきましょう。

1 ハードウエアの接続

まず、ハードウエアの接続を確認してサーボを動作させます。

モーションプロセッサーへのサーボの接続端子は、32 個あります。基板の周囲を取り囲む様に並べて配置してあるのが各サー ボ用の出力端子です。この端子は、ソフトウエア上の設定で、通常の信号(PWM)のほかに,Hlgh や Low のほかにサーボの設 定切り替えの信号の送出も可能です。通常のサーボコントロール時には、ソフトウエア側からの指定は "PWM" にしておきます。

※ PWM では、パルス信号(例えば 1.5msec)を繰り返して(10~20msec 間隔)送ることで、サーボの出力軸の位置を指定します。これは一般的なラジコン用サーボに使用されているコントロール方法です。では、High や LOW は、どのような場合にしようするかというと、信号線(白線)と黒線の間に抵抗と LED を付けて点灯の制御などが可能です。ただし、これは使用している CPU の定格内の電流にする必要があります。この電流の制限を気にしないで使用するためには、外部に回路を組んでこの制御に出力端子を使用するようにします。

簡単な動作確認をおこないますから、多くのサーボを接続するのではなく、まず CH1 に 1 個のサーボを接続してください。接続する際には、コネクターの向きに注意します。コネクターは、基本的に基板の外側に黒線が来る向きが正解です。

これからパソコンと接続して操作を行うために基板を次のような状態にしてください。

- 1、基板上のディップスイッチは3番のみを ON にします。
- 2、下の写真のようにサーボ、電源、通信用のインターフェースケーブルを接続します。



電源スイッチを入れると、基板上の緑のランプが点滅を始めます。点滅をしているのが正常な状態です。

次にソフトウエアを立ち上げて、実際にサーボの動作を行います。

2 セットアップを行う。

ソフトウエア HeartToHeart2 のインストールの方法や起動方法については、マニュアル本編をごらんになって行ってください。なお、マニュアルでは関連付けを行うように記述してあると思いますが、ご使用の環境によりうまく関連付けが出来ない場合には、4DRuntime.exe をダブルクリックして起動して、ここから、ファイルを開くことで起動してください。

HeartToHeart2 を起動すると右のようなメニュー画面が表示されます。 最初に「Set Up」を開いてください。

Compo	rt 1	•		Se	etupClose
H. 01 111		Light Grey 💌	GH. 17 SV. 17		Light Grey 💌
H. 02 SV. 02		Light Grey 💌	GH. 18 SV. 18		Light Grey 💌
H. 03 SV. 03		Light Grey 💌	GH. 19 SV. 19		Light Grey 💌
H. 04 SV. 04		Light Grey 💌	GH. 20 SV. 20		📃 Light Grey 💌
H. 05 SV. 05		Light Grey 💌	GH. 21 SV. 21		📃 Light Grey 💌
H. 05 SV. 08		Light Grey 💌	GH. 22 SV. 22		Light Grey 💌
H. 07 SV. 07		Light Grey 💌	GH. 23 SV. 23		📃 Light Grey 💌
H. 08 SV. 08		Light Grey 💌	GH. 24 SV. 24		📃 Light Grey 💌
H. 09 SV. 09		Light Grey 💌	GH. 25 SV. 25		📃 Light Grey 💌
H. 10 SY. 10		Light Grey 💌	GH. 26 SV. 28		📃 Light Grey 💌
H. 11 SV. 11		Light Grey 💌	GH. 27 SV. 27		📃 Light Grey 💌
H. 12 SV. 12		Light Grey 💌	GH. 28 SV. 28		📃 Light Grey 💌
H. 13 SV. 13		Light Grey 💌	GH. 29 SV. 29		📃 Light Grey 💌
H. 14 SV. 14		📃 Light Grey 💻	GH. 30 SV. 30		📃 Light Grey 💌
H. 15 SV. 15		📃 Light Grey 👱	GH. 31 SV. 31		📃 Light Grey 💌
H. 16 SV. 16		📃 Light Grey 👱	GH. 32 SV. 32		📃 Light Grey 💌
		Batt. mon. OFF	▼ Sur v. vo It	5.0V T	imelimit 5SEC
19200	BPS	HOS	T Control Soft	tware Upo	late



セットアップでは、3つのことを設定します。

ComPort で、使用するパソコンの通信ポートの番号を指定します。選択可能な番号は、1 又は2 ですが、キーボードから入力 することでこれ以外の番号を指定することが可能です。USB シリアル変換などをご使用の場合で番号が大きくなっている場合 等には直接番号を入力します。

※ USB シリアル変換をご使用の場合、シリアルポートの番号を確認するためには、ウインドウズのデバイスマネージャーを開く必要があります。WindowsXp の場合には、マイコンピュータを右クリック >> プロパティ >> ハードウエアから開くことが出来ます。

バッテリーモニターの設定は、最初は OFF にしておきます。 これは、電源電圧が一定以下になった場合に、動作をストッ プする機能ですが、最初は使用しません。

一番下の「HOST Control Software Update」は最初に一度
 だけ指定してください。

これをクリックするとファイル選択のダイアログが開きま すから、モーションプロセッサーの CD-ROM からハードディ スクにコピーしたファイルのうち、「SCC.MOT」を指定し ます。これを最初に一度指定することにより、通信がうま く出来なくなるトラブルを回避することが出来るようです。

鳥 デバイス マネージャ	
ファイル(E) 操作(A) 表示(V) ヘルプ(H)	
□ ■ KATSU-DESKTOP □ > DVD/CO-ROM ドライブ □ DVD/CO-ROM ドライブ □ DE ATA/ATAPI ユントローラ □ ● SOSI & RAID コントローラ □ ● SOSI & RAID コントローラ □ ● SOSI & RAID コントローラ □ ● マンドコータ □ ● サウンド、ビデオ、およびゲーム コントローラ □ ● サウンドイ、ビデオ、およびゲーム コントローラ □ ● サウンド、ビデオ、およびゲーム コントローラ □ ● サウンド □ ● ディスク ドライブ □ ● ティン パンタ・フェイス デバイス □ ● サリント □ ● アリンク ボート (UPTI) □ ● ● □ マウンとそのほかのポインティング デバイス □ ● モンタ	

3 サーボを動かしてみる

サーボを動作させるためには、モーションプロセッサーとパソコンのソフトウエア間の通信を行わせる必要があります。このためのスイッチがポジションエディターの BoardLink です。

E POSITION EDITOR			
RECHO 2 HOME POS ホームボジショ	PTSET01	🗖 Realtime Link	BOARD Link
GainSetSV GAIN SET 01 🚽 GainSetLD	Read BOARD DATA		45
TrackingCLR -5 -2 -1 +1 +2 +5	Return to Home	T.trin	CLOSE
TrackingSV DATAO1 TrackingLD	P to H	Teaching	
SV. 06 PHN -			

BoardLink をクリックしてチェックを入れると、ポジションエディター上の、 各サーボに対応するスライドバーを動かすことで、サーボを動作させることが できます。

POSITION EDITOR 2 HOME POS ホームボジショ RECHO GainSetSV GAIN SET 01 -GainSetLD TrackingCLR -5 -2 -1 +1 +2 +5 DATA01 Track ingLD **TrackingSV** -SV.06 PHN SV.01 PHN ▼ 109 129 0 • 1 IN PHN SV. 02 O 21 -

クリックしてチェックを入れても、一定時間で、チェックが外れてしまう場合。

これは、通信が確立しないことを表します。前述のSetUpでの通信ポートの番号が違うと当然ながら通信できません。また、ケーブルの接続位置やコネクターの位置が正しくない場合なども同様です。

以上の接続が正しい場合でも、うまくいかない場合の操作手順を下記に示します。

ソフトウエアの書込みに不備がある。

モーションプロセッサーでは、デュアルプロセッサーシステムを採用しており、1 番目の CPU で各サーボ制御や、入力端子からの読み込みなどを行います。2 番目の CPU では、通信に関する部分と、動作シナリオデータの保持とこれに基づく動作生成を行います。

モーションプロセッサーでシナリオを書き込む作業は、基本となる通信関連の部分と、シナリオデータを一つのファイルにして、 CPUのプログラムとして書込むことになります。初期状態のモーションプロセッサーには、この基本となる通信関連部分のファ イルが書き込まれてるはずですが、操作中にこのファイルが消えてしまうと通信ができなくなります。この場合、付属の SCC. MOT ファイルを書き込むことで通信が出来るようになります。

SCC.MOT を再度書き込む場合には、HeartToHeart2 を終了してから、外部プログラムを使用します。

SCC.MOT ファイルの再書込み手順



2 書き込み用のアプリケーションを起動します。

アプリケーションは、ハードディスクにコピーしたファイルの中の Flash フォルダの中にあります。 ダブルクリックして起動してください。

4DRuntime FlashSta.exe

HtoH2

3 最初の画面で、使用するシリアルポートを選択して OK を押します。



参照して指定するためには、Refer のボタンを押します。 ファイルを開く ? × Refer... OK Cancel ファイルの場所①: 👻 🕂 🖻 🗕 C HtoH2 CONTROL.mot scc96.mot 指定するファイルは、H2H2フォルダの中の「SCC. R MOT」です。指定すると ID が無い警告が表示されま すが OK を押して閉じてく M16C Flash Start ださい。 開(⊙) ファイル名(N) scc.mot Can not found the ID file. ファイルの種類(工): キャンセル Motlora Hex File (*.s:*.mot:*.s2) -ID Check ÖK FilePath C:¥MyRobot¥HtoH2¥scc.mot 00 00 00 00 00 00 00 ID ファイル名を指定したら、次に ID は全て 00 を記入し ます。 MCII Type MCU TYPE は M16C/20 62 のままとします。 M16C/20 62 ○ M16C/80 M32C ○ 38000
 M16C/80 M32C ○ 380000
 M16C/80 M32C ○ 38000
 C 880 OK Cancel Refer...

SCC.MOT ファイルの再書込み手順

▲ 実際に書き換えのウインドウが表示されます。

いくつものボタンがありますが、実際に使用するのは2つ

だけです。Settinngのボ タンを押すと書込みに使 用する通信ポートのス ピードを設定することが 可能です。手元の環境で は、57600が使用可能で したが、場合によっては もっと遅いスピードでな ※数字が小さいほど書込みのスピードは下がります。



5 書込みを行うために「E.P.R」のボタンを押します。



🍓 M16C Flash Start Load (ID)... Program... Blank... Erase Setting. Read. Status... Download... E. P. R.. Version... B. P. R., VDC status (M32C/83) ON VDC_OFF Exit

E.P.R は、Erase(消去) Program(書込み) Read(読み込み確認)を連続 して行うためのボタンです。最初の Erase では OK のボタンを押してくだ さい。あとは、最後に OK のボタンを押すと終了です。

※ここで書込みができないダイアログボックスなどが表示された場合には、 前述の Settinng で通信スピードを遅くして再度行います。

M16C F	lash Start	×
⚠	Can not set new baudr Baudrate is last baudra	ate. ate now.
	CCC OK	

4 サーボを動かす L3Program を作成する。

本体のマニュアルの場合ですと、モーションや L2PROGRM などで説明しますが、最終的に動作させて性能を引き出すためには、 やはりL3PROGRAM を作成することが必要です。ここからの手順では、サーボを動作させる L3PROGRAM と受信機から信 号を受け取る L3PROGRAM を作成し、最終的にこれを組み合わせることで、無線コントロールユニットを使用する場合の簡単 な動作例を作成します。

L3 プログラムを作成して動作テストを行う場合には、ポジションエディターで、ボードリンクがチェックされていないと実際 に動作を確認することが出来ませんから、最初の手順と同じくボードリンクにはチェックを入れたままにしてポジションエディ ターのウインドウは開いておきます。

次にメニューから L3PROGRAM をクリックします。



サンプルプログラム 「for Test」

:Start1 V01 = 0V02 = 0:Start2 S01 = 70 V01 = V01 + 1 Jumpif(V01,<,250,Start2) V02 = V02 + 1V01 = 0jumpif(V02,<,10,Start2) V01 = 0:Move1 V03 = 0 :Move2 S01 = 0V03 = V03 + 1 Jumpif(V03,<,250,Move2) V01 = V01 + 1 Jumpif(V01,<,10,Move1) Jump(Start1) (N255 = 0)exit

同じ動作をさせるためには、もっと簡単な書き方も可能ですが、プログラムとしての要素を 出来る限り取り入れた内容としました。 いくつかの行頭の: (コロン) があるのはラベルと呼ぶプログラムの目印のような働きをしま す。その下の V01 と V02 は変数です。このプログラムでは、この変数をカウントすることで 一定の時間間隔を作っています。S01 はサーボ変数と呼ばれるもので、この変数に数字を代 入することで対応する番号のサーボを動作させることができます。jumpif は条件分岐の命令 です。カッコ内の比較元と比較演算子、比較先を評価して、それが真の時に指定されたラベ ル先にプログラムの実行が移ります。 最初の Jumpif では、V01 変数が 250 より小さいときに Start2 のラベルに戻ります。 このプログラム前半では、CH01 のサーボを 70 に位置に移動しています。これ以外の部分は、 全てこの 70 にした位置を保持するためのカウンタです。 後半では、サーボの位置を0にして同じくカウンタで保持しています。

このサンプルは、CH01に接続したサーボを一定の角度に連続して動かすプログラムです。

最後の3行ですが、実はこの状態では、jump(Start1)で最初に戻りますから、最後の2行 は意味を持ちません。これはあとで、少し手を加えて使用するためのものです。

 mpif(V03,<,250,Move2)</td>
 このプログラムは、一定角度の往復を永遠に繰り返すことになります。しかし、無線制御で

 1 = V01 + 1
 繰り返すだけでは面白くありませんので、この動作を1回のみにしておきましょう。

 mpif(V01,<,10,Move1)</td>
 左のリストの最後から3行目 Jump(Start1)がループするジャンプ命令です。この行頭に:(セミコロン)をつけることで無効にします。

 255 = 0
 255 = 0

このプログラムを入力したら、save します。

プログラムの入力は、この通りに行ってください。入力に間違いがあると正常に動作しません。 入力規則として、代入(V01 = 0 など)や計算式(V02 + 1)の場合には、それぞれの要素間に半角スペースが必要です。また、 同然ながら全て、半角英数字です。(全角で入力しても、プログラムとして認識されません。)

送信機側の設定

S4

0

今回の例では無線コントロールユニットを使用することにしていますから送信機側のソフトウエアの設定が必要になります。 送信機側は、パソコンのソフトウエア「RCB コマンダー」を使用します。

送信機とパソコンとの間も RS-232C で接続しますが、モーションプロセッサーで使用した通信ケーブルでは通信できません。 必ず、無線コントロールユニット付属のものを使用してください。



S3

S2

低速度通信を出力する(⊻) 通信方法の設定(W)

ボタンのカスタマイズ 😒 コントローラーのプロパティ(Y) 現在のDesktopを保存する(<u>D</u>) Desktopのファイルを開く(E)

🗊 RC Bcommander

(

S1

左の RCB コマンダーの画面は、ボタンの配置や名称を、仮想コ ントローラとしての位置や表示に合わせています。 ここでの設定例は、左の上矢印キーを例にしていますが、実際 に割り当てるキーはどのキーを同じ設定にしても、同じ操作を 行うことが出来ます。

ボタンの機能設定のうち今回の例では、名前や名称は余り重要で はありません。ボタンのカスタマイズを選択して開く、「設定の 割り当て | が重要です。



今回の例では、どのキーでもかまいませんから、上矢印の設定 (左の写真) に割り当てたボタンを用意します。操作のテストを 行う場合に、このキーを使用します。

設定終わったら、右クリックで「低速度通信を出力する」にチェッ クを入れて送信を開始します。

※低速度通信を出力するにチェックが入った状態では、RCBコ マンダーを終了できません。終了する場合にはチェックを外し てください。

5 受信機データの取り込み

右の写真は受信機とサーボ、電源を接続した図です。これでは、まだ受信機用の電源スイッチが OFF になっていますから、使用する際には忘れずに電源を ON にします。



データチェック

受信機などを接続し、電源スイッチを入れたら、送信 機側の電波を発射します。

VARIABLE LINK のボタンを押すたびにデータの更新が 行われます。データのうち V12 と V13 が受信機からの データです。送受信機の接続が正常で動作していれば、 V12=240、V13=0 のデータが表示されます。

先ほど RCB コマンダー側で設定したキーを押した状態で、データの更新(VARIABLE LINK を押す)を行うと データは V12=240 V13 = 1 に変わります。

このチェックが正常に行えれば、これは送信機からの 信号が正常に受信できており、モーションプロセッサー 側にこれが入力されていることが確認できたことにな ります。

もし、チェックの結果が異なるようでしたら、接続設 定を再度確認する必要があるでしょう。

DAT	A CHEC	K									
	7 6	5	4 3	2	1 0						
)-INPUT							.INK		CI	ose	
I-INPUT							.INK				
P-INPUT		ᅳᅋᅳ		ᆜ				_		_	_
SERYO	01 0	02	0 03		04	0 05	0 06		07	0 08	
SERVO	09 0		0 11			0 13			15	0 16	
ERYU	25 0	18	0 19			0 21		H	23	0 24	H۳.
Ch ŦŪ	23 0	20	0 27	<u> </u>	20	0 23	0.50		31	FRVNII	
MR 00	0 01	0	02	len 0	0	04	0 05	0	1 30	0 07	0
AR. 08	0 09	0	10	0 11	0	12	240	0		0 15	0
AR. 16	0 17	0	18	0 19	0	20	0 21	0	22	0 23	0
AR. 24	0 25	0			N				ARRAY	(N255	0
NUN	TYPE	1	DATA		- 2	ī.			WART	ABLE L	I WIK
1	ह र आ के 25 हत				0	ARR	AY LIN	K			
2	87 31 30 35 81		-		0	Gy	roO	▼ S1	/01		-
- 3	配列数数67				Ū.	0			-	GyroS	et
4	配列建数的				0	64	ra Set		0 [11 [2
5	配列安数[4				0		10 021	vP ,	- ,		-
8	配列安数65				0					_	<u>^</u>
7	配列安数的				0						
8	配列安数印				0						
9	配列安数68				0						
10	配列変数(9)				0						
11	配列変数[1	0			0						
12	配列变数61	1			0						
13	配列安数61	2			0						
14	配列变数61	3			0						
15	配列安数[1	4			0						
10	83 31 xb 35 61	6			0						

7 受信機データ処理プログラムの作成。

;V12 FIRST BYTE V13 SECOND BYTE	このプログラムは、本編のマニュアルにあるものと同じです。
;INITIALYZEV12 = 0V13 = 0;	┃ V13 が 0 ですと、ループします。JUMPIF がその条件分岐です。
;TAIKI	データが変更されたら、V12のデータは V1 に代入。
. ,	V13 のデータは戻り値変数である(N255 に代入されます。
:MAIN	ここで exit でプログラムが終了します。
JUMPIF(V13,=,0,MAIN)	
V11 = V12	ここでは、判定させるためのデータに V13 しか使用していませんが、最後の 4 行を
(N255 = V13	(N255 = V12 * 256
exit	(N255 = (N255 + V13))
	JUMPIF ((N255,=,61440,MAIN)
	exit
	この様にすれば、1回の判定でシフトキーまでを含めたデータを判定することが可能です

このプログラムも、先ほどのサーボ動作のプログラムと同じように、Radio Control Test などといった名称を付けてセーブしてください。

8 シナリオの作成。



Sce	n ar io	Editor												×
L1 Ma	tion Li	st L2 Program	List L3	Program List	t		Registered	Scenario li	st		_	,Clo:	se	[-
[<u>.</u>	Registered L	3Program lis	t		RECNO	Separ in 3	COMMENT	STEPCO	NTROL		4		1-
	REGNUN	L3PGNANE	DATE	TINE	•					ľ		nome positi	ion serop	
	2	for Test	05/06/15	15:29:50								Scnario ez	secut ion	
	1	Radio Control lest	05/04/04	15:48:04		-							STOP	-
												New Scr	nar io	
												Senar io	Setup	
												Сору Sc	nar io	1
						-					-	Delete S	cnar io	1
						L3	orogram setup >>	Aut	o Setup					
							Registered Sce	nario item l	ist			<< SET Senar	io	1
						Add I	TEM Ins.ITEM	Delete ITEM	DATA Setup		Bran	ch condition	List	
						ProcID	PGNAME		timeout 🔺	044	1	L3 Radio Co	EN L	1
							1 L3 Radio Contro	1	0	LINE		L.Lin. H.Lin.	Br.ID A	d I
				1			2 L3 for Test		0					
											-			
											-			L
											-			L
							-							L
											+			L
														L
											-			
					•									Ι.

これで、L3 プログラムが登録できました。 次にこの分岐設定を行います。

F	legistered Sc	enario item I	list				<< \$E1	' Scnar	io	
Add ITE	Ins.ITEM	Delete ITEM	UP		Bra	nch co	ndition	n List		
ProcID PGNAME timeout Add ITEM Delete ITE									TEN	
1	L3 Radio Contro	ol in the second se	0		LIN	E	L.Lim.	H.Lim.	Br.ID	
2	L3 for Test		0			1	1	1	2	
					P	2	0	0	1	
										-
	<u> </u>									-

L3 Radio Contorl の分岐設定は、2 個の設定 を追加します。

Add ITEM をクリックして項目を追加してから、その内容を変更します。

LINE 1は、その値が1の時に ID2、すなわち、 L3 for Test へ分岐します。LINE2 は、値が0 の場合には、もう一度自分自身へ戻ります。

この参照している値は、もう一度、Radio Control の L3 プログラムの内容を思い出し てください。プログラム無いで設定している (N255 と言う変数(戻り値変数)でこれは、 受信機からの受信した信号を意味していま す。

ID2 の L3 forTest では、常に戻り値が 0 にな ります。終了後の分岐は、再度 ID1 の Radio Control を実行するように設定しています。

Registered Scenario item list << SET Scnar io 3 Scnar io 3 Branch condition List Add ITEM Ins.ITEM Delete ITEM DATA Setup 2 L3 for Test ProcID PGNAME ٠ timeout Add TEM Delete ITEM 1 L3 Radio Control 0 LINE L.Lim H.Lim Br.ID 🔺 2 L3 for Test Û 1 0 0 1

シナリオを作成しましたら、このシナリオを Scenario excution を押すことでテストすることが出来ます。

9 シナリオのエクスポート。

📕 HeartToHeart2 Menu	×
Set Up	<u>*</u>
POSITION EDITOR	
MOTION	
LEVEL 2 PROGRAM	
LEVEL 3 PROGRAM	
DATA CHECK	
Key Control	
ScenarioEditor	
ScenarioExport	\triangleright
QUIT	
KONDO	Ver1.0
•	

シナリオが作成できたら、シナリオを外部ファイルへ書き出して、モーションプロセッサに書き 込むためのファイルを作成します。 この作業を行うには、ScenarioExportを使用します。

シナリオエクスポートは、書き込むシナリオとホームポジションを指定して Secenario data Output をクリックします。

現在作成した例では、ホームポジションを特に作成していませんので、こちらの指定は任意です。

İ Sc	enario Export							
	Registered	Scenario list		_	Reg	istered home posit	ion list	
RECNO	NAME	COMMENT	STEPCONTROL	▲	REGNO	HONENANE	SEL	
	Scnario 3		0	_	2	ホームポジション	V	
			0		1	初期状態	Г	
		4	0					
					I			
	<u></u>							
					<u> </u>			
						1		
				τl	Sce	nario data outp	ut) C	lose
Ľ				_				
								► E

エクスポートを行うと、H2H2 フォルダの中に「CONTROL.MOT」という名称のファイルが作成されます。 このファイルをボードに書き込むためには、前述の SCC.MOT ファイルの書込みと同じ手順で行います。 (書き込むファイルの指定の際に、SCC.MOT ではなく、CONTROL.MOT を指定します。)

10 実行

無事ファイルが書き込めたら、最終的なテストです。

モーションプロセッサーに電源、受信機、サーボを接続します。なお、スタンドアロン(単独)で実行させる際には、モーショ ンプロセッサーのディップスイッチをすべて OFF にする必要があります。



左の写真では高速シリアル端子にはパソ コンからのインターフェースが接続され ていますが、スタンドアロンで動作させ る際には当然必要ありません。

以上の接続及び設定に間違いが無ければ、送信機側の上矢印キーを操作すると、サーボが一定角度動作するはずです。

この設定は無線コントロールユニットの場合を説明しましたが、KRT-2を使用する場合もまったく同じです。 ただし、KRT-2のキーの配置が若干違う点があること、送信機側では特に設定を行なう必要が無い点が異なります。

ここで取り上げた動作設定は非常に簡単な例でしたが、例の中でサーボが動く設定の部分を、前進動作のL3Program に置き 換えれば、すぐに前進の動作が可能です。

たとえば、KHR-1 をご使用の場合でしたら、RCB-1 で公開されているサンプルモーションをインポートして、最終的に L3program に持ってきます。

コンバートした L3Program の最後に、(N255=0 を加えるだけで済むはずです。

また、同様な手順で、L3program を追加して、シナリオエディターの分岐先を増やせば、通常必要な動作についてすぐに作成できるでしょう。