

(2) ロボットアームの角度制御

目次

1 実験目的	2-1
2 ロボットアーム実験装置の概要と制御目的	2-1
2.1 ロボットアーム実験装置	2-1
2.2 アクチュエータと D/A 変換	2-1
2.3 センサと A/D 変換	2-2
2.4 制御目的と P 制御	2-3
2.5 PI 制御	2-3
3 実験 1 — A/D, D/A 変換の動作確認と P, PI 制御 —	2-4
3.1 実験装置のセッティング	2-4
3.2 D/A 変換とアクチュエータの動作確認	2-4
3.3 A/D 変換とセンサの動作確認	2-6
3.4 ロボットアームの表現	2-6
3.5 P 制御	2-8
3.6 積分動作の効果 (PI 制御)	2-9
3.7 L ^A T _E X を利用した実験結果の印刷	2-10
4 モデルに基づく改良型 PID 制御	2-11
4.1 ロボットアームのモデリング	2-11
4.1.1 運動方程式と伝達関数	2-11
4.1.2 パラメータ同定	2-11
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	2-12
4.2.1 P-D 制御 (微分先行型 PD 制御)	2-12
4.2.2 I-P 制御 (比例先行型 PI 制御)	2-12
4.2.3 I-PD 制御 (比例・微分先行型 PID 制御)	2-13
5 実験 2 — 改良型 PID 制御の比較 —	2-15
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	2-15
5.2 P-D 制御	2-16
5.3 I-P 制御	2-17
5.4 I-PD 制御	2-18
5.5 L ^A T _E X を利用した実験結果の印刷	2-20
6 検討課題	2-20
参考文献	2-20
付録	2-21
A.1 実験 1 の実験結果	2-21
A.2 実験 2 の実験結果	2-22

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)

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, 平面上を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ギアを介して 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により計算された指令電圧は, I/O ボード “Q8-USB” で D/A 変換された後, アンプを介して DC モータに入力される。

アームの回転角はギアを介して角度センサであるポテンショメータにより電圧値として検出され, I/O ボード “Q8-USB” で A/D 変換された後, パソコンに取り込まれている。図 2.2 に制御対象の信号の流れを示す。

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量であるため, D/A 変換によってアナログ量 (図 2.2 に示すように, サンプリング周期 h ごとに一定値のアナログ量) に変換してからアンプに加える必要がある。本実験装置の I/O ボード “Q8-USB” 上の D/A 変換部はレンジ: ± 10 [V], 分解能: 16 ビットであり, 約 0.30518 [mV] ($= \{+10 - (-10)\} / 2^{16}$ [V]) の刻み幅である。

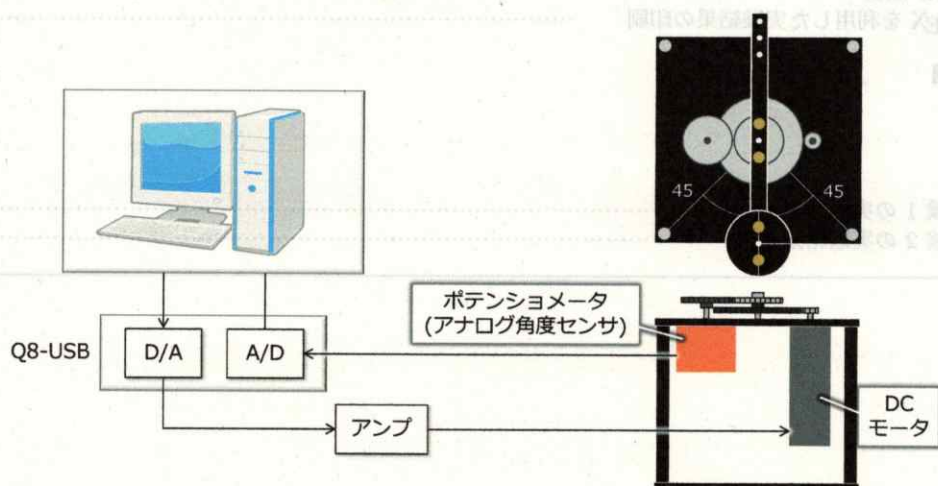


図 2.1 ロボットアーム実験装置

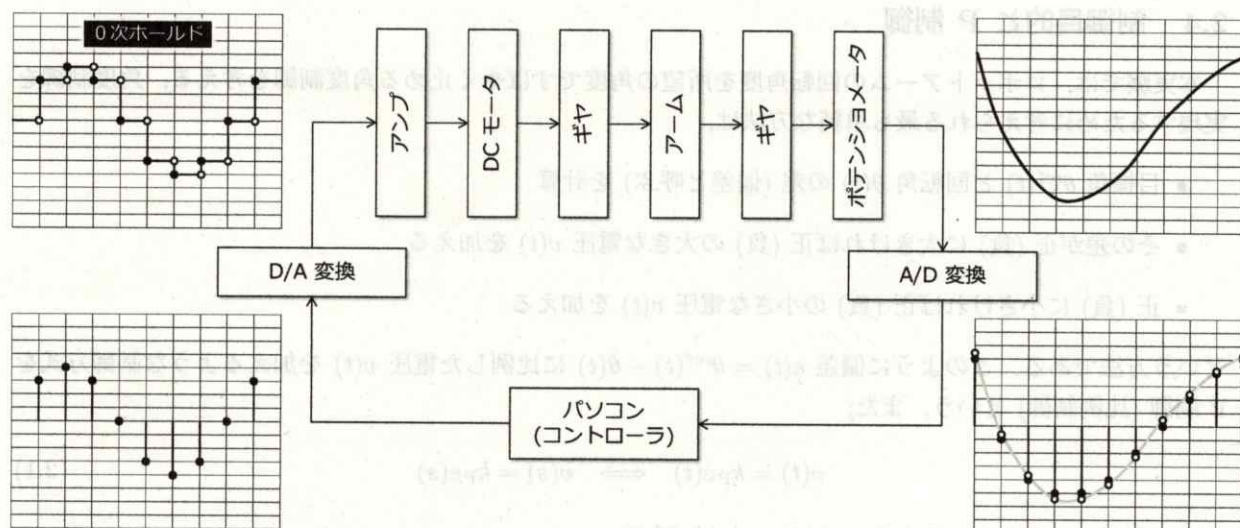


図 2.2 制御対象のブロック線図

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ヨメータが用いられている。ポテンシヨ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 h [s] でデジタル量に変換する必要がある (図 2.3 参照)。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: ± 5 [V] または ± 10 [V]、分解能: 16 ビットである。本実験では、レンジを ± 5 [V] に設定しているため、約 0.15259 [mV] ($= \{+5 - (-5)\} / 2^{16}$ [V]) の刻み幅 (量子化サイズ) で量子化することになる。

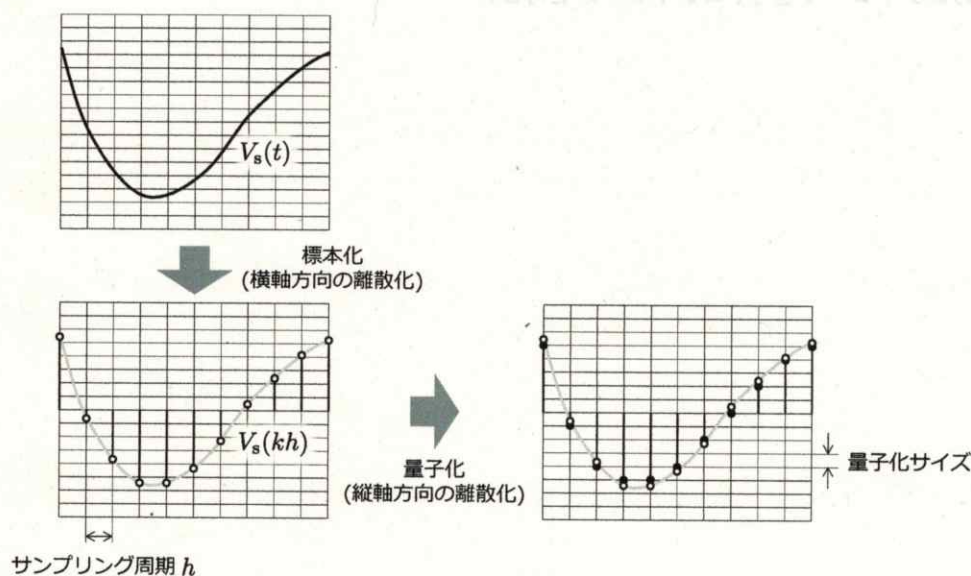


図 2.3 A/D 変換における標本化と量子化

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式を 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 2.4 P 制御 (比例制御)

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。



図 2.3 A/D 変換における積分動作の効果

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。

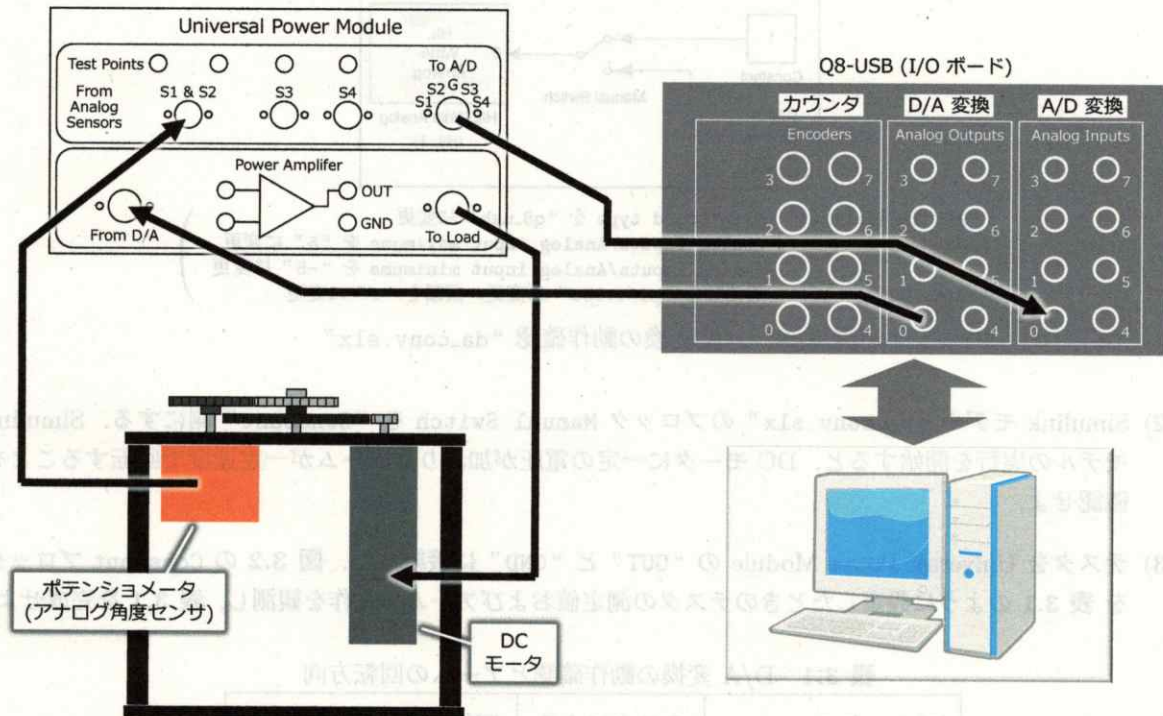


図 3.1 ロボットアーム実験装置の結線

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 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^{*1}．Simulink モデルをビルドし，エラーがないことを確認した後，

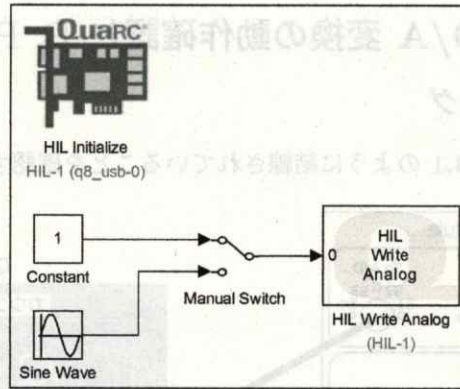
```
>> print -s -dpdf da_conv.pdf  ⏏ ※結果確認 (3.7 節) で使うので必須
>> print -s -dbitmap da_conv.bmp  ⏏ ※ Word でレポートを書く場合
```

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】)．

^{*1}MATLAB コマンドウィンドウで

```
>> cd D:\student.5S\group01\da  ⏏
と入力し，カレントディレクトリを変更してから実験を行うこと．次の実験に移る時は
>> cd ../ad  ⏏
などと入力する．ドット 1 つ “.” で現在位置，ドット 2 つ “..” で 1 つ上のディレクトリを指す．以後各実験ごとにカレントディレクトリを変更すること．
```



- (HIL Initialize: Main/Board type を “q8_usb” に変更
 Analog Inputs/Analog input maximums を “5” に変更
 Analog Inputs/Analog input minimums を “-5” に変更
 Sine Wave: 周波数を “pi/0.008” に変更, 振幅を “5” に変更)

図 3.2 D/A 変換の動作確認 “da_conv.slx”

- (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タ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.

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]	テスタによる 測定値 [V]	回転方向 (時計回り, 反時計回り, 静止)
-5		
-2.5		
-1		
-0.5		
0		
0.5		
1		
2.5		
5		

- (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).

3.3 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。

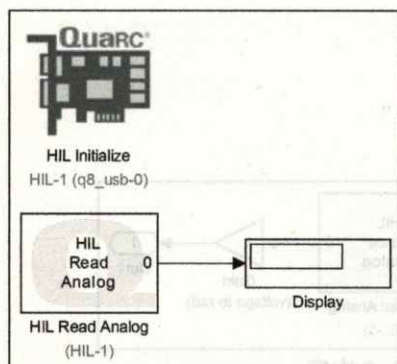


表 3.2 A/D 変換の動作確認 (角度: 反時計回りが正)

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		
0	—	—
-45		

図 3.3 A/D 変換の動作確認 “ad_conv.slx”

- (2) テスタを Universal Power Module の “S1” と “GND” に接続し、アームを ± 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 [V] あたりの角度変位を求めよ (符号に注意)。

センサ電圧 1 [V] あたりの角度変位 = [deg] = [rad]

- (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。

センサ電圧: 最小値 = [V], 最大値 = [V]

3.4 ロボットアームの表現

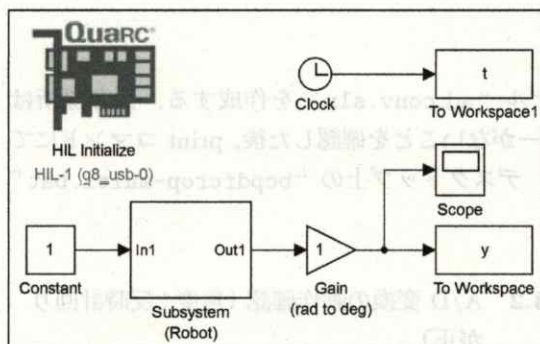
- (1) 以下の動作をするような 図 3.4 の Simulink モデルを完成させる。

- アンプに加える電圧が $-10 \sim 10$ [V] の範囲を越えないようにする。
- 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 [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。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。

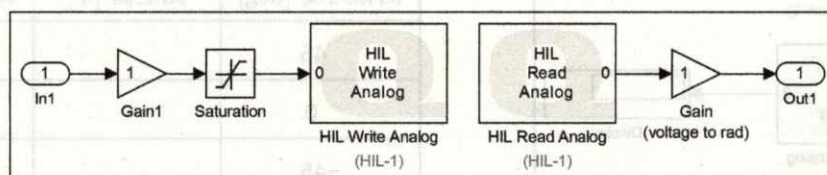
- (2) 3.2 節 (4) で外したケーブルを接続する,
- (3) オシロスコープを Universal Power Module の “S1” と “GND” に接続し、センサ電圧が 0 [V] になるよう、アームを動かす^{*2}。

^{*2}デジタルテストでは 0 [V] を取りにくいので、オシロスコープにて 0 [V] を取る。以降この方法で電気的な 0 度を開始位置とする。



(To Workspace: 変数名を “y”, 保存形式を “配列” に変更
To Workspace1: 変数名を “t”, 保存形式を “配列” に変更)

(a) “ad_da_conv.slx”



(b) “Subsystem (Robot)” の内容

図 3.4 ロボットアームの表現 “ad_da_conv.slx”

(4) 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。

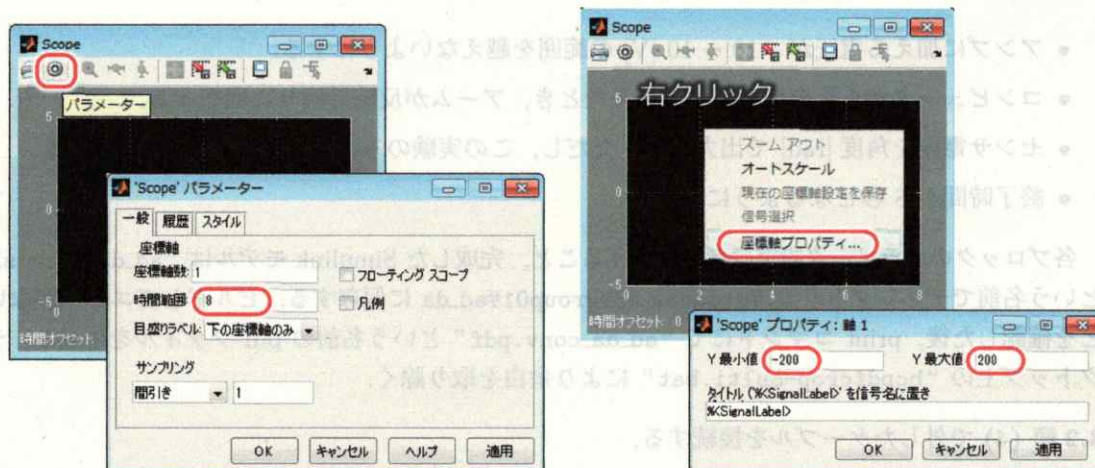
(5) すべての準備が整ってから “ad_da_conv.slx” を実行する。実行終了後、コマンドウィンドウで

```
>> 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

- “autoplots_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。



時間範囲を “8” に変更

Y 最小値を “-200”, Y 最大値を “200” に変更

図 3.5 Scope の設定

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

$$v(t) = k_P(\theta^{\text{ref}}(t) - \theta(t)) \quad (\theta^{\text{ref}}(t) : \theta(t) \text{ の目標値}) \quad (3.1)$$

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。

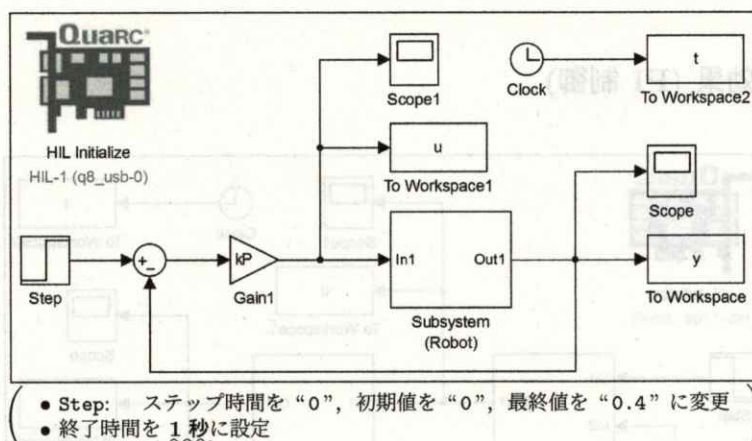


図 3.6 P 制御の実機実験を行うための Simulink モデル “ex_pcont.slx”

- (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。レンジは 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^{*3}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”

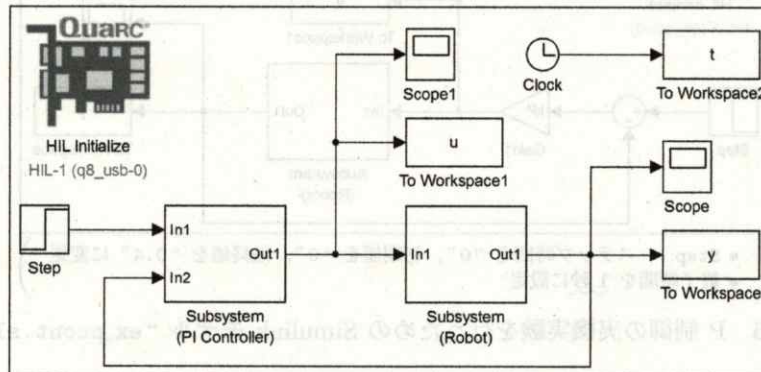
を実行することによって、MATLAB 上でグラフを作成する。グラフの pdf ファイルは自動的に生成される。

^{*3}変数の値を変更しただけなら再ビルドの必要はない。以降同様。

考察のポイント

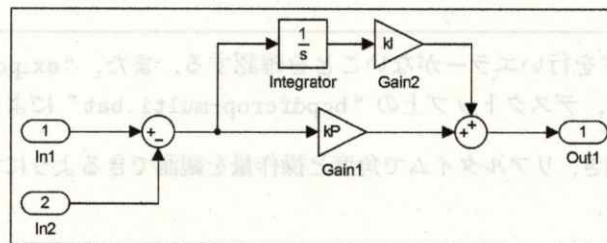
- k_p を大きくするにしたがって、過渡特性 (オーバーシュート, 立ち上がり時間) や操作量 (入力電圧) の大きさがどのように変わっているかを考察せよ。その理由も示すこと。
- とくに定常特性 (定常偏差) においては、偏差が残る理由を理論 (最終値の定理) と実機の特性を併せて考察すること。

3.6 積分動作の効果 (PI 制御)



- Step: ステップ時間を "0", 初期値を "0", 最終値を "0.4" に変更
- 終了時間を 4 秒 に設定

(a) "ex_picont.slx"



(b) "Subsystem (PI Controller)"

図 3.7 PI 制御の実機実験を行うための Simulink モデル "ex_picont.slx"

- (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。レンジは 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 $k_I = 6, k_I = 12$ と設定して “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よ。

考察のポイント

- 積分動作により定常偏差がどのようになっているかを考察せよ。また、その理由も示すこと。
- 積分動作によりアームに力を加えたときの振る舞いがどのようになるかを考察せよ。また、その理由も示すこと。

3.7 L^AT_EX を利用した実験結果の印刷

ディレクトリ D:\student_5S\group01\tex1 にある “print1.tex” をタイプセット (コンパイル) し、pdf ファイルをプリントアウトせよ。



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

図 4.1 に示す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·m²]：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で決まるパラメータである。

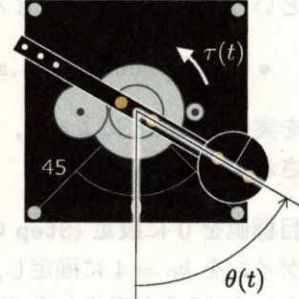


図 4.1 ロボットアーム

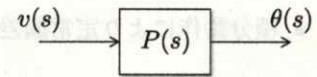


図 4.2 ロボットアームのブロック線図

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

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

となる。(4.3) 式をブロック線図で表すと、図 4.2 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。

$y_d(t) = \dot{\theta}(t)$ とすると (4.2) 式は

$$\dot{y}_d(t) = -ay_d(t) + bv(t) \quad (4.4)$$

となるから、 $v(s) = \mathcal{L}[v(t)]$ から $y_d(s) = \mathcal{L}[y_d(t)]$ 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

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

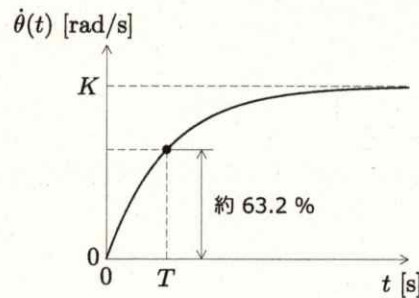


図 4.3 $y_d(s) = P_d(s)v(s)$ の単位ステップ応答

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、図 4.3 から時定数 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)

$$v(t) = k_P e(t) - k_D \frac{d\theta(t)}{dt} \iff v(s) = k_P e(s) - k_D s \theta(s) \quad (4.8)$$

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

$$T(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}, \quad \omega_n = \sqrt{bk_P}, \quad \zeta = \frac{a + bk_D}{2\omega_n} \quad (4.9)$$

となる. したがって, (4.9) 式を 2 次遅れ系の規範モデル

$$y_M(s) = T_M(s)\theta^{\text{ref}}(s), \quad T_M(s) = \frac{\omega_M^2}{s^2 + 2\zeta_M\omega_M s + \omega_M^2} \quad (4.10)$$

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

$$k_P = \frac{\omega_M^2}{b}, \quad k_D = \frac{2\zeta_M\omega_M - a}{b} \quad (4.11)$$

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

$$G_f(s) = \frac{1}{1 + T_f s} \quad (4.12)$$

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

$$\begin{aligned} v(s) &= k_P e(s) - k_D G_f(s) s \theta(s) \\ &= k_P e(s) - \frac{k_D s}{1 + T_f s} \theta(s) \end{aligned} \quad (4.13)$$

となる. このときの P-D 制御系を 図 4.4 に示す.

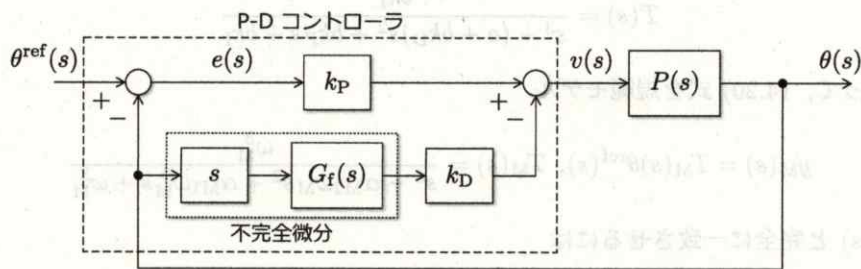


図 4.4 P-D 制御

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)

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

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

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

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,

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

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

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.

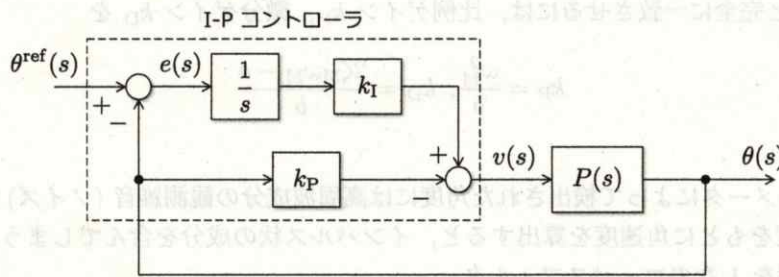


図 4.5 I-P 制御

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 $\theta^{\text{ref}}(s)$ 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

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

と選ばばよい. ただし, ω_M は速応性に関するパラメータ, α_{M1} , α_{M2} は減衰性に関するパラメータであり,

バタース標準形: $\alpha_{M1} = 2, \alpha_{M2} = 2$

二項係数標準形: $\alpha_{M1} = 3, \alpha_{M2} = 3$

ITAE 最小標準形: $\alpha_{M1} = 2.15, \alpha_{M2} = 1.75$

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。

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

このときの I-PD 制御系を図 4.6 に示す。

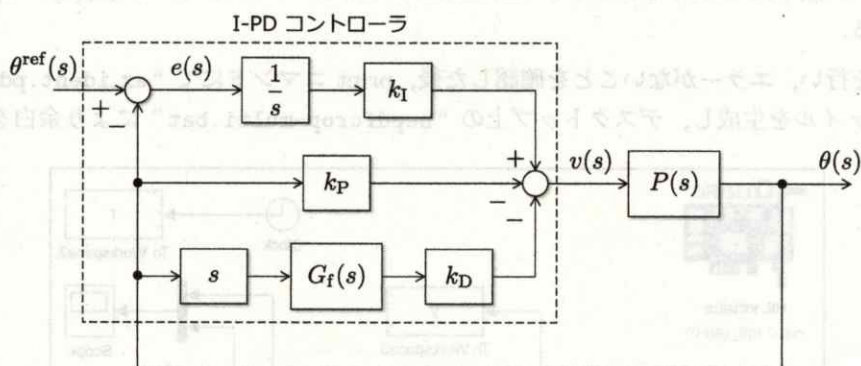


図 4.6 I-PD 制御

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 $D:\text{student_5S}\backslash\text{group01}\backslash\text{ident}$ 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 $G_f(s) = 1/(1+0.01s)$ 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。

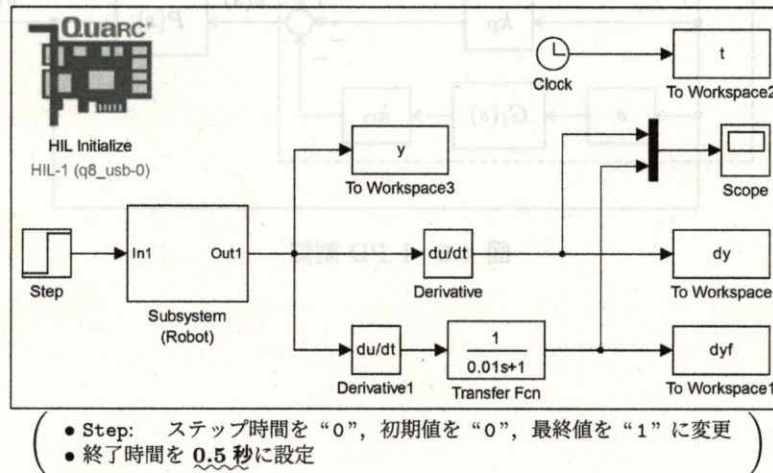


図 5.1 “ex_ident.slx”

- (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。

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

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 を完成せよ。

表 5.2 P-D コントローラ

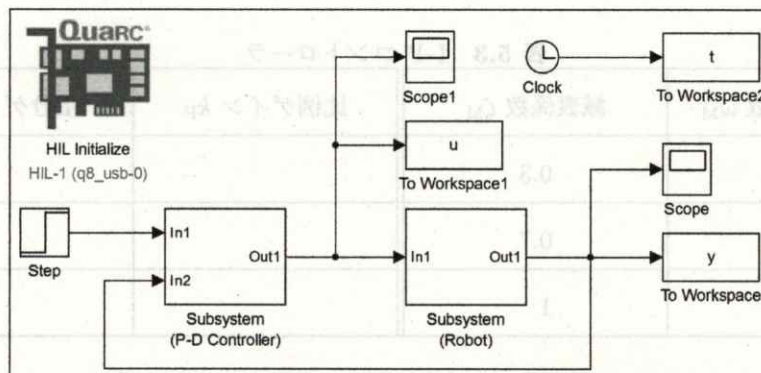
固有角周波数 ω_M	減衰係数 ζ_M	比例ゲイン k_P	微分ゲイン k_D
15	0.3		
15	0.7		
15	1		

- (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)” の内容は各自で考える (角速度 $\dot{\theta}(t)$ はローパスフィルタ $G_f(s) = 1/(1 + 0.01s)$ に通すこと)。

つぎに、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

```
>> kP = ***; kD = ***;
```

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。



- Step: ステップ時間を “0”，初期値を “0”，最終値を “1” に変更
- 終了時間を 1 秒 に設定

図 5.2 “ex_pdcont.slx”

- (3) 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。レンジは 付録 A.2 の 図 A.6 に合わせる。
- (4) 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

```
>> save pdcont_data1 t u y kP kD
```

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

- “autoplots_pdcont.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) 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。 ω_M , ζ_M を表 5.2 のように与えたとき、アームを手で 45 [deg] 程度動かしてから手をはなすと、アームがどのように振る舞うかを調べよ。

考察のポイント

- 微分動作を加えることによって過渡特性がどのように改善できるのかを考察せよ。
- 規範モデル (4.10) 式のステップ応答の定常偏差と実験結果が異なっているのはなぜか。
- 実験 (4) において、減衰係数を変化させるとアームがどのように振る舞うかを、マスバネダンパ系におけるバネ係数・ダンパ係数との関係から考察せよ。

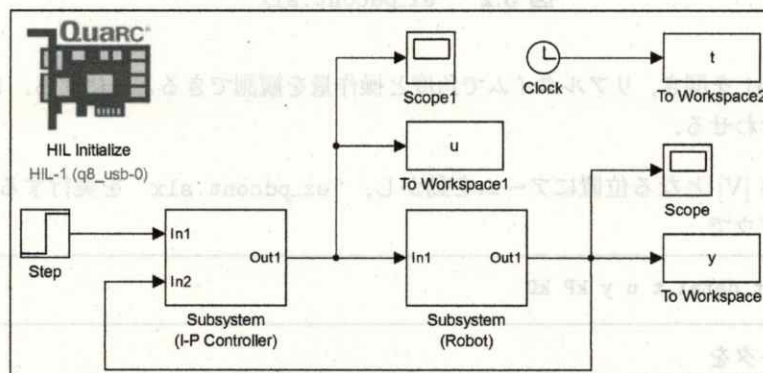
5.3 I-P 制御

- (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。

表 5.3 I-P コントローラ

固有角周波数 ω_M	減衰係数 ζ_M	比例ゲイン k_P	積分ゲイン k_I
15	0.3		
15	0.7		
15	1		

- (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。



- Step: ステップ時間を “0”, 初期値を “0”, 最終値を “1” に変更
- 終了時間を 1 秒に設定

図 5.3 “ex_ipcont.slx”

つぎに、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 (***) には表 5.3 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_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。

(3) 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。レンジは 付録 A.2 の 図 A.7 に合わせる。

(4) 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。

考察のポイント

- PI 制御と比べて各特性がどのようになっているのかを考察せよ。
- P-D 制御と比べて定常偏差がどのようになっているのかを考察せよ。
- 規範モデル (4.10) 式のステップ応答の過渡特性と実験結果が異なっているのはなぜか。

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。

表 5.4 I-PD コントローラ

ω_M	α_{M1}	α_{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			
20	3	3			
20	2.15	1.75			

5.5 L^AT_EX を利用した実験結果の印刷

ディレクトリ D:\student_5S\group01\tex2 にある “print2.tex” をタイプセット（コンパイル）し、pdf ファイルをプリントアウトせよ。

6 検討課題

1. ポテンシオメータの動作原理や長所・短所について調べよ。
2. A/D 変換, D/A 変換の基本原理について調べよ。
3. レンジ: $\pm 5[V]$, 分解能: 12 ビットとしたとき, $0x0C00$ を D/A 変換するとアナログ量はいくらになるか。
4. (4.9) 式, (4.20) 式を導出せよ。

参考文献

- [1] 大山：ためしながら学ぶモータ制御 — 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)
- [2] 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)
- [3] 須田ほか：PID 制御，朝倉書店 (1992)
- [4] 江口：初めて学ぶ PID 制御の基礎，東京電機大学出版局 (2006)
- [5] 森：演習で学ぶ PID 制御，森北出版 (2009)
- [6] 川田，西岡：MATLAB/Simulink によるわかりやすい制御工学，森北出版 (2001)
- [7] 川田：MATLAB/Simulink と実機で学ぶ制御工学 — PID 制御から現代制御まで —，TechShare (2013)
- [8] 足立：MATLAB による制御工学，東京電機大学出版局 (1999)