

「スカイベリーカーゴ・トレーサー」ベースキット

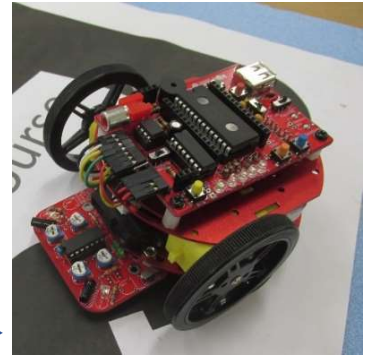


SkyBerryCarGO-TR 組立説明書

ライトレース・障害物回避・迷路走行プログラミング

- 小型マイコン(別途SkyBerryJAM®やArduinoなど)と接続して、センサ型プログラミングロボットカーとなるベースキット●ホイール(車輪)型
- センサボード(反射型赤外線センサx6)付属

【内容】ホイール(車輪)型ロボットベース一式・スカイベリーセンサボードキット一式・単4x3電池ボックス・スナップソケット・取り付けビス(別途:小型マイコン・単4アルカリ電池x3)

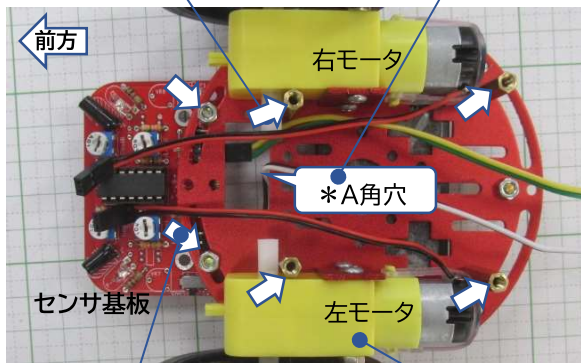


*以下 番号順 ①~⑬に組み立ててください

組立完成例▶

●1 下部ベース板上面

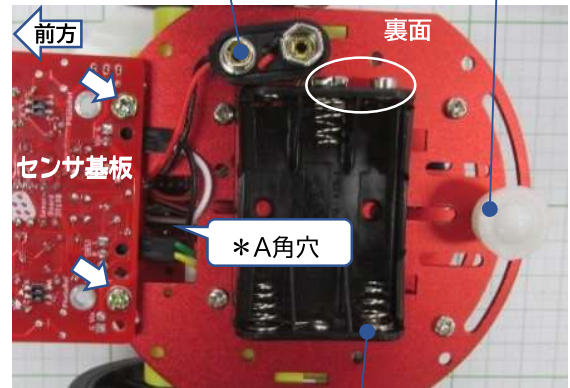
- ①スパーサ固定x4
- *A角穴から配線コード上へ引き出し



- ②センサ基板 ネジ固定x2 付属スパーサを間にいれる
- ③モータ固定 配線は前方へ

●2 下部ベース板 裏面

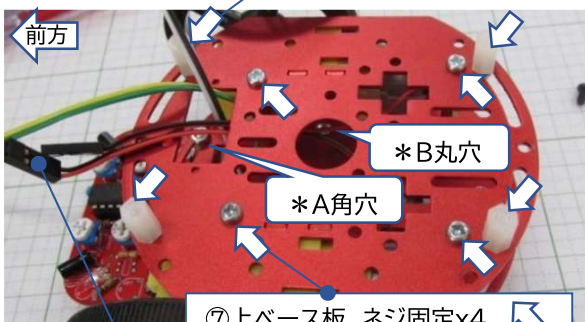
- *電源スナップソケット ●4で*A角穴から引出し
- ⑤キャスト固定



- ④電池ボックス 両面テープではりつけ 端子は上向きに

●3 上部ベース板

- ⑥スパーサ(スカイベリージャム付属) 4隅に取付けx4 (*2020バージョン以前の場合はかなりゆるめに取付)

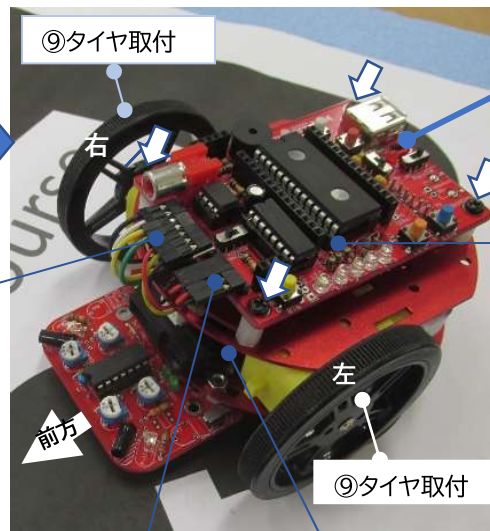


- ⑧各種配線材 *A穴から上に引き出し

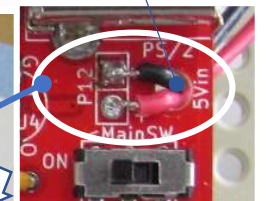
- ⑩センサー基板配線
G → G
Vin → 5V
右R → IN1
左L → IN4
中央C → A2

●4 その他の取り付け・配線(スカイベリージャム取付の場合)

- *スナップソケットとスカイベリージャムとの配線 穴を通してP12 赤+黒G はんだ接続 電池用スナップは、使わないときは、必ず電池ボックスから外しておく



- ⑨タイヤ取付
- ⑫モータ配線 *極性はプロ グラムで調整 右→M1 左→M2



- ⑩SkyBerryJAM本体 ネジ固定x4
- *電池用スナップソケットは上部ベースの穴*B丸穴を通して、下部ベースの*A角穴から下へ引き出す(●2参照)

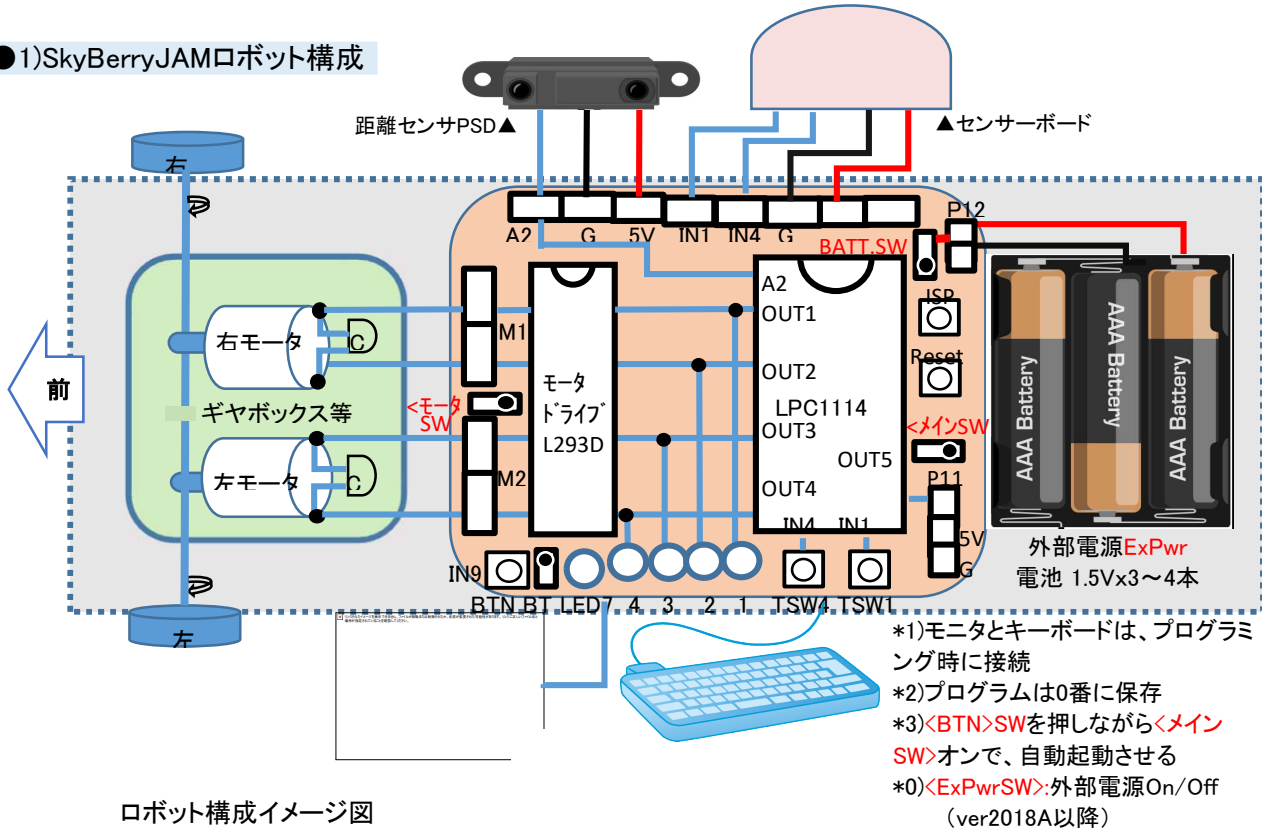
- ⑬PSDセンサがある場合 取付・配線 赤色 →5V 黄色 →A2 黒色 →G *接続順はバージョンにより異なる

●企画・設計: 栃木県立栃木工業高等学校
skyberryjam@tochigi-edu.ed.jp
●生産・販売: VKステック 宇都宮市鶴田町
vks-tec@trade.gmob.jp
榎木総業(株) 東京都足立区加平3-18-8
ksh@kashinoki.co.jp

SkyBerryJAMロボットプログラミング編

SkyBerryJAMは、直流モータドライブIC(L293D)を標準で搭載していますので、組み込みロボットプログラミングが簡単に実現できます

●1)SkyBerryJAMロボット構成



ロボット構成イメージ図

- ①各直流モータの端子には、雑音防止のために、セラミックコンデンサC(0.01~0.1uF)を接続します。
- ②<M1>ピン端子には右モータ、<M2>には左モータを接続します。(各モータの2本の配線向きは、後で調整します。)
- ③外部電源として、単3アルカリ電池や充電電池等3~4本を<P12>に接続します。<ExPwrSW>でその外部電源をOn/Offします
- ④距離センサ(アナログ型)は、A2(アナログ入力2)に接続します。他のセンサはIN1,4などに接続します。
- ⑤プログラムは0番に保存、(モニタ・キーボードを外し)<BTN>を押しながら<メインSW>オンして自動起動させます。(または、BTブート端子をオンしておく、<メインSW>オンで常に0番が自動起動します。)

●2)モータ動作確認と調整 (JAM本体にはモニタとキーボード接続、MainSWとモータSWをOnにしておく)

OUTポート	4	3	2	1
データ例	1	0	0	1
LED	●	○	○	●

例) **OUT 9** または **OUT `1001**
 OUTポートに9(2進値1001)を出力するとLEDは、●○○●と点灯されモータドライブIC-L293Dにより左モータ前進・右モータ後退=全体右回転

動作データ	モータ動作
00	停止(フリー)
10	回転(前進)
01	反転(後退)
11	ブレーキ

L293D ↑前進 後退 ↓前進 後退
 左モータ 右モータ

ロボット本体は、台などをおいて浮かした状態にして、次のコマンドを入力して、モータ回転とLED表示を確認します。

	<左モータ>	<右モータ>	LED表示	動作データ	動作データ	モータ動作
			4 3 2 1	2進値	10進値	
① OUT 1<enter>	-	回る	○○○●	0001	1	右後退
② OUT 2<enter>	-	逆に回る *前進向きに調整	○○●○	0010	2	右前進
③ OUT 0<enter>	-	停止	○○○○	0000	0	停止
④ OUT 4<enter>	回る	-	○●○○	0100	4	左後退
⑤ OUT 8<enter>	逆に回る	- *前進向きに調整	●○○○	1000	8	左前進
⑥ OUT 0<enter>	停止	停止	○○○○	0000	0	停止

*前進する向きになるように調整(<M1><M2>それぞれで、2本の配線を入替え)し、上の表の動作となるようにします。

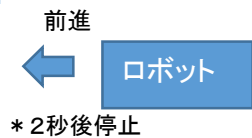
●3)ロボット基本動作コマンドの確認

次のコマンドを入力して、モータ回転とLED表示を確認します。

	<左モータ>	<右モータ>	<全体>	LED表示	動作データ	動作データ	ロボット動作
				4 3 2 1	2進値	10進値	
① OUT 8+2<enter> =OUT 10<enter>	↑	↑	=前進	●○●○	1010	10	前進
② OUT 4+1<enter> =OUT 5<enter>	↓	↓	=後退	○●○●	0101	5	後退
③ OUT 8+1<enter> =OUT 9<enter>	↑	↓	=右回転	●○○●	1001	9	右回転
④ OUT 4+2<enter> =OUT 6<enter>	↓	↑	=左回転	○●●○	0110	6	左回転
⑤ OUT 0<enter>	-	-	=停止	○○○○	0000	0	停止

●例1 前進(2秒間)プログラム

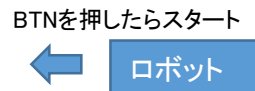
1' ZENSHIN	先頭にはプログラムの内容を ' で書いておくとよい	
10 WAIT 60	1秒程度待つ	LEDの表示
20 OUT 10:WAIT 120	前進 2秒待つ	●○○●○○ 1010
30 OUT 0	停止	○○○○ 0000
40 END	プログラム終了	
SAVE 0 <enter>	0番に保存	



モニター・キーボードを外し、<BTN>を押しながら<メインSW>オンして自動起動させます。
 ロボット実行動作後は、<メインSW>OFF、モニター・キーボードをつなぎ、再び<メインSW>ONする
 LOAD 0<enter> 動作確認後、保存しておきたい場合は、0番を一度呼び出して
 SAVE 1 <enter> 他の番号、例えば1番に保存しておく

●例2 前進・後退・左右回転 連続動作プログラム

NEW		
1' RENZOKU		LEDの表示
10 WAIT 60	1秒程度待つ	○○○○ 0000
20 OUT 10:WAIT 120	前進2秒	●○○●○○ 1010
30 OUT 5:WAIT 120	後退2秒	○●●○○● 0101
40 OUT 9:WAIT 120	右回転2秒	●○○●○○ 1001
50 OUT 6:WAIT 120	左回転2秒	○○○○ 0000
60 OUT 0: GOTO 10	停止、10行目へ移動 無限繰り返し(ループ)	
SAVE 0 <enter>	0番に保存	



(以下同様に0番に保存し、動作後保存しておきたい場合は、他の番号で保存しておく)

●例3 BTN(一番左のボタン)を押したらスタートするように例2を改造する

LOAD 0<enter>	
15 IF BTN(=)0 THEN GOTO 15	BTN()は、BTNボタンを調べる命令で、押していない0の間は、
SAVE 0 <enter>	15行目(現在行)を無限に繰り返し、ボタンを押すまで待っている

(例題2に、上の1行を追加する)
 (以下同様に0番に保存し、動作後保存しておきたい場合は、他の番号で保存しておく)

●例4 <TSW1>を押したら右回転(3秒間)、<TSW4>を押したら左回転(3秒間)するプログラム

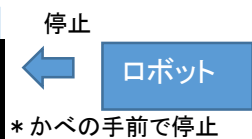
NEW	
1' BTN KAITEN	
10 IF IN(1)=0 THEN OUT 9:WAIT 180:OUT 0	IN(1)はTSW1を調べる命令で、押すと0となる
20 IF IN(4)=0 THEN OUT 6 :WAIT 180:OUT 0	IN(4)はTSW4を調べる命令で、押すと0となる
30 GOTO 10	10行目へ移動 無限繰り返し(ループ)
SAVE 0 <enter>	0番に保存

(以下同様に0番に保存し、動作後保存しておきたい場合は、他の番号で保存しておく)

●例5 前進して、前にかべがあれば停止するプログラム(距離センサ利用)

NEW	
1'ZENSHIN SENSOR	
10 WAIT 60	1秒程度待つ
20 D=ANA(2):PRINT D	ANA(2)は距離センサを調べる命令 0~1024でかべまでの距離
30 IF D>600 THEN OUT 0:END	壁が近い600以上なら停止 ENDプログラム終了
40 OUT 10:WAIT 5	そうでないなら 少し前進
50 GOTO 20	20行目へ移動 無限繰り返し(ループ)
SAVE 0 <enter>	0番に保存

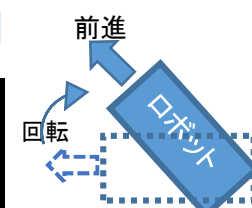
(20行目の PRINT D があるのでモニター画面をつないでおくと、センサの値が確認できる 600の値は調整する必要あり)
 (以下同様に0番に保存し、動作後保存しておきたい場合は、他の番号で保存しておく)



●例6 前進して、かべがあったら向きを変え、再び前進するプログラムに例5を改造する

LOAD 0<enter>	
30 IF D>600 THEN OUT 9:WAIT 120	壁が近い600以上なら右回転2秒間
SAVE 0 <enter>	(600や2秒間の値は、調整して変更する)

(例題5の30行目を上に変更する)
 (以下同様に0番に保存し、動作後保存しておきたい場合は、他の番号で保存しておく)



●番外 動作スピードを調整するには、PWM命令を使います

例) 前進 PWM 4,1000:PWM 2,1000	OUT4, 2 に10ms周期で信号出す
停止 PWM 4,0:PWM 2,0	(1000がスピード値となる 最大2000)