



## (2) ロボットアームの角度制御

### 1 実験目的

制御系設計には

- 運動方程式や回路方程式などから制御対象のモデル（微分方程式，伝達関数など）を導出する
- モデルに含まれるパラメータを実験により決定する（パラメータ同定）
- 制御対象の出力（制御量）を所望の値にするような制御対象の入力（操作量）を決めるためにコントローラの設計を行う

というモデリングの作業と

- 制御対象の出力（制御量）を所望の値にするような制御対象の入力（操作量）を決めるためにコントローラの設計を行う

というコントローラ設計の作業が含まれる．本実験では，平面上に回転するアームの角度制御を例として，制御系設計の一連の流れを習得することを目的としている．

### 2 ロボットアーム実験装置の概要と制御目的

#### 2.1 ロボットアーム実験装置

図 2.1 に示すように，ロボットアーム実験装置はアームが水平にして DC モータにより回転するようになっている．DC モータを駆動させるためにパソコン上で計算された指令電圧は，I/O ボード "Q8-USB" により D/A 変換され，アンプを介して DC モータに入力される．さらに，アームの角度はシャフトに取り付けてあるポテンショメータによって角度として検出され，I/O ボード "Q8-USB" により A/D 変換されて後，パソコンに取り込まれている．

#### 2.2 アクチュエータと D/A 変換

コンピュータから出力される指令電圧はデジタル値であるため，D/A 変換によってアナログ値に変換される（図 2.2 に示す）．アナログ周期ごとに一定値のアクションに変換してからアンプに加える必要がある．指令電圧の I/O ボード "Q8-USB" による D/A 変換には，16 ビットが使われており，分解能は  $0.30518[V] = (\pm 10 - (-10))/2^{16}[V]$  の刻み幅である．

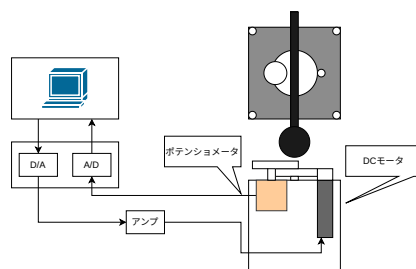


図 2.1 ロボットアーム実験装置

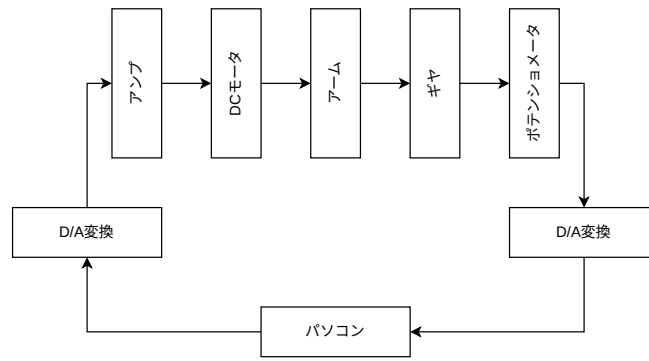


図 2.2 制御対象のブロック線図

## 2.3 センサと A/D 変換

本実験装置では，ロボットアームの角度を検出するためのセンサにポテンシオメータが用いられている．ポテンシオメータは可変抵抗であり角度に比例した電圧（アナログ量） $V_s[V]$  を返す．センサの出力電圧とロボットアームの角度との関係は， $1[V]$  あたり  $-35[deg]$  であり， $-175 \sim 175[deg]$  の角度を検出可能である．

コンピュータでこのセンサ電圧  $V_s(t)[V]$  を利用するには，A/D 変換によってアナログ量をサンプリング周期  $[s]$  でデジタル量に変換する必要がある．A/D 変換では， $V_s(kh)[V]$  のように  $V_s(t)$  を横軸（時間軸）方向に離散化する標本化の作業と， $V_s(kh)$  を縦軸方向に離散化する量子化の作業を行う．本実験装置の I/O ボード "Q8-USB" 上の A/D 変換部はレンジ： $\pm 5[V]$  または  $\pm 10[V]$ ，分解能： $16$  ビットである．本実験では，レンジを  $\pm 5[V]$  に設定しているため，約  $0.15259[mV] = (\pm 5 - (-5))/2^{16}[V]$  の刻み幅（量子化サイズ）で量子化することになる．

## 2.4 制御目的と P 制御

本実験では，ロボットアームの回転角度を所望の角度ですばやく止める角度制御を考える．角度制御を実現するために考えられる最も単純な方法は，

- 目標角  $\theta_{\text{ref}}(t)$  と回転角  $\theta(t)$  の差（偏差と呼ぶ）を計算
- その差が正（負）に大きければ正（負）の大きな電圧  $v(t)$  を加える
- 正（負）に小さければ正（負）の小さな電圧  $v(t)$  を加える

という方法である．このように偏差  $e(t) = \theta_{\text{ref}}(t) - \theta(t)$  に比例した電圧  $v(t)$  を加えるような制御方法を P 制御（比例制御）という．また，

$$v(t) = k_P e(t) \iff v(s) = k_P e(s) \quad (2.1)$$

という形式のコントローラを P コントローラと呼ぶ．

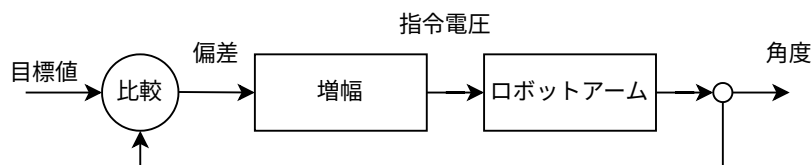


図 2.3 P 制御

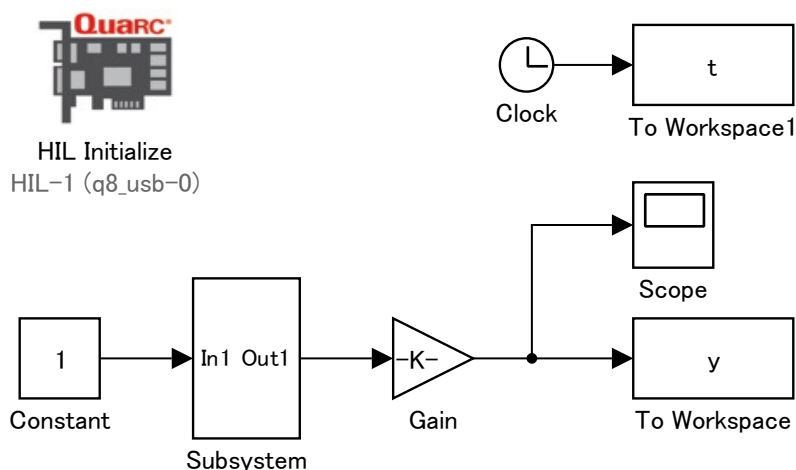


図 2.4 P 制御

## 2.5 PI 制御

P 制御では様々な要因でステップ状の目標値に対する定常偏差が生じることがある．この定常偏差を零にするためには，コントローラに積分動作  $k_I/s$  を含ませればよく，

$$v(t) = k_P e(t) + k_I \int_0^t e(\tau) d\tau \iff v(s) = \left( k_P + \frac{k_I}{s} \right) e(s) \quad (2.2)$$

という形式のコントローラを PI コントローラと呼ぶ．

## 3 実験 1 – A/D, D/A 変換の動作確認と P, PI 制御 –

### 3.1 実験装置のセッティング

ロボットアーム実験装置が図 3.1 のように結線されていることを確認する．

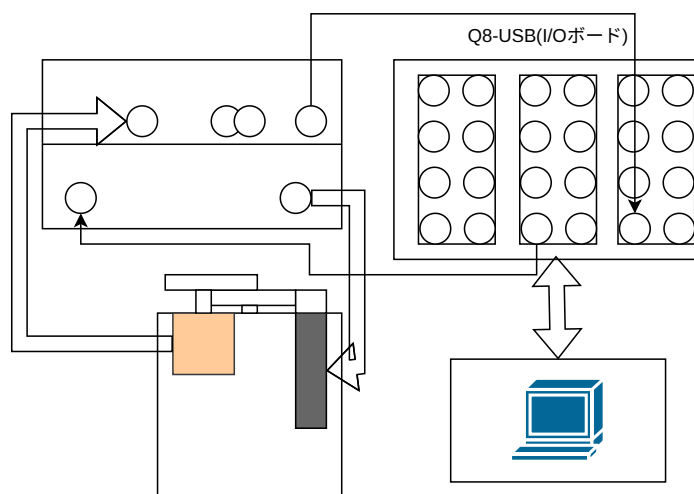


図 3.1 ロボットアーム実験装置の結線

## 3.2 D/A 変換とアクチュエータの動作確認

1. D/A 変換の動作確認をするために図 3.2 の Simulink モデル “da\_conv.slx” を作成する．保存場所は D:\#student\_5S\#group01\#da (“01” は班の番号であり，2 班なら “01” を “02” に読み替え）とする．Simulink モデルをビルドし，エラーがないことを確認した後，

```
>> print -s -dpdf da_conv.pdf 結果確認 (3.7 節) で使うので必須
```

```
>> print -s -dbitmap da_conv.bmp Word でレポートを書く場合
```

と入力して Simulink モデル “da\_conv.slx” の pdf ファイル “da\_conv.pdf” を生成する．さらに，生成した pdf ファイルの余分な空白を取り除くため，“da\_conv.pdf” をデスクトップ上の “bcpdfcrop-multi.bat” にドラッグする．その結果，“da\_conv-crop.pdf” が生成される．

なお，QuaRC により利用できる Simulink ブロック HIL Initialize や HIL Write Analog については補足 1.1，パラメータ設定については補足 1.2, 1.3, ビルドと実行の方法については補足 1.4 を参照すること（別紙「補足事項：QuaRC の使用方法」）．

2. Simulink モデル “da\_conv.slx” のブロック Manual Switch を “Constant” 側にする．Simulink モデルの実行を開始すると，DC モータに一定の電圧が加わり，アームが一定速度で回転することを確認せよ．
3. テスタを Universal Power Module の “OUT” と “GND” に接続する．図 3.2 の Constant ブロックを表 3.1 のように設定したときのテストの測定値およびアームの動作を観測し，表 3.1 を完成せよ．
4. Universal Power Module の “To Load” に接続されているケーブルの DC モータ側を取り外し，オシロスコープを Universal Power Module の “OUT” と “GND” に接続する．（オシロスコープの電圧レンジは 5 [V/DIV]，時間レンジは 2 [msec/DIV] に設定する．

Simulink モデル “da\_conv.slx” のブロック Manual Switch を “Sine Wave” 側にする．ブロック “Sine Wave” の周波数を “ $\pi/0.008$ ” [rad/s]，振幅を “5”，“10”，“15” と設定したとき，D/A 変換された電圧をオシロスコープで観測し，デジタルカメラで撮影せよ（付録 A.1 の 図 A.1 を参照）．

### 3.2.1 実験結果

以下に結果を示す．

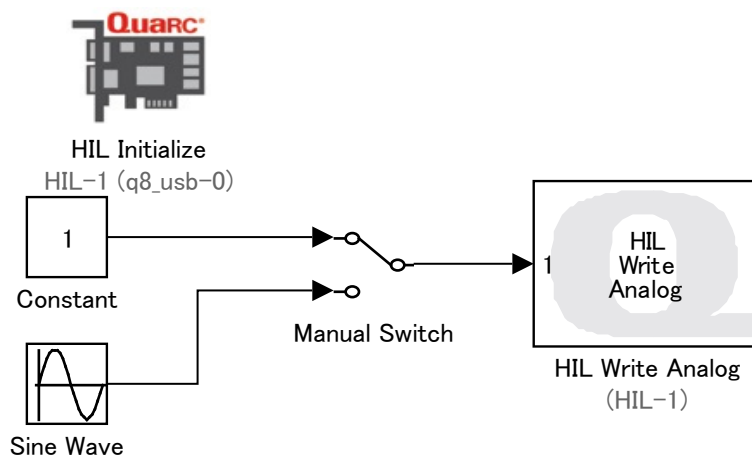
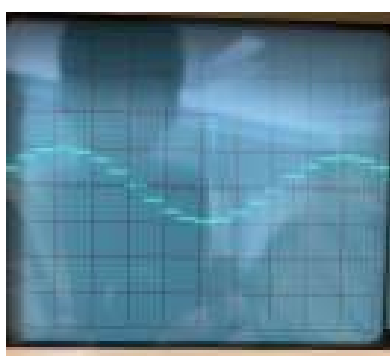


図 3.2 da\_conv

表 3.1 D/A 変換の動作確認とアームの回転方向

ブロック Constant の設定値 [V]	テストによる測定値 [V]	回転方向 (時計回り, 反時計回り, 静止)
-5	-5	cww
-2.5	-2.494	cww
-1	-0.994	csww
-0.5	-0.495	cww
0	$4.7 \times 10^{-3}$	静止
0.5	0.504	cw
1	1.003	cw
2.5	2.502	cw
5	5	cw



(a)



(b)



(c)

図 3.3 オシロスコープ a=5,b=10,c=15

### 3.2.2 実験考察

da\_conv により設定値 [V] がマイナスのとき反時計回り (ccw), プラスのときに時計回り (cw) となることが表 3.1 よりわかる。また, 出力値をテスターで計測したところほぼ設定値と同程度の電圧が測定できた。これにより constant の値をそのまま出力していることがわかる。

次にオシロスコープでは, 振幅を 5,10,15 と変化させ観測したが, すべて 16 個の線により表されていた。これは以下の式よりわかる。

周波数  $f$  を  $\omega = \frac{\pi}{0.008} [\text{rad/s}]$  の場合に求めるには, 角周波数  $\omega$  と周波数  $f$  の関係を使用する。

$$\omega = 2\pi f \quad (3.1)$$

この式からより

$$f = \frac{\omega}{2\pi} \quad (3.2)$$

となり, 角周波数  $\omega = \frac{\pi}{0.008}$  の場合,

$$f = \frac{\frac{\pi}{0.008}}{2\pi} = \frac{1}{2 \times 0.008} = \frac{1}{0.016} = 62.5 \text{ [Hz]} \quad (3.3)$$

つまり，角周波数が  $\frac{\pi}{0.008} \text{ [rad/s]}$  の場合，周波数  $f$  は  $62.5 \text{ [Hz]}$  となる．よって

$$T = \frac{1}{f} \quad (3.4)$$

$$T = \frac{1}{62.5} = 0.016 \text{ [s]} \quad (3.5)$$

したがって，周期  $T$  は  $0.016 \text{ [s]}$  となる．そのため，16 個の線により表された．

### 3.3 A/D 変換とセンサの動作確認

1. A/D 変換の動作確認をする図 3.3 の Simulink モデル“ad\_conv.slx”を作成する．保存場所は D:\#student\_5S\#group01\#ad とする．ビルドしてエラーがないことを確認した後，print コマンドにて“ad\_conv.pdf”という名前の pdf ファイルを生成し，デスクトップ上の“bcpdfcrop-multi.bat”により余白を取り除く．
2. テスタを Universal Power Module の“S1”と“GND”に接続し，アームを  $\pm 45$  度に動かす（目測で良い）．そのときの Display 値とテストの値を観測し，表 3.2 を完成せよ．
3. 表 3.2 の結果から，センサ電圧  $1 \text{ [V]}$  あたりの角度変位を求めよ（符号に注意）．

$$\text{センサ電圧 } 1 \text{ [V] あたりの角度変位} = \quad \quad \quad [\text{deg}] = \quad \quad \quad [\text{rad}] \quad (3.6)$$

4. アームを 1 回転させたとき，センサ電圧の範囲をテストにより測定せよ．

$$\text{センサ電圧：最小値} = \quad \quad \quad [\text{V}], \quad \text{最大値} = \quad \quad \quad [\text{V}] \quad (3.7)$$

#### 3.3.1 実験結果

以下に結果を示す．

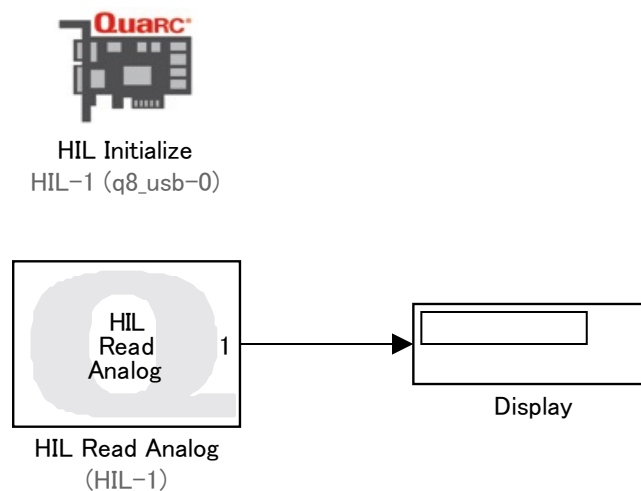


図 3.4 ad\_conv

表 3.2 A/D 変換の動作確認（角度：反時計回りが正）

アームの回転角度 [deg]	テストによる測定値 [V]	Display による測定値 [V]
45	<b>1.030</b>	<b>1.031</b>
0	—	—
-45	<b>-1.520</b>	<b>-1.521</b>

$$\text{センサ電圧 } 1 \text{ [V] あたりの角度変位} = -35.266 \text{ [deg]} = -0.6155 \text{ [rad]} \quad (3.8)$$

$$\text{センサ電圧：最小値} = -4.95 \text{ [V]}, \quad \text{最大値} = 5.05 \text{ [V]} \quad (3.9)$$

### 3.3.2 実験考察

この実験では、ポテンショメータを使用し角度を検出した。ポテンショメータはロータリエンコーダなどのパルス信号と異なり、アナログ電圧を出力する絶対的な位置情報を持つアナログセンサである。実験では 45, -45 度のときの電圧を計測し、計 90 度で 2.552[V] 変化した。これにより 1 [V] あたりの角度変位を求めることができた。

## 3.4 ロボットアームの表現

- 以下の動作をするような図 3.4 の Simulink モデルを完成させる。
  - アンプに加える電圧が  $-10 \sim 10 \text{ [V]}$  の範囲を超えないようにする。
  - コンピュータから正の指令電圧を与えたとき、アームが反時計回りに回転する。
  - センサ電圧を角度  $[\text{rad}]$  で出力する。ただし、この実験のみ Scope は deg で表示させる。
  - 終了時間が 8 秒となるようにする。

各ブロックのパラメータ設定は各自で考えることとし、完成した Simulink モデルは “ad\_da\_conv.slx” という名前でディレクトリ D:\#student\_5S\#group01\#ad.da に保存する。ビルドしてエラーがないことを確認した後、print コマンドにて “ad\_da\_conv.pdf” という名前の pdf ファイルを生成し、デスクトップ上の “bcpdfcrop-multi.bat” により余白を取り除く。

- 3.2 節 (4) で外したケーブルを接続する。
- オシロスコープを Universal Power Module の “S1” と “GND” に接続し、センサ電圧が 0[V] になるように、アームを動かす。
- Scope をダブルクリックして開き、リアルタイムで角度データを観測できるようにする。Scope のレンジは付録 A.1 の図 A.2 に合わせる。設定方法は図 3.5 に示す。
- すべての準備が整ってから “ad\_da\_conv.slx” を実行する。実行終了後、コマンドウィンドウで

```
>> save ad_da_data t y
```

と入力し、“ad\_da\_data.mat” という名前の mat ファイルでデータを保存する。また、配布する M ファイル

- “autoplot\_ad\_da.m”

を実行することによって、MATLAB 上でグラフを作成する。グラフの pdf ファイルは自動的に生成される。



### 3.4.1 実験結果

以下に結果を示す．

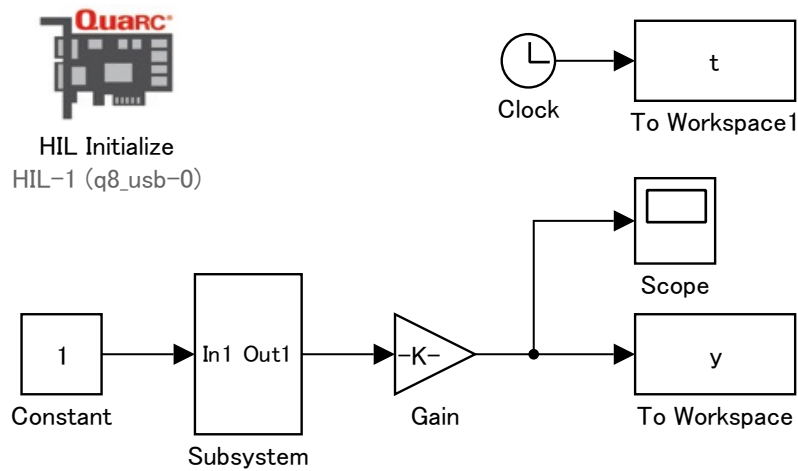


図 3.5 ad\_da\_conv

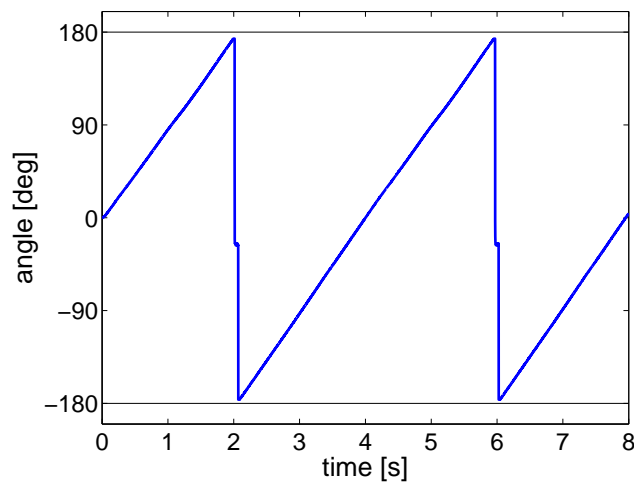


図 3.6 A/D,D/A 変換の動作確認

### 3.4.2 実験考察

ポテンショメータには回転範囲に制限があるため，設定された範囲（ここでは 180 度）に到達すると，値が再び最小値（-180 度）に戻っている．これは，ポテンショメータが無限に回転するのではなく，ある範囲内でのみ動作するように設計されているためだと考えられている．

## 3.5 P 制御

ここではアームの角度制御のために P コントローラ

$$v(t) = k_P (\theta^{\text{ref}}(t) - \theta(t)) \quad (\theta^{\text{ref}}(t) : \text{目標値}, \theta(t) : \text{現在の角度}) \quad (3.10)$$

を用い，比例ゲイン  $k_P$  を大きくするにしたがって過渡特性，定常特性がどのように変化するかを調べる．

1. P 制御を行うため，図 3.6 の Simulink モデル “ex\_pcont.slx” を作成する．モデルはディレクトリ D:\#student\_5S\#group01\#pcont に保存する．つぎに，

```
>> kP = 2;
```

と入力した後，ビルドを行いエラーがないことを確認する．また，“ex\_pcont.pdf” という名前の pdf ファイルを生成し，デスクトップ上の “bcpdfcrop-multi.bat” により余白を取り除く．

2. Scope と Scope1 を開き，リアルタイムで角度と操作量を観測できるようにする．レンズは付録 A.1 の図 A.3 に合わせる．
3. センサ電圧が 0 [V] となる位置にアームを動かし，“ex\_pcont.slx” を実行する．実行終了後，

```
>> save pcont_kP02_data t u y kP
```

と入力し，mat ファイル “pcont\_kP02.data.mat” にデータを保存する．

同様に， $k_P = 4$ ,  $k_P = 20$  と設定して “ex\_pcont.slx” を実行し，実験結果のデータを

- “pcont\_kP04.data.mat” ( $k_P = 4$ )
- “pcont\_kP20.data.mat” ( $k_P = 20$ )

という名前の mat ファイルに保存する．

最後に，配布する M ファイル

```
>> autoplot_pcont.m
```

を実行することによって，MATLAB 上でグラフを作成する．グラフの pdf ファイルは自動的に生成される．

### 3.5.1 実験結果

以下に結果を示す．

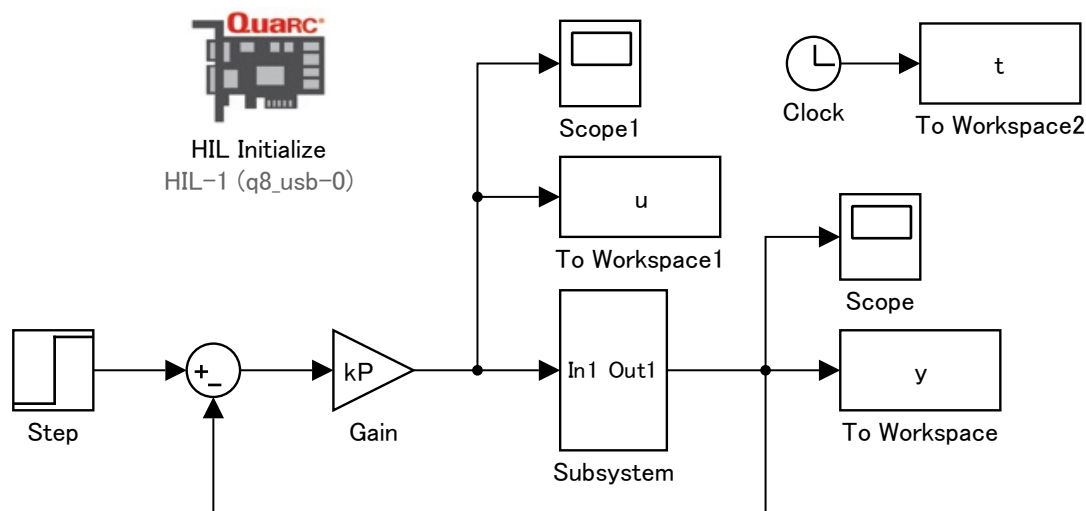


図 3.7 ex\_pcont

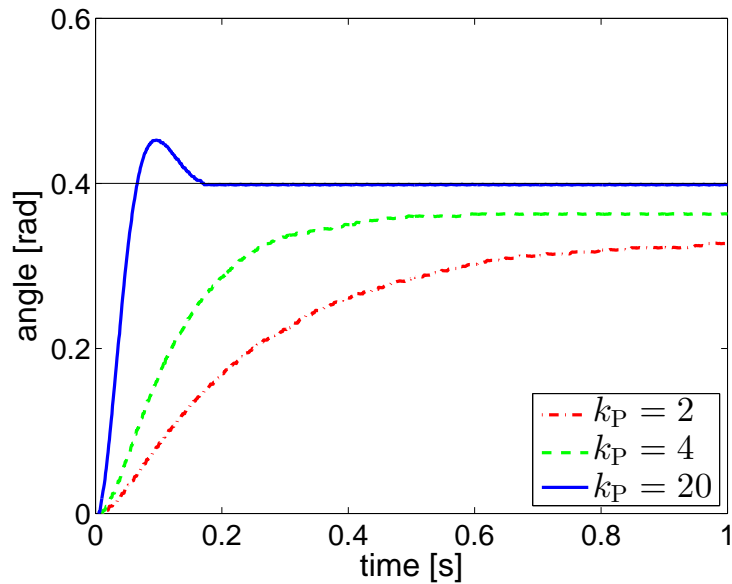


図 3.8 figure\_pcont\_angle

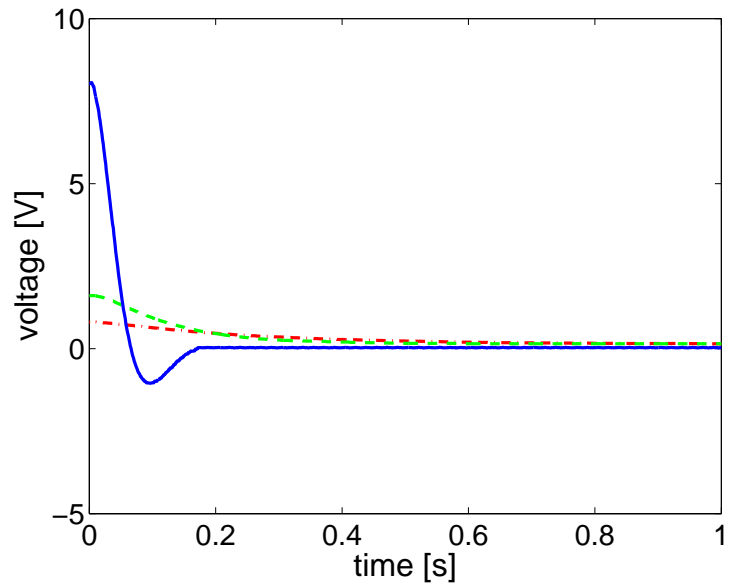


図 3.9 figure\_pcont\_volt

### 3.5.2 実験考察

図 3.8 と図 3.9 より，比例ゲイン  $k_P$  を増加させると，過渡特性における立ち上がり時間が短くなり，応答がより速くなることが観察できる．これは，比例ゲイン  $k_P$  が大きいほど制御系が目標値に迅速に追従しようとするためであり，その結果として立ち上がり時間が短縮される．一方で，比例ゲインが大きくなるとオーバーシュートの値も増加する傾向がある．これは，ゲインが高いことで目標値に達する速度が増加しすぎ，目標値を超過してしまうためである．また，操作量（入力電圧）の大きさも比例ゲイン  $k_P$  によって増加する．これは，制御対象が目標値から離れているときに強い制御力が働くためであり，制御量が過大になりやすい．図 3.9 に示されるように， $k_P$  の増加に伴って初期の入力電圧が大きくなるが，これは目標角度に迅速に追従するために必要な制御信号が大きくなるためである．次に，定常特性（定常偏差）について考察する．比例制御のみでは，制御対象に定常偏差が残ることがある．この理由は，比例制御が誤差に比例した制御量を出力するため，目標値に完全に一致するためには無限大のゲインが必

要となるためである．したがって，有限の  $k_P$  のもとでは完全に目標値に一致することは難しい．理論的には，最終値の定理を用いて定常偏差を考察することができる．入力がステップ応答の場合，最終値の定理より，システムの入力  $y(t)$  の定常値は以下で与えられる．

$$\lim_{t \rightarrow \infty} y(t) = \lim_{s \rightarrow 0} s \cdot Y(s) \quad (3.11)$$

ここで， $Y(s)$  はシステムの入力のラプラス変換である．比例制御のみを用いたシステムでは，定常偏差がゼロにならないため，入力と出力の間に誤差が残る．これは，実機の摩擦や負荷変動などの外乱要因が存在するためでもある．ポテンショメータやエンコーダのセンサー精度や摩擦力，モータの不調などが影響を及ぼし，理想的な追従が難しい要因ともなっていると考えられる．

### 3.6 積分動作の効果 (PI 制御)

1. PI 制御を行うため，図 3.7 の Simulink モデル “ex\_picont.slx” を作成する．モデルはディレクトリ D:\student.5S\group01\picont に保存する．つぎに，

```
>> kP = 4; kI = 0;
```

と入力した後，ビルドを行いエラーがないことを確認する．また，“ex\_picont.pdf” という名前の pdf ファイルを生成し，デスクトップ上の “bcpdfcrop-multi.bat” により余白を取り除く．

2. Scope と Scope1 を開き，リアルタイムで角度と操作量を観測できるようにする．レンズは付録 A.1 の図 A.4 に合わせる．
3. センサ電圧が 0 [V] となる位置にアームを動かし，“ex\_picont.slx” を実行する．実行終了後，

```
>> save picont_kP04_kI00_data t u y kP kI
```

と入力し，mat ファイル “picont\_kP04\_kI00.data.mat” にデータを保存する．

比例ゲインを  $k_P = 4$  に固定し，積分ゲインを  $k_I = 6$ ， $k_I = 12$  として “ex\_picont.slx” を実行し，実験結果のデータを

- “picont\_kP04\_kI06.data.mat” ( $k_P = 4, k_I = 6$ )
- “picont\_kP04\_kI12.data.mat” ( $k_P = 4, k_I = 12$ )

という名前の mat ファイルに保存する．最後に，配布する M ファイル “autoplot\_picont.m” を実行することによって，MATLAB 上でグラフを作成する．グラフの pdf ファイルは自動的に生成される．

4. 目標値を 0 に設定 (Step の最終値を “0” に設定) し，また，終了時間を “inf” に設定する．比例ゲインを  $k_P = 4$  に固定し，積分ゲインを  $k_I = 0, k_I = 12$  として実験を行い，手でアームに力を加えたとき (入力外乱を加えることに相当)，反力がどうなるか調べる．

#### 3.6.1 実験結果

以下に結果を示す．

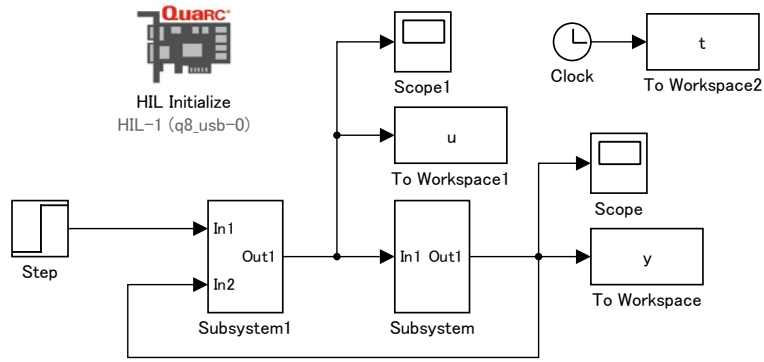


図 3.10 ex\_picont

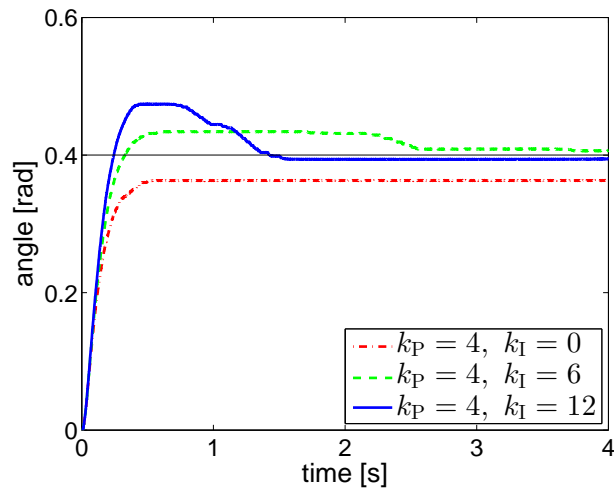


図 3.11 figure\_picont\_angle

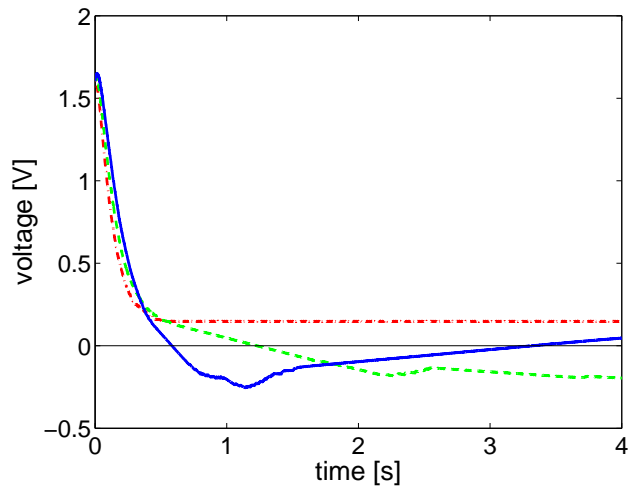


図 3.12 figure\_picont\_volt

### 3.6.2 実験考察

積分動作の導入により，定常偏差が収束したことが図 3.11 よりわかる．定常偏差が収束した理由は，積分動作が偏差の累積に基づいて補正を行うためである．制御系において，積分動作は偏差が存在する限り制御信号を加え続け，目標値に到達するまでその効果を増大させる．これにより，定常偏差が時間とともに蓄積されて制御信号として反映

され，結果として定常偏差が収束する形で解消されるため，最終的にシステムは目標値に到達する．さらに，アームに外力が加わった場合，P 制御の場合，差分におおじた出力のみであったが，PI 制御の場合，偏差が蓄積され，徐々に出力が大きくなる．

## 4 モデルに基づく改良型 PID 制御

ここではまず，ロボットアームのモデリングについて説明する．ついで，このモデルに基づいて部分的モデルマッチング法（北森の方法）による改良型 PID コントローラのパラメータ調整について説明する．

### 4.1 ロボットアームのモデリング

#### 4.1.1 運動方程式と伝達関数

ロボットアームの運動方程式は

$$J\ddot{\theta}(t) = -c\dot{\theta}(t) + \tau(t) \quad (4.1)$$

で与えられる．ただし， $\tau(t)$  [N・m]: アームの回転軸に加わるトルク， $c$ : アームの回転軸の粘性摩擦係数， $J$  [kg・m<sup>2</sup>]: アームの慣性モーメントである．(4.1) 式に，DC モータ，アンプの特性式やギア比を考慮すると，ロボットアームシステム全体の数学モデルは，

$$\ddot{\theta}(t) = -a\dot{\theta}(t) + bv(t) \quad (4.2)$$

という形式となる．ここで， $v(t)$  [V]: モータを駆動させるための指令電圧， $a, b$ : アーム，DC モータ，アンプの特性やギア比に関わるパラメータである．

角度制御を考えた場合，制御量は  $\theta(t)$ ，操作量は  $v(t)$  であるから， $v(s) = \mathcal{L}[v(t)]$  から  $\theta(s) = \mathcal{L}[\theta(t)]$  への伝達関数  $P(s)$  は

$$P(s) = \frac{\theta(s)}{v(s)} = \frac{b}{s(s+a)} \quad (4.3)$$

となる．

#### 4.1.2 パラメータ同定

(4.3) 式に含まれるパラメータ  $a, b$  は測定器で測ることができない未知パラメータである．そこで，実験によりパラメータ同定を行う．

$$y_d(t) = \dot{\theta}(t) \quad (4.1)$$

とすると (4.2) 式は

$$\dot{y}_d(t) = -ay_d(t) + bv(t) \quad (4.2)$$

となるから，

$$v(s) = \mathcal{L}[v(t)] \quad (4.3)$$

から

$$y_d(s) = \mathcal{L}[y_d(t)] \quad (4.4)$$

への伝達関数  $P_d(s)$  は 1 次遅れ要素

$$P_d(s) = \left( \frac{y_d(s)}{v(s)} \right) = \frac{b}{s+a} \quad (4.5)$$

となる。(4.5) 式を 1 次遅れ要素の標準形で表すと

$$P_d(s) = \frac{K}{1 + Ts}, \quad T = \frac{1}{a}, \quad K = \frac{b}{a} \quad (4.6)$$

であるから,  $v(t) = 1$  [V] を加えたときの角速度 (単位ステップ応答)  $y_d(t) = \dot{\theta}(t)$  は図 4.3 のようになる。したがって, 時定数  $T$ , ゲイン  $K$  を求めれば, 未知パラメータ  $a, b$  が次式のように定まる。

$$a = \frac{1}{T}, \quad b = Ka = \frac{K}{T} \quad (4.7)$$

## 4.2 部分的モデルマッチング法による PID コントローラの設計

### 4.2.1 P-D 制御 (微分先行型 PD 制御)

P-D コントローラ (微分先行型 PD コントローラ)

$$v(t) = k_P e(t) - k_D \frac{d}{dt} \theta(t) \iff v(s) = k_P e(s) - k_D s \theta(s) \quad (4.5)$$

を用いると, (4.5), (4.8) 式より  $\theta^{\text{ref}}(s)$  から  $\theta(s)$  への伝達関数は 2 次遅れ要素

$$T(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}, \quad \omega_n = \sqrt{bk_P}, \quad \zeta = \frac{a + bk_D}{2\omega_n} \quad (4.6)$$

となる。したがって, (4.9) 式と 2 次遅れ系の標準モデル

$$y_M(s) = T_M(s)\theta^{\text{ref}}(s), \quad T_M(s) = \frac{\omega_M^2}{s^2 + 2\zeta_M\omega_M s + \omega_M^2} \quad (4.7)$$

の伝達関数  $T_M(s)$  と完全に一致させるには, 比例ゲイン  $k_P$ , 微分ゲイン  $k_D$  を

$$k_P = \frac{\omega_M^2}{b}, \quad k_D = \frac{2\zeta_M\omega_M - a}{b} \quad (4.8)$$

と定めればよい。

なお, ポテンショメータによって検出された角度には高周波成分の観測雑音 (ノイズ) が含まれているため, 検出された角度をもとに角速度を算出すると, インパルス状の成分を含んでしまう。そこで, 実際には, 検出された角度を 1 次のローパスフィルタ

$$G_f(s) = \frac{1}{1 + T_f s} \quad (4.9)$$

(4.12) に通して高周波成分の観測雑音を除去した後, 角度信号を微分する必要がある。以上のことを考慮すると, P-D コントローラは

$$v(s) = k_P e(s) - k_D G_f(s) s \theta(s) = k_P e(s) - \frac{k_D s}{1 + T_f s} \theta(s) \quad (4.10)$$

(4.13) となる。

### 4.2.2 I-P 制御 (比例先行型 PI 制御)

I-P コントローラ (比例先行型 PI コントローラ) は

$$v(t) = -k_P \theta(t) + k_I \int_0^t e(\tau) d\tau \iff v(s) = -k_P \theta(s) + \frac{k_I}{s} e(s) \quad (4.14)$$

を用いると, (4.5), (4.14) 式より  $\theta(s)$  から  $\theta(s)$  への伝達関数は 3 次遅れ要素

$$T(s) = \frac{bk_I}{s^3 + as^2 + bk_P s + bk_I} \quad (4.15)$$



となる．ここでは，(4.15) 式を 2 次遅れ系の規範モデル (4.10) 式の伝達関数  $T_M(s)$  と近似的に一致させるため，

$$\frac{1}{T(s)} = 1 + \frac{k_P}{k_I}s + \frac{a}{k_I}s^2 + \frac{1}{bk_I}s^3 \quad (4.16)$$

$$\frac{1}{T_M(s)} = 1 + \frac{2\zeta_M}{\omega_M}s + \frac{\omega_M^2}{\omega_M^2}s^2 \quad (4.17)$$

の 2 次までの項が一致するように比例ゲイン  $k_P$ ，積分ゲイン  $k_I$  を次式のように決定する．

$$k_P = \frac{2\zeta_M\omega_M a}{b}, \quad k_I = \frac{\omega_M^2 a}{b} \quad (4.18)$$

このように規範モデルと部分的に一致させるようなパラメータ調整法を部分的モデルマッチング法と呼ぶ．

#### 4.2.3 I-PD 制御（比例・微分先行型 PID 制御）

I-PD コントローラ（比例・微分先行型 PID コントローラ）

$$v(t) = -k_P\theta(t) + k_I \int_0^t e(\tau)d\tau - k_D \frac{d\theta(t)}{dt} \iff v(s) = -k_P\theta(s) + \frac{k_I}{s}e(s) - k_D s\theta(s) \quad (4.19)$$

を用いると，(4.5)，(4.19) 式より  $\theta(s)$  から  $\theta(s)$  への伝達関数は 3 次遅れ要素

$$T(s) = \frac{bk_I}{s^3 + (a + bk_D)s^2 + bk_Ps + bk_I} \quad (4.20)$$

となる．したがって，(4.20) 式を規範モデル

$$y_M(s) = T_M(s)\theta_{\text{ref}}(s), \quad T_M(s) = \frac{\omega_M^3}{s^3 + \alpha_{M2}\omega_M s^2 + \alpha_{M1}\omega_M^2 s + \omega_M^3} \quad (4.21)$$

の伝達関数  $T_M(s)$  と完全に一致させるには

$$k_P = \frac{\alpha_{M1}\omega_M^2}{b}, \quad k_I = \frac{\omega_M^3}{b}, \quad k_D = \frac{\alpha_{M2}\omega_M - a}{b} \quad (4.22)$$

と選べばよい．ただし， $\omega_M$  は速度応答性に関するパラメータ， $\alpha_{M1}$ ， $\alpha_{M2}$  は減衰性に関するパラメータであり，

$$\begin{aligned} \text{バターワース標準形：} & \alpha_{M1} = 2, \quad \alpha_{M2} = 2 \\ \text{二項係数標準形：} & \alpha_{M1} = 3, \quad \alpha_{M2} = 3 \\ \text{ITAE 最小標準形：} & \alpha_{M1} = 2.15, \quad \alpha_{M2} = 1.75 \end{aligned} \quad (4.11)$$

が用いられることが多い．なお，ITAE は Integral of Time multiply Absolute Error の略字である．実際には高周波成分の観測雑音を除去するため，次式の I-PD コントローラを用いることになる．

$$v(s) = -k_P\theta(s) + \frac{k_I}{s}e(s) - \frac{k_D s}{1 + T_f s}\theta(s) \quad (4.23)$$

## 5 実験 2: 改良型 PID 制御の比較

### 5.1 ローパスフィルタによる観測雑音除去とパラメータ同定

(1) 図 5.1 の Simulink モデル “ex.ident.slx” を作成し，ディレクトリ D:#student\_5S#group01#ident に保存する．図 5.1 におけるブロック Transfer Fcn はローパスフィルタ  $G_f(s) = \frac{1}{1+0.01s}$  を表している．

ビルドを行い，エラーがないことを確認した後，print コマンドにて “ex.ident.pdf” という名前の pdf ファイルを生成し，デスクトップ上の “bcpdfcrop-multi.bat” により余白を取り除く．

(2) Scope をダブルクリックして開き，リアルタイムで角度データを観測できるようにする．レンジは付録 A.2 の図 A.5 に合わせる．

(3) オシロスコープを Universal Power Module の “S1” と “GND” に接続し，センサ電圧が 0[V] となる位置にアームを動かす．

(4) “ex.ident.slx” を実行する．実行終了後，

```
>> save ident_data t y dy dyf
```

と入力し，mat ファイル “ident\_data.mat” にデータを保存する．

(5) 配布する M ファイル

```
ident_para.m
```

を実行することによって，実験結果のグラフを描くとともに定常値  $y_\infty = K$  および  $y_\infty = K$  の 63.2% に至るまでの時間  $T$  を抽出する．この結果を基に未知パラメータ  $a$ ， $b$  を定め，表 5.1 に記入せよ．

#### 5.1.1 実験結果

以下に結果を示す．

表 5.1 同定されたパラメータの値

ゲイン $K$	時定数 $T$	$a$	$b$
1.46	0.046	21.7	31.8

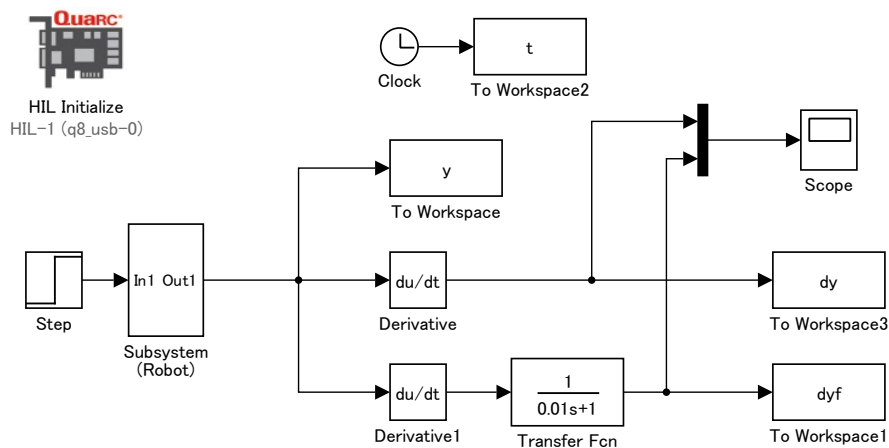


図 5.1 ex\_ident

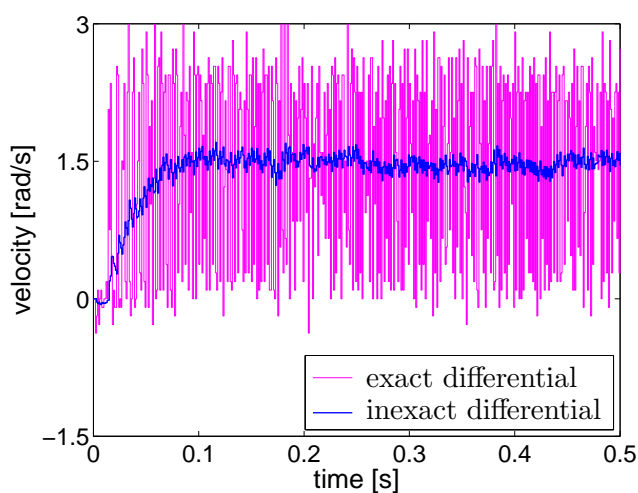


図 5.2 figure\_lowpass

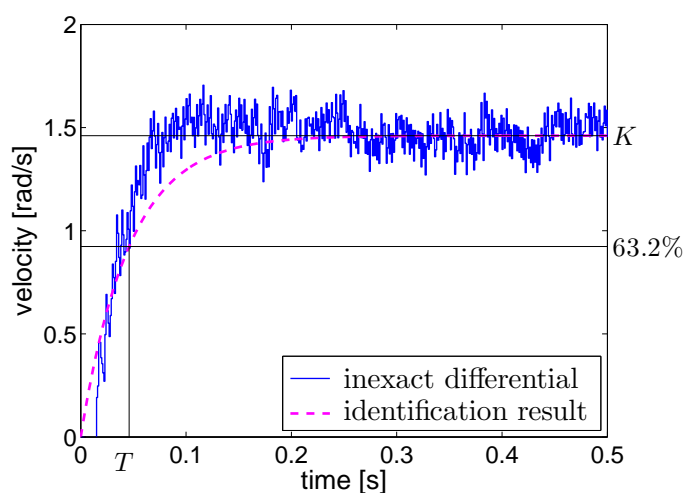


図 5.3 figure\_ident

### 5.1.2 実験考察

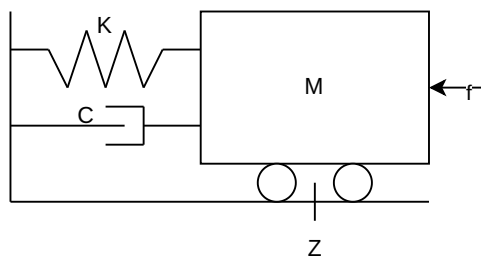


図 5.4 マスバネダンパ系

実験により得られたパラメータは，表に示すように，ゲイン  $K = 1.46$ ，時定数  $T = 0.046$ ，および未知パラメータ  $a = 21.7$ ， $b = 31.8$  である．図 5.2 の結果より，ローパスフィルタを適用していない場合（exact differential）は，観測雑音により信号に大きな振動が生じているのが確認できる．一方，ローパスフィルタを適用した場合（inexact differential）では，結果図 5.3 から得られたゲイン  $K = 1.46$  は，システムが入力ステップ応答に対して約 63.2% に達する値であり，実験により同定された時定数  $T = 0.046$  は，システムがこの値に収束するまでの応答時間を表して

いる．さらに，カットオフ周波数に関連する時定数  $T = 0.046$  が適切に設定されたことで，ローパスフィルタが観測雑音を効果的に減衰しながらも，システムの応答性を損なわないようなバランスがとられている．フィルタのカットオフ周波数をさらに低く設定すると，高周波成分の除去効果は増加するが，システムの応答が遅延し，出力信号が目標値に到達するまでの時間が長くなると考えられる．が除去され，滑らかな信号が得られている．このことから，ローパスフィルタによって高周波雑音が減衰され，観測値が安定化していることがわかる．また，図 5.3 に示される識別結果から，得られたゲイン  $K = 1.46$  は，システムが入力ステップ応答に対して約 63.2% に達する値であり，実験により同定された時定数  $T = 0.046$  は，システムがこの値に収束するまでの応答時間を表している．さらに，カットオフ周波数に関連する時定数  $T = 0.046$  が適切に設定されたことで，ローパスフィルタが観測雑音を効果的に減衰しながらも，システムの応答性を損なわないようなバランスがとられている．フィルタのカットオフ周波数をさらに低く設定すると，高周波成分の除去効果は増加するが，システムの応答が遅延し，出力信号が目標値に到達するまでの時間が長くなると考えられる．

## 5.2 P-D 制御

1. (4.11) 式にしたがって P-D コントローラ (4.13) 式のパラメータを定め，表 5.2 を完成させよ．
2. 図 5.2 の Simulink モデル "ex.pdcont.slx" を作成し，ディレクトリ D:#student.5S#group01#pdcont に保存する．ただし，"Subsystem (P-D Controller)" の内部は各自作成する（角速度  $\dot{\theta}$  はローパスフィルタ  $G_f(s) = \frac{1}{1+0.01s}$  に通す）．
3. 次に，表 5.2 に示す比例ゲイン  $k_P$ ，微分ゲイン  $k_D$  を，コマンドウィンドウで

$$>> k_P = ***; k_D = ***; \quad (5.1)$$

と入力（\*\*\* には表 5.2 の数字を入力する）した後，ビルドを行い，エラーがないことを確認する．また，"ex.pdcont.pdf" という名前の pdf ファイルを生成し，デスクトップ上の "bcpdfcrop-multi.bat" により余白を取り除く．

4. Scope と Scope1 を開き，リアルタイムで角度と操作量を観測できるようにする．レンズは付録 A.2 の図 A.6 に合わせる．
5. センサ電圧が 0 [V] となる位置にアームを動かし，"ex.pdcont.slx" を実行する．実行終了後，コマンドウィンドウで

$$>> savepdcont\_data1tuyk_Pk_D \quad (5.2)$$

と入力し，データを

- "pdcont.data1.mat" ( $\omega_M = 15, \zeta_M = 0.3$ )
- "pdcont.data2.mat" ( $\omega_M = 15, \zeta_M = 0.7$ )
- "pdcont.data3.mat" ( $\omega_M = 15, \zeta_M = 1$ )

という名前の mat ファイルでディレクトリ D:#student.5S#group01#pdcont に保存する．最後に，配布する M ファイル

$$\cdot \text{"autoplot\_pdcont.m"} \quad (5.3)$$

を実行することによって，MATLAB 上でグラフを作成する．グラフの pdf ファイルは自動的に生成される．

6. 目標値を 0 に設定 (Step の最終値を "0" に設定) し, また, 終了時間を "inf" に設定する.  $\omega_M, \zeta_M$  を表 5.2 のように与えた後, アームを平らに 45 [deg] 程度動かしてから手を放すとき, アームがどのように応答するか調べよ.

### 5.2.1 実験結果

以下に結果を示す.

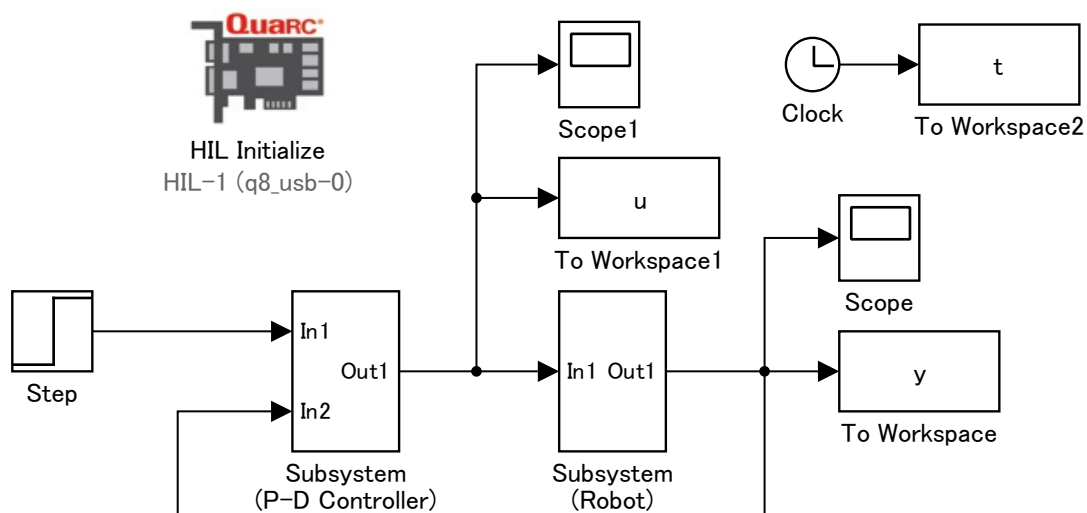


図 5.5 ex\_pdcont

表 5.2 P-D コントローラ

固有角周波数 $\omega_M$	減衰係数 $\zeta_M$	比例ゲイン $k_P$	微分ゲイン $k_D$
15	0.3	7.075	-0.399
15	0.7	7.075	-0.022
15	1	7.075	0.261

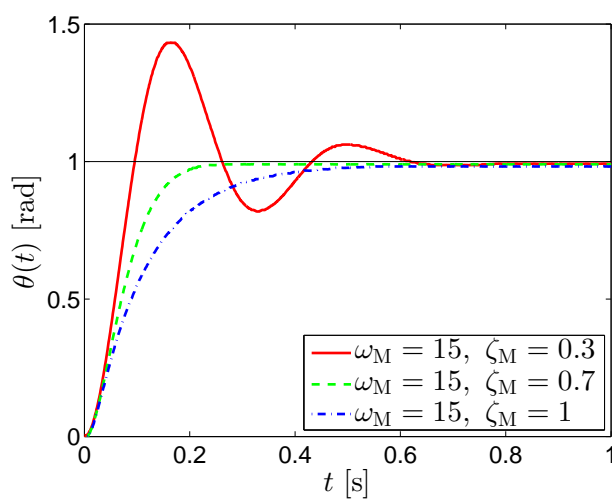


図 5.6 ex\_pdcont

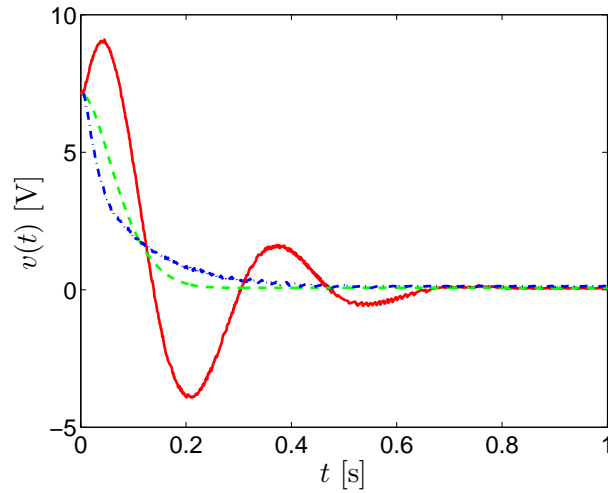


図 5.7 ex\_pdcont

(5) 減衰係数が大きくなるに連れ、即応性がなくなり、ゆっくりになりダンパの動きが見られた。

### 5.2.2 実験考察

微分動作は、制御対象の誤差の時間変化を考慮し、制御入力を調整するものである。これにより応答の速度が向上し、オーバーシュートの抑制や立ち上がり時間の短縮が期待できる。微分動作を加えることで、過渡応答が改善され、立ち上がり時の急激な変動が抑制されと考えられる。

次に、規範モデル式 (4.10) に基づくステップ応答と実験結果における定常偏差の違いについて述べる。理論的な規範モデルは、理想的な条件での応答を示しているが、PD 制御である場合、定常偏差が残る可能性がある。このことは次の最終値の定理からも確認できる。

$$\lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sE(s)$$

積分動作がない場合、 $e(t)$  はゼロに収束せず、定常偏差が残存することになる。次に、減衰係数を変化させたときのアームの挙動について、マスバネダンパ系の視点から考察する。マスバネダンパ系の運動方程式は以下のように表される。

$$m\ddot{z} + c\dot{z} + kz = f$$

これを  $f$  について解くと、

$$f = m\ddot{z} + c\dot{z} + kz$$

となる。ここで、 $m$  は質量、 $c$  はダンパ係数（減衰係数に関係）、 $k$  はバネ定数（固有角周波数に関係）である。この系の伝達関数  $P(s)$  は、出力を  $z(s)$ 、入力を  $f(s)$  として

$$P(s) = \frac{z(s)}{f(s)} = \frac{1}{ms^2 + cs + k}$$

と表される。さらに、この系を 2 次遅れ系の標準形に変換するため、分母の式を整理すると

$$P(s) = \frac{\frac{1}{m}}{s^2 + \frac{c}{m}s + \frac{k}{m}}$$

となる。ここで、2 次遅れ系の標準形

$$P(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

と比較すると、固有角周波数  $\omega_n$  と減衰係数  $\zeta$  はそれぞれ以下のように表されることがわかる。

$$\omega_n = \sqrt{\frac{k}{m}}, \quad \zeta = \frac{c}{2\sqrt{mk}}$$

これにより、バネ定数  $k$  を変化させると固有角周波数  $\omega_n$  が変化し、ダンパ係数  $c$  を変化させると減衰係数  $\zeta$  が変化することがわかる。

### 5.3 I-P 制御

1. (4.18) 式にしたがって I-P コントローラ (4.14) 式のパラメータを定め、表 5.3 を完成する。
2. 図 5.3 の Simulink モデル “ex\_ipcont.slx” を作成し、ディレクトリ D:#student\_5S#group01#ipcont に保存する。ただし、“Subsystem (I-P Controller)” の内容は各自で考える。
3. 表 5.3 の比例ゲイン  $k_P$ 、積分ゲイン  $k_I$  をコマンドウィンドウで

```
>> kP = ***; kI = ***;
```

と入力し (\*\*\*)には表 5.3 の数値を入力する)、ビルドを行い、エラーがないことを確認する。また、“ex\_ipcont.pdf” という名前の pdf ファイルを生成し、デスクトップ上の “bcpdfcrop-multi.bat” により余白を取り除く。

4. Scope と Scope1 を開き、リアルタイムで角度と操作量を観測できるようにする。レンズは付録 A.2 の図 A.7 に合わせる。
5. センサ電圧が 0[V] となる位置にアームを動かし、“ex\_ipcont.slx” を実行する。実行終了後、コマンドウィンドウで

```
>> save ipcont_data1 t u y kP kI
```

と入力し、データを

- “ipcont\_data1.mat” ( $\omega_M = 15, \zeta_M = 0.3$ )
- “ipcont\_data2.mat” ( $\omega_M = 15, \zeta_M = 0.7$ )
- “ipcont\_data3.mat” ( $\omega_M = 15, \zeta_M = 1$ )

という名前の mat ファイルでディレクトリ D:#student\_5S#group01#ipcont に保存する。最後に、配布する M ファイル

```
autoplot-ipcont.m
```

を実行することによって、MATLAB 上でグラフを作成する。グラフの pdf ファイルは自動的に生成される。

#### 5.3.1 実験結果

#### 5.3.2 実験考察

I-P 制御は、偏差に積分動作を適用した量に基づいて、目標値の変化を滑らかにし、その後に比例要素を作用させた操作量を出力します。この制御構造により、I-P 制御は目標値の急激な変化を緩和し、システムに与える影響を抑制する働きを持つ。これは図 5.9 から読み取れる。P-D 制御では積分動作がないため、外乱や負荷変動によって生じ

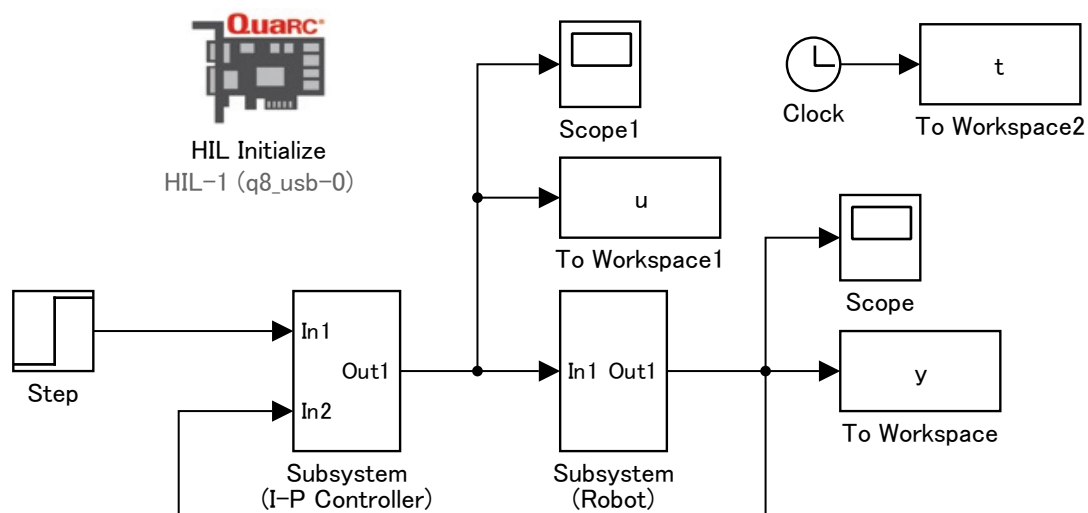


図 5.8 ex\_ipcont

表 5.3 I-P コントローラ

固有角周波数 $\omega_M$	減衰係数 $\zeta_M$	比例ゲイン $k_P$	積分ゲイン $k_I$
15	0.3	6.14150	153.53773
15	0.7	14.33018	153.53773
15	1	20.47169	153.53773

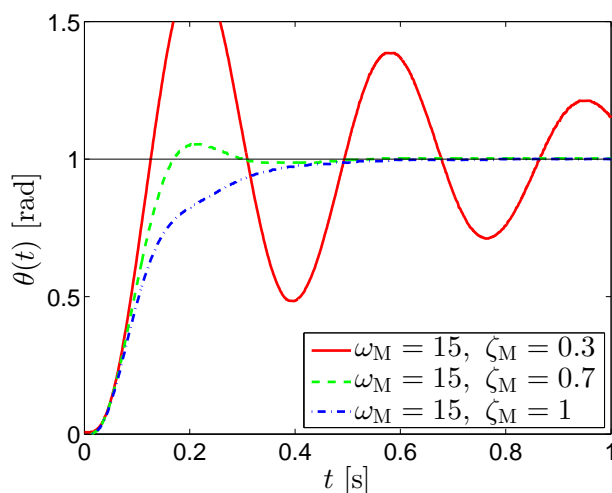


図 5.9 ex\_ipcont

た定常偏差がそのまま残ってしまう一方，I-P 制御には積分動作が含まれているため，定常偏差が時間とともに収束し，最終的には目標値に到達する．また，動作が規範モデルとことなる理由は，パラメータ調整に部分的モデルマッチング法が用いられているためである．部分的モデルマッチング法は，特定の周波数帯域において理想モデルに近似するように設計されるが，全体の応答を完全に再現するものではない．したがって，高周波数成分や外乱の影響が含まれる実験結果においては，理想モデルと異なる結果が出たと考えられる．

## 5.4 I-PD 制御

1. (4.22) 式にしたがって I-PD コントローラ (4.23) 式のパラメータを定め，表 5.4 を完成せよ．



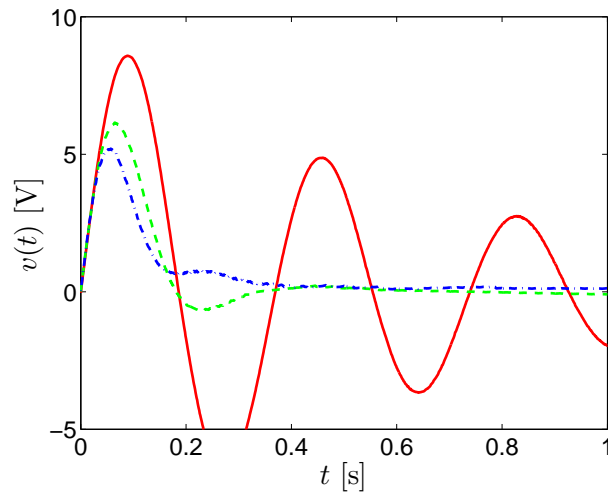


図 5.10 ex\_ipcont

- 次に，表 5.4 の比例ゲイン  $k_P$ ，積分ゲイン  $k_I$ ，微分ゲイン  $k_D$  を，コマンドウィンドウで次のように入力します．

```
kP = ***; kI = ***; kD = ***;
```

- Scope と Scope1 を開き，リアルタイムで角度と操作量を観測できるようにする．レジメは付録 A.2 の 図 A.8 に合わせる．
- センサ電圧が 0 [V] となる位置にアームを動かし，“ex\_ipdcont.slx” を実行する．実行終了後，コマンドウィンドウで

```
>> save ipdcont_data1 t u y kP kI kD
```

と入力し，データを以下のファイルとして保存します．

- “ipdcont\_data1.mat” ( $\omega_M = 20, \alpha_{M1} = 2, \alpha_{M2} = 2$ ) - “ipdcont\_data2.mat” ( $\omega_M = 20, \alpha_{M1} = 3, \alpha_{M2} = 3$ ) - “ipdcont\_data3.mat” ( $\omega_M = 20, \alpha_{M1} = 2.15, \alpha_{M2} = 1.75$ )

これらの mat ファイルをディレクトリ D:

\$student\_55

group01

ipdcont に保存します．最後に，以下の M ファイルを実行します．

- “autoplots\_ipdcont.m”

この実行によって，MATLAB 上でグラフを作成し，グラフの pdf ファイルは自動的に生成されます．

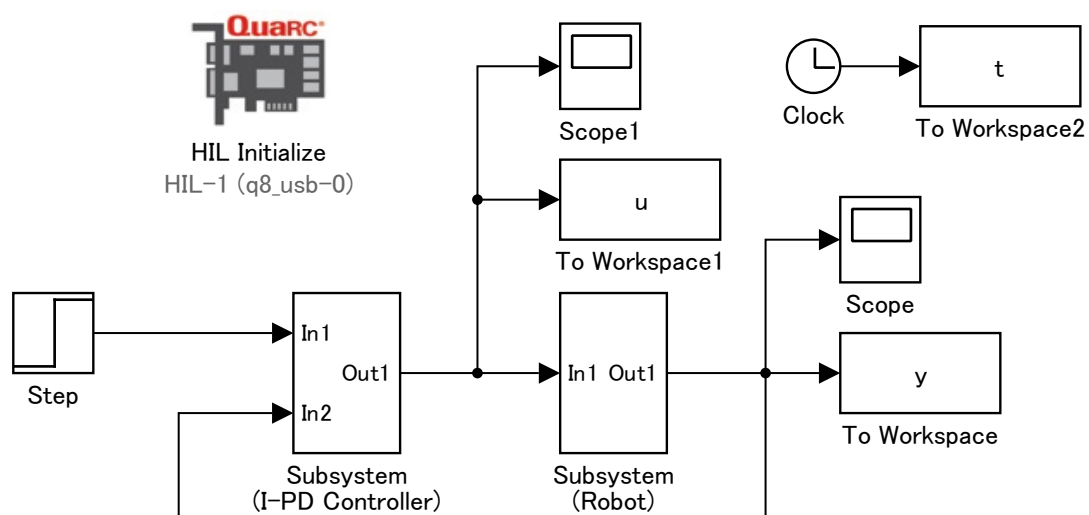


図 5.11 ex\_ipdcont

#### 5.4.1 実験結果

表 5.4 I-PD コントローラ

$\omega_M$	$\alpha_{M1}$	$\alpha_{M2}$	比例ゲイン $k_P$	積分ゲイン $k_I$	微分ゲイン $k_D$
20	2	2	25.157	251.572	0.575
20	3	3	37.736	251.572	1.204
20	2.15	1.75	27.044	251.572	0.418

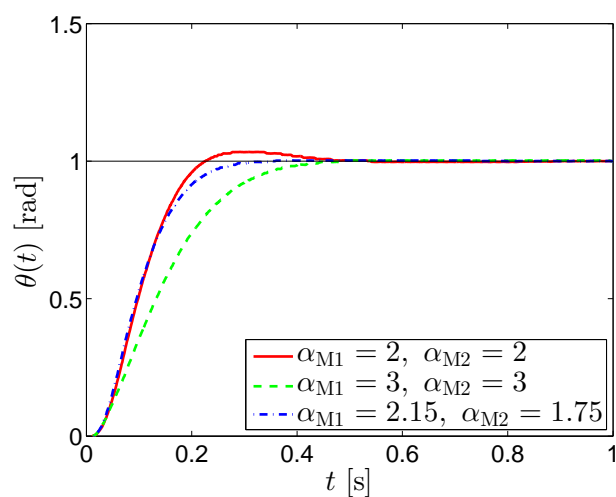


図 5.12 figure\_ipdcont\_angle

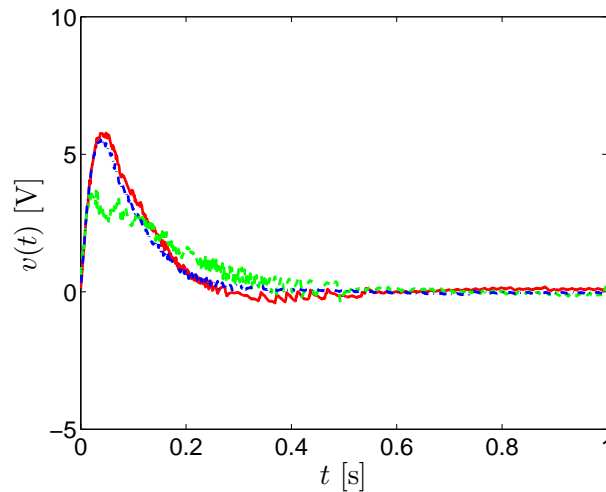


図 5.13 figure\_ipdcont\_volt

#### 5.4.2 実験考察

I-PD 制御は，比例項 ( $k_P$ ) と積分項 ( $k_I$ )，および微分項 ( $k_D$ ) が組み合わせることで，応答速度と定常精度が良くなっている．図 5.12 より，角度応答では目標値への到達が速く，また定常状態での偏差がほとんどないことがわかる．これは積分項によって定常偏差が補正されるため，P-D 制御と比較して定常偏差がほぼなくなっているためだと考えられる．また，I-P 制御と比較しても過渡応答が速く，振動が抑えられている．

一方で，図 5.13 に示される電圧応答においては，微分ゲインが大きいときに振動が見られる．これは微分項が高周波成分やノイズに敏感であるためであり，急激な変動に対して過大な操作量が発生しやすいためである．そのため，一般的に微分動作の前にローパスフィルターなどのフィルターを通すことが一般的である．

## 6 調査課題

### 1

ポテンショメータは，可変抵抗器の一種であり，主に位置や角度を測定するためのセンサとして使用される．その動作原理は，接点（スライダ）を抵抗体上で移動させることによって，抵抗値を変化させ，その変化によって位置や角度を検出する．抵抗体の両端に電圧を印加し，スライダの位置に応じた電圧が出力されるため，位置情報が得られる．

$$\text{出力電圧} = \frac{\text{スライダの位置}}{\text{全抵抗体の長さ}} \times \text{印加電圧}$$

また，ポテンショメータはアブソリュート位置センサであるため，電源のオン・オフに関係なく位置情報を保持できる．しかし，スライダと抵抗体の接触部分が摩耗しやすいため，耐久性や寿命が求められる用途には不向きである．

### 2

A/D 変換は，アナログ信号をデジタル信号に変換するプロセスである．アナログ信号が時間的および値の上で連続的であるのに対し，デジタル信号は離散的なサンプル点と有限のビットで表現される．まず，サンプリングにより，アナログ信号の時間的に連続したデータを一定の時間間隔で分割して離散化する．次に，量子化を行う．量子化は，サンプリングによって得られた信号の値を有限のビット数で表現できるように離散化するプロセスである．量子化の

精度はビット数に依存し、ビット数が多いほどより詳細なアナログ信号の情報を保持できる。最後に、符号化が行われ、量子化された値をデジタル形式として記録する。D/A 変換は、デジタル信号を再びアナログ信号に戻すプロセスである。デジタル信号は不連続なデータで構成されているため、変換によって生成されるアナログ信号は理想的な連続信号とは異なる場合があるが、近似的な再現が可能である。

### 3

レンジが  $\pm 5 \text{ V}$ 、分解能が 12 ビットの場合、アナログ出力の電圧は、

$$1 \text{ LSB} = \frac{10 \text{ V}}{4096} = \frac{10}{4096} \approx 0.00244 \text{ V}$$

次に、0x0C00 のデジタル値をアナログ値に変換する。0x0C00 は 10 進数に直すと 3072 であるため、

$$\text{アナログ値} = (3072) \times 0.00244 \text{ V} - 5 \text{ V} = 2.5 \text{ V}$$

よって 0x0C00 を D/A 変換するとアナログ値は 2.5[V] となる

### 4

式 4.9 は以下の通りである。

$$T(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}, \quad \omega_n = \sqrt{bk_P}, \quad \zeta = \frac{a + bk_D}{2\omega_n}$$

システムの運動方程式に比例微分 (PD) 制御を適用すると、次の式が得られる。

$$m \frac{d^2 x}{dt^2} + c \frac{dx}{dt} + kx = k_P e + k_D \frac{de}{dt}$$

ここで、簡略化のために  $b = \frac{1}{m}$  および  $a = \frac{c}{m}$  とおくと、上記の式は次のように表される。

$$\frac{d^2 x}{dt^2} + a \frac{dx}{dt} + bx = bk_P e + bk_D \frac{de}{dt}$$

ラプラス変換を適用して、ラプラス変数  $s$  を用いて表現する。これにより次の式が得られる。

$$(s^2 + as + b)X(s) = bk_P E(s) + bk_D sE(s)$$

ここで、出力と入力の比  $T(s) = \frac{X(s)}{E(s)}$  を整理すると、以下の伝達関数が得られる。

$$T(s) = \frac{X(s)}{E(s)} = \frac{bk_P + bk_D s}{s^2 + as + b}$$

ここで、自然周波数  $\omega_n$  と減衰係数  $\zeta$  を次のように定義する。

$$\omega_n = \sqrt{bk_P}$$

$$\zeta = \frac{a + bk_D}{2\omega_n}$$

これらを代入し、伝達関数を整理すると次の式が得られる。

$$T(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

式 4.20 は次のように与えられる。

$$T(s) = \frac{bk_I}{s^3 + (a + bk_D)s^2 + bk_Ps + bk_I}$$

PID 制御を適用したシステムの運動方程式は，次のように与えられる．

$$m \frac{d^2x}{dt^2} + c \frac{dx}{dt} + kx = k_P e + k_D \frac{de}{dt} + k_I \int e dt$$

ここで，簡略化のために  $b = \frac{1}{m}$  および  $a = \frac{c}{m}$  とおくと，上記の式は次のように書き直される．

$$\frac{d^2x}{dt^2} + a \frac{dx}{dt} + bx = bk_P e + bk_D \frac{de}{dt} + bk_I \int e dt$$

ラプラス変換を適用して，ラプラス変数  $s$  を用いて表現すると，積分項  $\int e dt$  のラプラス変換は  $\frac{E(s)}{s}$  となり，以下の式が得られる．

$$(s^2 + as + b)X(s) = bk_P E(s) + bk_D sE(s) + \frac{bk_I}{s} E(s)$$

次に，この式を  $X(s)$  と  $E(s)$  の関係として整理する．右辺を  $E(s)$  でくくり出すと，以下のようになる．

$$(s^2 + as + b)X(s) = \left( bk_P + bk_D s + \frac{bk_I}{s} \right) E(s)$$

ここで， $T(s) = \frac{X(s)}{E(s)}$  として，両辺を  $E(s)$  で割ると次のようになる．

$$T(s) = \frac{X(s)}{E(s)} = \frac{bk_I}{s^3 + (a + bk_D)s^2 + bk_Ps + bk_I}$$

## 参考文献

- 1) 川田 (編著), 東, 市原, 浦久保, 大塚, 甲斐, 國松, 澤田, 永原, 南 : 倒立振子で学ぶ制御工学, 森北出版 (2017)
- 2) 川谷, 外川 : 現代制御理論を使った倒立振子の実験 [1]–[3], トランジスタ技術, No. 5, pp. 315–322, No. 6, pp. 367–373, No. 7, pp. 363–370 (1993)
- 3) 「初学者のための図解でわかる制御工学」特集号 (基礎編), システム/制御/情報, Vol. 56, No. 4 (2012)
- 4) 大山 : たしなみながら学ぶセンサ制御 - フィードバック制御から現代制御理論の応用まで, インターフェース, No. 9, pp. 80–119 (1993)
- 5) 大山, 工藤, 岡本, 藤沢 : 現代制御理論に基づいたデジタル制御系の設計法, インターフェース, No. 12, pp. 215–250 (1986)
- 6) 川田 : MATLAB/Simulink による制御工学入門, 森北出版 (2020)
- 7) 川田 : MATLAB/Simulink による現代制御入門, 森北出版 (2011)
- 8) 川田 : MATLAB/Simulink と実験で学ぶ制御工学 - PID 制御から現代制御まで -, TechShare (2013)
- 9) RS コンポーネンツ 「ポテンシオメーター完全ガイド～多彩な使い道と種類を徹底解説」  
[https://jp.rs-online.com/web/content/discovery/ideas-and-advice/potentiometers-guide?srsId=AfmBOopoFLKTmQ70MID\\_a9scNLFYl4ibGXCN-zv51aO\\_NanDwfYN5FM](https://jp.rs-online.com/web/content/discovery/ideas-and-advice/potentiometers-guide?srsId=AfmBOopoFLKTmQ70MID_a9scNLFYl4ibGXCN-zv51aO_NanDwfYN5FM) (2024 年 10 月 27 日取得)
- 10) CONTEC 「アナログ入出力の基礎知識・用語解説」  
<https://www.contec.com/jp/support/basic-knowledge/daq-control/analog-io/> (2024 年 10 月 27 日取得)