

A 日程作業報告書

1 作業の進捗

<http://www2.denshi.numazu-ct.ac.jp/mirsdoc2/mirsmg4d/>のドキュメントに沿って A 日程の作業を行った。

ドキュメントの進捗状況を簡単に表 1 にまとめる。作成中に起こった問題点などはあとで説明する。

表 1

ドキュメント番号	進捗
MIRSMG4D-SYST-0003	すべて完了。 http://www2.denshi.numazu-ct.ac.jp/mirsdoc2/mirsmg4d/syst/num0003a/index.html に従って arduino のセットアップを行った。
MIRSMG4D-SYST-0004	すべて完了。 http://www2.denshi.numazu-ct.ac.jp/mirsdoc2/mirsmg4d/syst/num0004a/index.html に従って Raspberry Pi のセットアップを行った。
MIRSMG4D-SYST-0005	すべて完了。 http://www2.denshi.numazu-ct.ac.jp/mirsdoc2/mirsmg4d/syst/num0005a/index.html に従い試験まで行った。
MIRSMG4D-SYST-0006	すべて完了。 http://www2.denshi.numazu-ct.ac.jp/mirsdoc2/mirsmg4d/syst/num0006a/index.html に従い超音波センサのアドレスを変更した。
MIRSMG4D-SYST-0007	3. まで完了。 http://www2.denshi.numazu-ct.ac.jp/mirsdoc2/mirsmg4d/syst/num0007a/index.html 「3.単体機能テスト」で正常に動作したので「4. 単体テストが正常に行えなかった場合の対応」は行っていない。
MIRSMG4D-SYST-0008	3. まで完了。 http://www2.denshi.numazu-ct.ac.jp/mirsdoc2/mirsmg4d/syst/num0008a/index.html

「4. Arduino と接続しての動作試験」は行っていないため、B 日程で行ってほしい。

2 メカニクス

2.1 部品チェック

2.1.1 部品チェック

MIRSMG4D-MECH-0001、MIRSMG4D-MECH-0002

を参考に標準機に使用する部品があるかチェックを行った。

結果を以下の表 2 に示す

表 2 標準機部品

表 2 部品表

チーム名 MIRSMG 2005 MIRSMG4標準機部品一覧(チームチェック用) 2020/07/29改定

種別	部品名	部品名	数量	確認状況	備考
ボード	Raspberry Pi	Raspberry Pi 4 / Model B	1	○	
Arduino	Arduino	Arduino UNO	1	○	
	モータードライブ	Cytron MD100	2	○	
	Arduino UNO用シールド基板	Arduino用ユニバーサル基板 ガラスコンポジット	1	○	有難に聞わず新たに作る
駆動系電機部品	ユニバーサル基板		1	○	有難に聞わず新たに作る
	近接式押しボタンスイッチ	UB-16HaSKP1R1	1	○	
	ターミナルブロック(青)	P-01306	2	○	
	ターミナルブロック(緑)	P-02333	2	○	
メカ部品		(部品コード)			
	上段シャーシ	MIRSMG4 UK	1	○	アクリル板
	中段シャーシ	MIRSMG4 MG	1	○	アクリル板
	下段シャーシ	MIRSMG4 LG	1	○	アルミ板
	短支柱	MIRSMG4 SP	4	○	アクリル棒
	長支柱	MIRSMG4 LP	4	○	アルミ棒
	タイヤホイール	MIRSMG4 TW	2	○	ABS樹脂
	モータマウント	MIRSMG4 MM	2	○	ABS樹脂
	モータマウントサポート	MIRSMG4 MS	2	○	ABS樹脂
	制御用バッテリーホルダ	MIRSMG4 OH	2	○	ABS樹脂
	駆動用バッテリーホルダ	MIRSMG4 DH	2	○	ABS樹脂
	超音波センサマウント	MIRSMG4 UM	1	○	ABS樹脂
	ハンパ	MIRSMG4 BP	1	○	ABS樹脂
	キャスター		2	○	
センサ・カメラ					
	超音波センサ	Devantech SRF02	2	○	
	ハワジング セン オス	モレックス 5045-04A	2	○	超音波センサにハンダ塗
	Webカメラ	ロジクール C270	1	○	
	タッチセンサ	SS-SGL	1	○	前適用
	2pinソケットタッチセンサ => FFP4		1	○	ケーブル先端に取り付け
ケーブル					
	HDMIケーブル (mini =>標準)		1	○	解体機にどのようなケーブル
	Raspberry Pi 電源ケーブル(Type C)		1	○	があったか、制御系用、駆動
	USBケーブル (Type A => B)	Raspberry Pi => Arduino	1	○	系用のもは確認して記入
	USBケーブル (Type A => C)	Raspberry Pi / バッテリー駆動用	1	○	しておくベターです
モータ					
	エンコーダ付きモータ	先帝技研 KSSN-IG36P-xxEN #1	2	○	*1 volはギア社に担当する数字が入る
その他					
	USBバッテリー	Sony CP-F10LA	1	○	
	機用バッテリー	NIMH 7 BV 3000mAh または 3000mAh	2	○	
	バッテリー充電器	TAMIYA AC-90M NIMH/Ni-Cd	1	○	機用バッテリー用
	無線キーボードマウス	Logitech ファイアレス MK245 Nano	1	○	
	ディスプレイ		1	○	

2.1.2 ネジのチェック

標準機で使用するネジ類についても MIRSMG4D-MECH-0003 を参考にチェックを行った。項目を以下の表 3 に示す。

表 2 のネジ類は全てであることが確認できた。

表 3 標準機に使用するネジ類

名称	用途	長さ[mm]	個数	SWの有無	ナットの有無
5mmキャップボルト	モータマウント固定	12	8	有	無
	短支柱固定	12	8	有	無
	長支柱固定	12	8	有	無
3mmねじ	ギアヘッド・モータマウント接続	15	8	有	無
	モータ軸Dカット固定用ネジ	10	2	無	無
	駆動用バッテリーホルダ固定	15	8	無	有
	制御用バッテリーホルダ固定	12	8	無	有
3mmねじ	超音波センサマウント固定	12	4	無	有
	Arduino固定	12	4	無	有
	モータドライバ固定	12	8	無	有
	電源ボード固定	12	4	無	有
2.6mmねじ	Raspberry Pi固定	25	4	無	有
	キャスター固定	10または12	6	有	有
スペーサ	Arduino固定	5	4		
	モータドライバ固定	5	8		
	電源ボード固定	10または15	4		
	Raspberry Pi固定	10	6		
	キャスター固定	3	6		

2.1.3 ネジの整理

作業がしやすいようにネジをまとめたその様子を図1に示す。

左、3mm以下の使用するネジ

右、5mmキャップボルト

真ん中、使わないネジ(余り)



図 1 ネジ

2.2 ダンパーの作成

2.2.1 一個目のダンパー作成

タッチセンサが直接壁などに当たることを防ぎ衝撃を吸収する、タッチセンサの範囲拡張などの理由から solidworks と 3D プリンタを使用してダンパーを作成した。作成したダンパーを図2に示す。

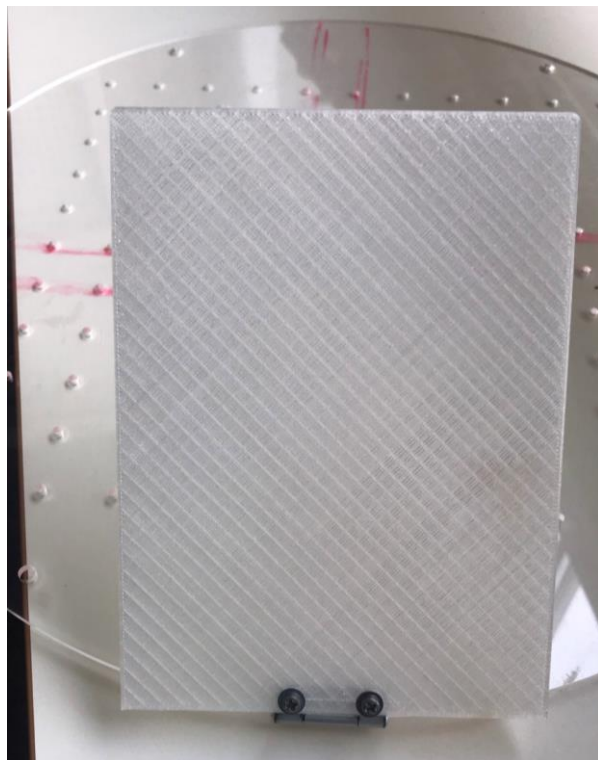


図 2 バンパー 1

2.2.2 ダンパーの取り付け

ダンパーを取り付けるために上段シャーシに穴あけを行った。M3ネジのサイズで穴あけを行った。その様子を図3に示す



図 3 ダンパー接続部

2.2.3 ダンパー2の作成

図2のダンパーだと標準機を組み立てた時超音波センサに干渉してしまうこ

とがわかったので干渉しないように新しくバンパーを作成した。

時間がなくプリントはできなかったので図は割愛

- 標準機の下段組み立て（途中）とモーターマウントの接続を行った。下段の組み立てたところを図4に示す。

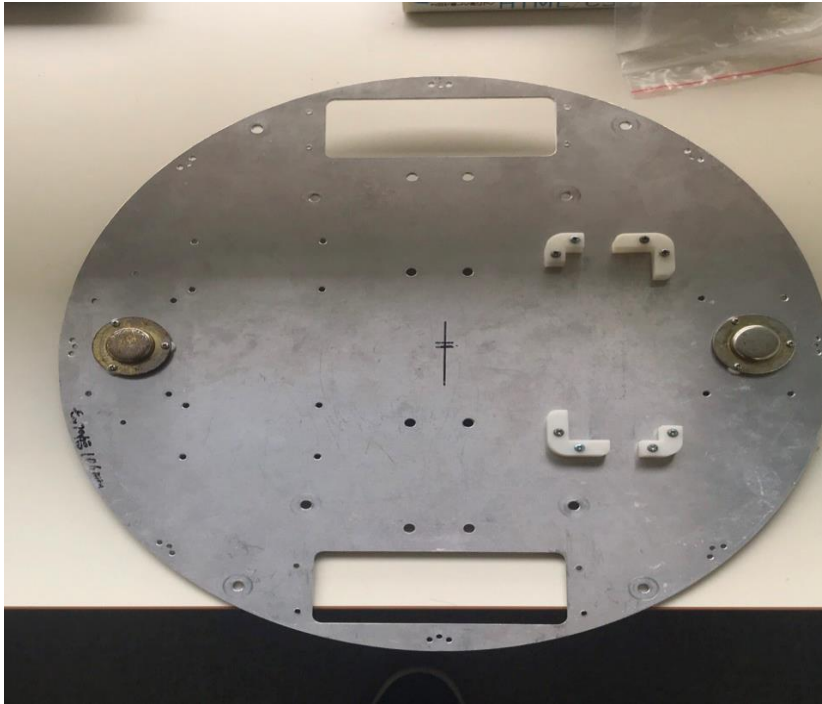


図 4 下段シャーシ途中経過

3 エレクトロニクス

3.1 引継ぎ基板の再利用チェック

引継ぎ基板が再利用可能かをチェックし図5にまとめた。

チェックの結果、駆動系電源ボードのパーツがドキュメントのものと違う、安定した動作ができない、という理由で駆動系電源ボードの新規作成(3.3)が決まった。

電子機械設計実習 機体ボード再利用チェックシート 00007/

チーム名: MEBS_2組 E 記入責任者: 木本 悠太郎

機体ボード名称	確認項目 (あり/なし)	確認結果 (○/×)	再利用判定 (○/×)	再利用理由 (○/×)
Arduino シールド基板	あり	○	○	
Raspberry Pi シールド基板	あり	○	○	
駆動系電源ボード	あり	○	×	

再利用機体部品チェック表

チェック欄には○、判定欄には○または×を入れること

Arduino シールド基板 チェック者: 木本 悠太郎 7月6日

部品配置	配線経路	非標準部品	電圧出力	判定	備考
チェック	チェック	チェック	チェック	○	
✓	✓	✓	✓	○	

Raspberry Pi シールド基板 チェック者: 木本 悠太郎 7月6日

部品配置	配線経路	非標準部品	電圧出力	判定	備考
チェック	チェック	チェック	チェック	○	
✓	✓	✓	✓	○	カメラピンは(1)ピン Q1は13ピン(1)ピン(1)ピン

駆動系電源ボード チェック者: 木本 悠太郎 7月6日

部品配置	配線経路	非標準部品	電圧出力	判定	備考
チェック	チェック	チェック	チェック	○	
✓	✓	✓	○	○	部品が異なる

図5 再利用チェックシート

3.2 Arduino シールド基板のチェック

昨年度の機体から引き継いだシールド基板のチェックを行い、今年度でも再利用できるかを確認した。その結果ドキュメントと配置と違うが標準機能の回路はそのまま残っていて使用できることが分かった。

端子の内容について以下に示す。

オス端子が出ている面から見て、その端子の上から順番につながっているピンを書いている

2ピン A5ピン、グラウンド

バッテリー接続用

3ピン上左 13ピン、5ボルト、グラウンド

用途不明だがセンサかと思われる

3ピン上右 10ピン、5ボルト、グラウンド

用途不明だがセンサかと思われる

3ピン中左 8ピン、9ピン、グラウンド

右輪モータ

3ピン中右 12ピン、11ピン、グラウンド

左輪モータ

3ピン下右 6ピン、5ピン、グラウンド

マーカ機構のモータとあったので標準機とは関係ないかと

4ピン左 5ボルト、グラウンド、7ピン、3ピン

右輪エンコーダ

4ピン右 5ボルト、グラウンド、4ピン、2ピン

左輪エンコーダ

とりあえず昨年（MIRS1903）のエレキ詳細設計書を見て分かったのは用途を書いています。

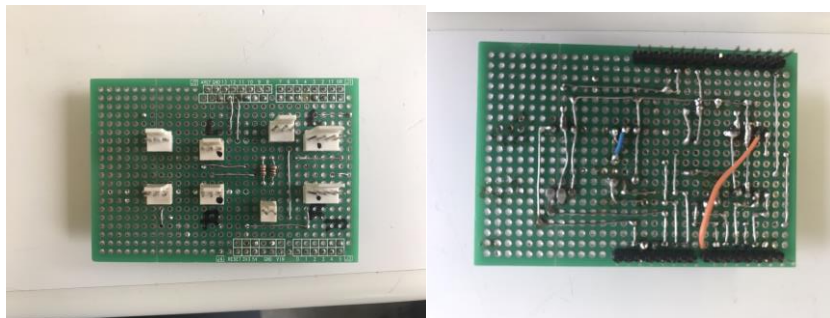


図 6 arduino シールド基板

3.3 Raspberry Pi シールド基板のチェック

昨年度の機体から引き継いだシールド基板のチェックを行い、今年度も再利用できるかを確認した。その結果ドキュメントと同様の部品配置で基本の回路が実

装されているのを確認した。

前回機体の増設により io14,io15,GND から 4 ピンのハウジングに接続されているが標準回路には関係ないためそのまま残している。

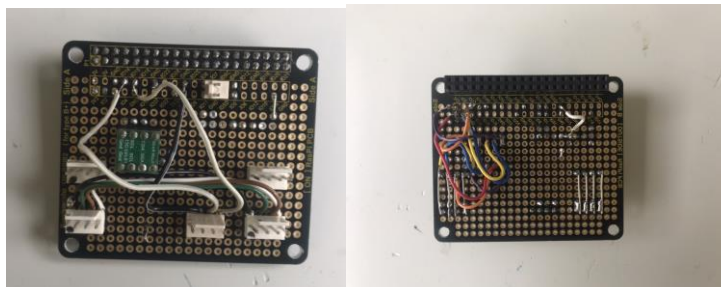


図 7 Raspberry Pi シールド基板

3.4 駆動用電源ボードの作成

3.4.1 実施内容

MIPS データベース（リンク先）を参考に、シールド基板の導通チェック、単体試験を行った。

3.4.2 結果

部品の接続が弱く、導通が不安定であった。更には MIPS データベースの回路とバッテリー入出力の端子が大きく異なっていた為に作り直すことにした。作り直した駆動用電源ボードを以下に示す。

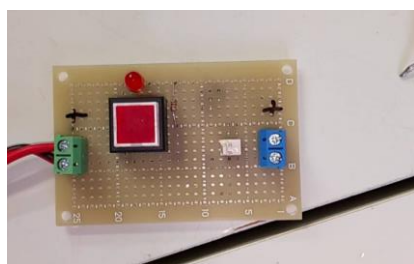


図 8 駆動用電源ボード表

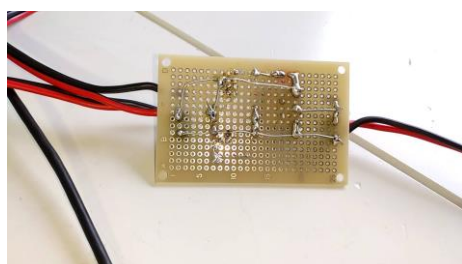


図 9 駆動用電源ボード裏

3.4.3 動作確認

新たに作った駆動用電源ボードの導通チェックを行い（スイッチを押したときと押ししていないときの差異も確かめた）、単体試験ではバッテリーに接続して、スイッチを押したときにLEDが光るか否かを確認した。

4 ソフトウェア

表1のMIRSMG4D-SYST-0007、MIRSMG4D-SYST-0008を行った

4.1 Arduino 単体動作試験

PC、arduino、arduino シールド基板、モーター、モータードライバ、駆動電源ボード、バッテリーを図10図11の回路図のように接続し、単体試験を行った。

回路図にはないが、駆動用電源ボードの2ピンとシールド基板を接続しないとバッテリー残量確認用のプログラムが正常に動作しない。

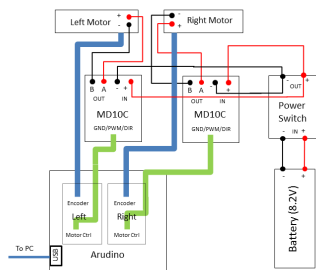


図 10 arduino 単体試験回路図

図 11 arduino 単体試験写真

4.1.1 Arduino 単体試験で起こった問題

- ・この試験を行う際に arduino シールド基板の前回機体で増設された3ピンのハウジングにモーターの回転制御を繋げてしまい、モーターがうまく動かなかった。
- ・arduino に書き込むテストプログラムの mg4_arduino_ver313 の23行目の slave(); をコメントアウトし忘れたため、うまく動作しなかった。
- ・左のモータドライバが壊れていたため試験がうまくいかなかった。別のものに交換すると問題なく動作したため、シールド基板や配線に問題はないと考えられる。

4.1.2 動作試験の結果

テストプログラムのテスト用関数を一つずつコメントを外し動作を確認しその結果を表3にまとめる。

表 3 テストプログラムの実行結果

実行した関数	結果
test_motor(x,y)	<p>x y に正の値を入力すると、各モーターが機体が前進する方向に回転し、負の値を入力すると後退する方向に回転した。</p> <p>入力された数値が大きいほど回転速度は速くなった。</p>
motor_set(50, 50); test_encoder();	<p>シリアルモニターに左右のエンコーダの値が表示された。</p> <p>(例)</p> <p>enc_l = 左モータのエンコーダ値 enc_r = 右モータのエンコーダ値</p> <p>通信速度を 115200bps に変更しないと正常に表示されない。</p>
motor_set (50, 50) ; test_distance ();	<p>走行距離が一定時間毎にシリアルモニターに表示された。</p> <p>(例)</p> <p>dist_l = 左モータの走行距離 dist_r = 右モータの走行距離</p>
test_vel_ctrl (25, 0);	<p>入力の大きさに比例してモータの回転速度が変わったが、実機での試験を行わないと実際に引数通りの速度で回転しているかはわからない。</p>
test_run_ctrl (STR, 25, 100)	<p>左右のモーターが直進の方向へ回転し、第二引数の値が大きくなれば回転速度も速くなった。第三引数の値を大きくすると回転している時間が多くなった。</p>
test_run_ctrl (ROT, 25, 100)	<p>左右のモータが機体が回転するように回転した。第二引数の値が大きくなるほどモータの回転速度は速くなった。</p> <p>第三引数の値が正の時は機体が反時計回りに回るように負の時は時計回りになるようにモータが回転する。</p>
test_batt ();	<p>シリアルモニタにバッテリーの電圧値が表示される。</p>

	(例) volt = バッテリー電圧値
test_decode ()	シリアルモニタに 30000 -255 0 が連続で表示された

4.2 Raspberry Pi 単体動作試験

MIRSMG4D-SYST-0008 に従い単体試験を行った結果を表 4 に示す。

表 4 Raspberry Pi 単体試験結果

プログラム名	実施内容？
test_io	GPIO7 と GND につながる 2 ピンにタッチセンサを取り付け ON、OFF の試験を行い正常に動くことを確認した。 プルアップなので、スイッチを押していないときは画面に 1、押されていないときは 0 が表示される。
test_uss	3 ピンのハウジングに超音波センサを取り付け、 Devantech SRF02 の IC2 アドレスを変更しようとしたが、Raspberry Pi が認識せず、i2cdetect コマンドを用いてもアドレスが表示されなかった。 原因はレベル変換器の故障によるものだったため、新品に取り換えると、センサを認識することができた。 超音波センサと test_uss.c のプログラムのアドレスを 0x71、0x72 に変更し実行したところ、それぞれの超音波センサの読み取った距離が標準出力された。
test_camera	この実行ファイルはもともと生成されない。
test_capture	USB カメラをラズパイに接続し実行すると、カメラの画像が保存された。
test_number	USB カメラを数字の書かれたパネルに向け、プログラムを実行する。 標準出力で dist[cm] を聞かれるので、大まかなパネルとカメラの距離を入力すると、写真が撮影される。その後 Enter キーを数回押すと標準出力にパネルの数字が何かを解析した結果が表示される。

	6のパネルで試し標準出力に6と結果が表示された。
test_server	このプログラムは実行する必要がない。
test_request	試験を行っていない。
test_dir_num	試験を行っていない。
test_position	試験を行っていない。

5 B日程で行ってほしいこと

5.1 Raspberry Pi と arduino の統合試験

4.3 での試験できない3つのテストプログラムを [MIRSMG4D-SYST-0007](#) の「4. Arduino と接続しての動作試験」を見て行ってください。

5.2 標準機の組み立て

A日程では標準機の組み立てを行うことができませんでした。B日程で組み立ててください。

5.3 印刷できなかったダンパーを印刷する

失敗したダンパーと交換する

ラボの3Dプリンターの右側→MIRS2005のファイル→dampa0806)

6 制作担当者

制作物	担当者
メカ(分解、部品整理、部品製作)	内山
基板作成、試験	富栞
動作試験、ソフト	和木、関野