

自動運転車の制御研究

システム科学技術学部 知能メカトロニクス学科

2年 B20N015 奥住 公祐

2年 B20N016 奥村 舜

指導教員 システム科学技術学部 知能メカトロニクス学科

職名 准教授 戸花 照雄

1. 目的

現代社会において私たちは交通手段として、自動車や鉄道、船舶や航空機などから多大な恩恵を受けている。これらの機械の多くにはコンピュータが搭載され、電気信号を介して精密に制御されている。今後は多数のセンサを持ち、自律移動が可能な自動運転車が普及すると考えられる。そこで私たちは昨年度の研究を発展させ、自動車模型をマイコンで制御することで、自動運転車についての理解を深めるべく今回の研究を始めた。

2. 原理

ドライバーによる自動車の運転を補助するには、外界の情報を受動素子で電気信号に変換し、その入力値をもとに能動素子であるモータやエンジンを制御する必要がある。そのため、本研究では受動素子である3種のセンサおよび能動素子のDCモータを用いることで自動車制御の入出力を行う。また、マイコン部分にはArduinoを用いる。以下、使用した電子機器の原理を記述していく。

2.1 Arduino

Arduino (アルディーノ) とは、マイコン本体が搭載されたハードのArduinoボード、および統合開発環境であるソフトのArduino IDEから構成された電子機器制御用のシステムである。Arduinoボードは、Atmel社製のAVRマイコン、デジタルおよびアナログの入出力ポートを備えた基板である。また、Arduino IDEはC言語に類似した独自のプログラミング言語をArduinoボードに転送するソフトウェアである。

Arduinoにはいくつかの種類があるが、今回は最も普及しているArduino Unoを使用する。

2.2 センサ

2.2.1 超音波センサ

超音波とは、一般的に人に聞こえない16,000Hz以上の音波のことである。超音波センサはこの超音波を用いることで、人に影響を与えることなく発信源と対象物体の距離を測ることができる。

距離を測る原理は以下のとおりである。

1. マイコンからのトリガをもとにパルス波形を生成する。
2. 生成されたパルス波形を超音波に変換し、対象物に照射する。
3. 対象物で反射された超音波を受信し、照射してからの経過時間をもとに距離を測定する。

2.2.2 フォトリフレクタ

フォトリフレクタは赤外線LEDとフォトトランジスタの2つからなる光センサである。LEDから発せられた赤外線を対象物体に照射し、反射光をフォトトランジスタで受信することにより、対象物体の有無を知ることができる。また、フォトリフレクタには以下の特徴がある。

1. 構造がシンプルなため、非常に安価である。
2. 光の反射を利用するため、対象物体が暗い色であると反射が起きにくい。
3. 光を利用するため、経路中に水蒸気や粒子が存在すると光が散乱する。

2.2.3 加速度センサ

ニュートンの法則 $F = ma$ を変形すると $a = F/m$ となる。この法則より、静止または等速直線運動が変化している物体に力が加わると加速度が生じる。加速度センサではこの原理を用いて物体の加速度を測定する。今回は3次元方向の加速度を測れる3軸加速度センサを用いる。

2.3 DCモータ (直流モータ)

外部電源を必要とする能動素子である。交通機械に用いる場合は主に動力源としてモータ、スクリュー、プロペラを回すために使用される。昨年度の研究ではトランジスタを介してON/OFF制御を行ったが、今回はモータの回転力を数式を用いて制御するためドライバを用いる。

2.4 コントローラ

ドライバに電力を供給するかどうかを人為的に制御するためコントローラを用いた。一般自動車におけるアクセルおよびハンドル部分である。実際の自動車のそれらと異なる点は、本自動車模型のモータは左右にそれぞれ分かれているため、右と左のモータを別々に制御することで旋回および前進を行う点である。ドライバへの電力の入出力にはモーメンタリ型プッシュスイッチを用いる。

3. 実験方法

・使用機器

[ソフトウェア] Arduino IDE

[ハードウェア] Arduino Uno, 超音波センサ(HC-SR04), フォトリフレクタ(LBR-127HLD), 加速度センサ(KXR94-2050), DCモータ, モーメンタリ型プッシュスイッチ, 抵抗やドライバなどの電子素子, 車体用の工作セット(タミヤ 楽しい工作シリーズ TK162)

3.1 ブレッドボードおよびArduinoの配線

昨年度の知識をベースに電子素子の配線を行う。今回は配線の簡単化のためにブレッドボードおよびジャンパーワイヤを用いた。2人での研究であったため、模型を2台製作し、1台はコントローラによる人為的制御、もう一台は加速度センサによる自動制御とした。

研究の開始当初はArduinoおよび各電子素子の使用方法が分からなかったため、実際に配線を行いながら模型を完成させ、その後に回路図を作った。そのため通常と作業工程が逆転している。

3.2 プログラミング

Arduinoと各電子素子との配線に合わせ随時プログラミングする。Arduinoは独自の言語を使用しているが、この言語はC言語と類似しているためC言語の知識を流用する。本プログラミング言語の特徴として以下が挙げられる。

1. `analogRead()`や`digitalRead()`といった独自の関数を用いる。
2. プログラムの開始が`setup()`関数であり、処理完了後は`loop()`関数が延々と繰り返される。

4. 実験結果

4.1 ブレッドボードおよびArduinoの配線

はじめのうちは各センサの配線方法およびArduinoの各PINの役割が分からなかったため、インターネットの情報を利用して配線を行ったことで導線が入り乱れ、回路が混沌としていた。そのため、実際に配線しながらすべての素子の使い方を理解し、研究の最後に素子ごとに利用するPINを分けることで回路を整理した。今回作った模型の外観を図1.1および図1.2, 回路図を図2.1および図2.2に示す。

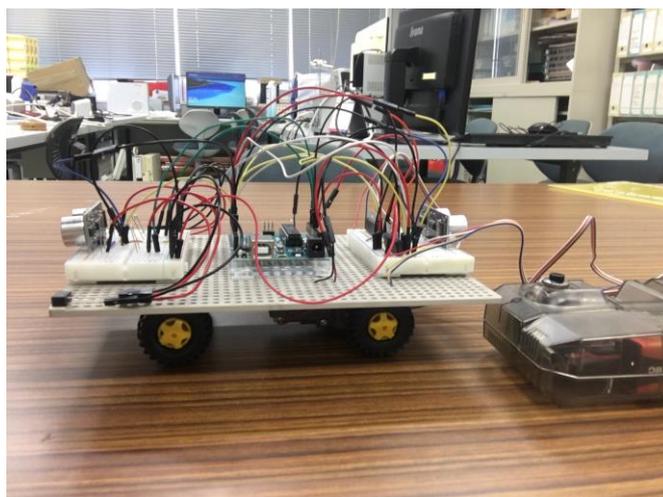


図 1.1 人為的制御模型の外観

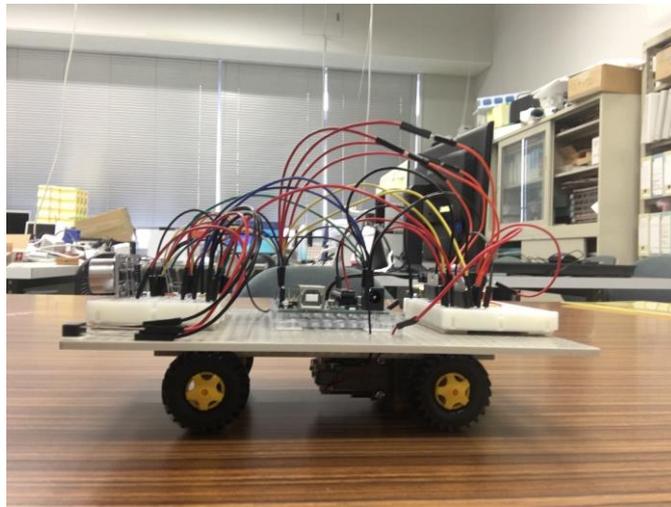


図 1.2 自動制御模型の外観

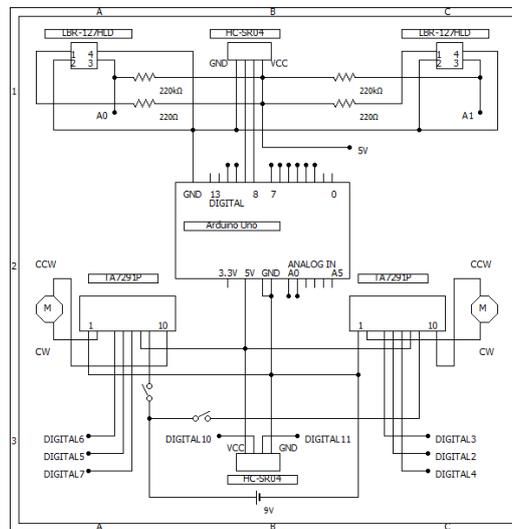


図 2.1 人為的制御回路

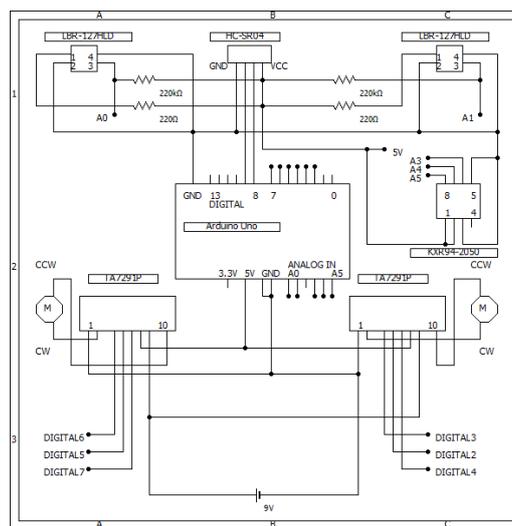


図 2.2 自動制御回路

4.2 プログラミング

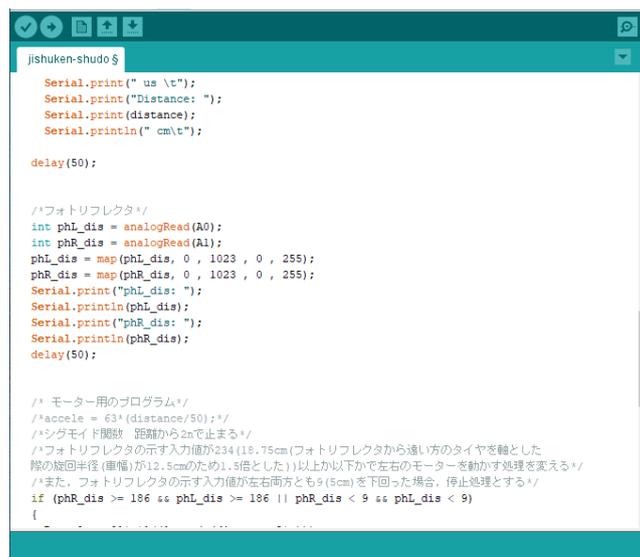
Arduino の制御には独自の言語を用いるが、これは C 言語と類似しているためおおむね円滑に行えた。超音波センサやフォトリフレクタ、加速度センサから受け取った入力値をもとに loop() 関数で制御処理を行い、模型の制御を行った。実際に行った制御は以下に記述する。なお、人為的制御模型と自動制御模型とで処理が異なる工程がある。

1. 前方の超音波センサで受け取った入力値を変数に代入する。(人為的制御模型のみ) 後方の超

音波センサで受け取った入力値も変数に代入する。

2. フォトリフレクタで受け取った入力値を変数に代入する。
3. 自動制御模型においては、加速度センサで受け取った入力値を変数に代入する。
4. 1. および 3. で得られた変数を数式に代入し、モータへの電圧値を決定する。前方付近に障害物がない場合は前進するよう出力値を定める。前方付近に障害物がある場合、人為的制御模型のみコントローラから入力があるとき後退、自動制御模型のみ停止するよう出力値を定める。
5. 4. の処理で前進するよう出力値が定まったとき、遠方の左右どちらかにのみ障害物がある場合、2 で得られた入力値により障害物を避けるよう片方のモータへの出力を止める。
6. 1～5 の処理を繰り返す。

以上の処理を記載したソースコードを図 3 に示す。



```
jishuken-shudo $  
  
Serial.print(" us \t");  
Serial.print("Distance: ");  
Serial.print(distance);  
Serial.println(" cm\t");  
  
delay(50);  
  
/*フォトリフレクタ*/  
int pH_L_dis = analogRead(A0);  
int pH_R_dis = analogRead(A1);  
pH_L_dis = map(pH_L_dis, 0, 1023, 0, 255);  
pH_R_dis = map(pH_R_dis, 0, 1023, 0, 255);  
Serial.print("pH_L_dis: ");  
Serial.println(pH_L_dis);  
Serial.print("pH_R_dis: ");  
Serial.println(pH_R_dis);  
delay(50);  
  
/* モーター用のプログラム */  
/*accele = 63*(distance/50);*/  
/*シグモイド関数 距離から2mで止まる*/  
/*フォトリフレクタの示す入力値が234(18.75cm(フォトリフレクタから遠い方のタイヤを軸とした  
際の旋回半径(幅幅)が12.5cmのため1.5倍とした))以上か以下かで左右のモーターを動かす処理を変える*/  
/*また、フォトリフレクタの示す入力値が左右両方とも9(5cm)を下回った場合、停止処理とする*/  
if (pH_R_dis >= 186 && pH_L_dis >= 186 || pH_R_dis < 9 && pH_L_dis < 9)  
{
```

図 3 Arduino IDE に記載したソースコードの一部

5. 考察

研究当初は超音波センサやフォトリフレクタから得られた自動車模型と前方物体との距離を比例制御で数式に代入し、自動車の衝突回避を実現しようとしていた。しかし実際にプログラミングを行うと、比例制御では模型が停止するまで速度変化が極めて遅く、かかる時間もかなり長いことが発覚した。そのため、私たちは自動車の緊急停止処理を行う関数を比例関数からシグモイド関数に変更することで障害物との距離が近くなった場合の急激な速度低下および停止を実現することができた。

また Arduino を用いた機械製作では、Arduino IDE 独自の入力値測定機能であるシリアルプロッタおよびシリアルモニタを使うことができ、センサが正常に作用することを外部素子で確認することなく行えたため非常に便利であった。しかし、Arduino は複数の電子機器が組み合わさって構成されているため、価格が PIC などの一般的なマイコンと比べて高価である。よって、Arduino などの工学初学者用のマイコン基板セットは実験用に使い、製品を量産化する場面に安価なマイコンを用いるべきである。

全体の考察として、2 年間にわたる自主研究を経て知能メカトロニクス技術の 1 つである自動運転車の構築をソフトとハードの両面から行えたことは素晴らしかったといえる。是非下級生にも来年度以降の自主研究を勧めたい。

6. 結論

自動車模型を Arduino およびセンサにより制御することで、自動運転車についての理解を深められた。

7. 参考文献

[1] 高本孝頼, みんなの Arduino 入門, 株式会社リックテレコム, 2014 年 2 月 17 日初版発行