

仙台市/仙台市産業振興事業団  
 ロボット博士の基礎からのメカトロニクスセミナー  
 第12回 C12/Rev 1.0

# アナログセンサを マイコンにつなぐ

仙台市地域連携フェロー  
 熊谷正朗  
 kumagai@tjcc.tohoku-gakuin.ac.jp

東北学院大学工学部  
 ロボット開発工学研究室 **RDE**

## 今回の目的

- アナログセンサの使用するまで
- ステップ0: なにを測定するか
- ステップ1: センサとマイコンの選択
- ステップ2: 処理の検討
- ステップ3: AD変換の準備
- ステップ4: センサ信号処理回路の用意
- ステップ5: 変換値の確認
- ステップ6: 信号処理の実装
- ステップ7: キャリブレーション/校正

C12 アナログセンサをマイコンにつなぐ Page. 2 基礎からのメカトロニクスセミナー

## イントロダクション

### ○ センサによる測定

- ◇測定対象の情報を取得、活用
  - ・センサで電気的変化にする。
  - ・電気的変化を電圧変化にする。
  - ・適切な電圧に増幅しフィルタをかける。
  - ・アナログデジタル(AD)変換器でデジタル化。
  - ・適切な処理で情報に変換。



C12 アナログセンサをマイコンにつなぐ Page. 3 基礎からのメカトロニクスセミナー

## イントロダクション

### ○ アナログセンサとデジタルセンサ

- ◇世の中:アナログ / 処理:デジタル
  - ・温度、明るさ、力、角度、風速、などなど
  - ・最終的にはデジタルの数値情報
  - ・どこかで、デジタル化
- ◇デジタル化の場所
  - ・マイコン[内蔵/接続]のAD変換器
  - ・センサ内部のAD変換器 (次回)
  - ・最初からデジタル検出(有無で判定)

C12 アナログセンサをマイコンにつなぐ Page. 4 基礎からのメカトロニクスセミナー

## イントロダクション

### ○ センシングの鉄則 参考:第6回 センサの基礎

- ◇センサの性能以上のことはできない
  - ・センサで測れないものは測れない。
  - ・センサの精度以上の精度は得られない。
  - 測定仕様からセンサを決める。

※信号処理による工夫

- ・センサの性能引き出しの**バランス**を変える。
- ・複数のセンサの組み合わせで**欠点を補う**。

C12 アナログセンサをマイコンにつなぐ Page. 5 基礎からのメカトロニクスセミナー

## イントロダクション

### ○ センシングの鉄則 参考:第6回 センサの基礎

- ◇センサの後ろで**情報が劣化する**
  - ・回路で入る**ノイズ**や回路の**誤差**
  - ・AD変換の性能による制約  
※変換**分解能**、変換の**誤差**
  - ・信号処理過程における劣化  
※処理方法起因、**数値計算の誤差**
- ◇回路での劣化が主
  - ・**アナログ回路部分を最少にする**。

C12 アナログセンサをマイコンにつなぐ Page. 6 基礎からのメカトロニクスセミナー

## ステップ0:なにを測定するか

### ○ 必要な情報の仕様

- ◇「なにが」必要か
  - ・測定が必要な情報の種類  
例) 温度、寸法、速度etc
- ◇「どのくらいの正確さ」が必要か
  - ・**分解能と精度**



C12 アナログセンサをマイコンにつなぐ Page. 7 基礎からのメカトロニクスセミナー

## ステップ0:なにを測定するか

### ○ 必要な情報の仕様

- ◇「なにが」必要か
- ◇「どのくらいの正確さ」が必要か
  - 例)
    - ・ロボットの傾きの計測
    - ・精度 ±0.1度 分解能 1/100度
    - ・1時間以上安定、応答100Hz



C12 アナログセンサをマイコンにつなぐ Page. 8 基礎からのメカトロニクスセミナー

## ステップ0: なにを測定するか

### ○ 必要な情報の仕様

◇「どのくらいの正確さ」が必要か ~ 指標

- ・精度 : 正しさ
- ・分解能: 細かさ (桁数)
- ・安定性: どのくらいの時間継続して使うか  
10秒~10分~10年
- ・再現性: 繰り返しの一致 ※精度の一部
- ・応答性: 頻繁な変化をとらえる速度  
(直線性: 信号処理で補正)

## ステップ0: なにを測定するか

### ○ 必要な情報の仕様

◇性能仕様の重要さ

- ・全ての選定、設計、コスト、現実性に直結
- ・精度→センサ、回路、AD変換、数値計算
- ・分解能→センサ、回路のノイズレベル、  
AD変換のビット数、数値計算桁数
- ・安定性、再現性→センサと回路
- ・応答性→センサ、(回路)、AD変換速度、  
処理に要する演算パワー

## ステップ1: センサとマイコンの選定

### ○ センサの選定

◇センサの種類

- ・目的とする計測が可能、性能を持つ。
- ・信号処理で目的を達成できる。  
例) 角度のセンサで速度のセンサ  
超音波センサで風速  
※直接測定できるに超したこと無し



## ステップ1: センサとマイコンの選定

### ○ センサの選定

◇信号処理を挟む場合の性能評価

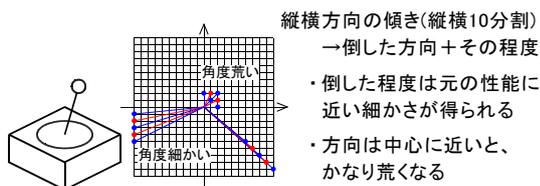
- ・応答性 : 低下する場合有り
- ・線形変換: 精度分解能は単純計算
- ・非線形変換: 局所的に悪化の可能性
- ・微分: ノイズの影響拡大、安定性改善
- ・積分: ノイズは低減、安定性が大幅悪化  
※ゼロ点がずれたものを積分する危険性
- ・事前に表計算等で検証必要 & 可能

## ステップ1: センサとマイコンの選定

### ○ センサの選定

◇信号処理を挟む場合の性能評価

- ・非線形変換: 局所的に悪化の可能性  
例) ジョイスティックを倒した方向



## ステップ1: センサとマイコンの選定

### ○ マイコンの選定

◇マイコン選定のキーポイント

- ・適切なAD変換器を内蔵していること  
分解能、精度、変換速度、チャンネル数
- ・信号処理をうる演算速度を持つこと  
1回あたりの処理時間 < 1/処理周波数
- ・開発体制、コストなど



## ステップ1: センサとマイコンの選定

### ○ マイコンの選定

◇速度の見積

- ・1秒あたりの処理回数 =  
必要な応答性の4~10倍  
→1回の計算時間、ADの変換速度  
※サンプリング定理 (→C05)
- ・マイコン側の処理を速くすることで、  
アナログ部の負担を低減する手法あり。
- ・一般にはさほど問題にならない。

## ステップ1: センサとマイコンの選定

### ○ マイコンの選定

◇AD変換部の選定

- ・外付けはコストなどのため非推奨
- ・マルチプレクサ(切替スイッチ)の外付けも同  
→ 分解能、精度、チャンネル数は  
マイコン選定の重要要素

◇この先、必要なもの

- ・AD変換部の電氣的仕様:  
入力電圧範囲、入力抵抗(容量)

## ステップ2: 処理の検討

### ○ 計測に必要な処理のリストアップ

#### ◇一般的に必要な処理

- ・大きさ調整 信号増幅、単位変換
- ・ゼロ点調整 オフセット調整  
※校正の一部、無信号値の補正
- ・ノイズ除去 ローパスフィルタ
- ・微積、関数変換 測定値の種類変更



## ステップ2: 処理の検討

### ○ 計測に必要な処理の分配

#### ◇一般的に必要な処理

- ・大きさ調整 → 回路 & 信号処理
- ・ゼロ点調整 → 回路 & 信号処理  
※校正の一部、無信号値の補正
- ・ノイズ除去 → 回路 & 信号処理
- ・微積、関数変換 → 信号処理



## ステップ2: 処理の検討

### ○ 計測に必要な処理の分配

#### ◇ポイント = 回路の処理を最低限に

- ・回路の仕事はADに適切に電圧を渡す。

#### ◇大きさの調整 → 回路 & 信号処理

- ・センサ出力をADの電圧レンジにフィットさせるための回路。
- ・AD値を必要な形式の数値に直す。

## ステップ2: 処理の検討

### ○ 計測に必要な処理の分配

#### ◇ゼロ点調整 → 回路 & 信号処理

- ・「ゼロ」状態で計測を行って、その値を基準として記憶→動作時に使用。  
※「ゼロ」以外でも既知の状態なら可  
※例)0度で100回ADして平均値
- ・回路でのゼロ点調整 (後述):
  - ・AD変換のレンジを有効活用
  - ・増幅してレンジ外にしないため調整

## ステップ2: 処理の検討

### ○ 計測に必要な処理の分配

#### ◇ノイズ除去 → 回路 & 信号処理

- ・特殊なノイズ(例:50Hz)はソフトで除去したほうが良いことが多いが、単なる高周波ノイズはアナログローパスフィルタが楽。(とくにリアルタイム制御に用いる場合)
- ・AD変換に入れる前に、そのサンプリング周波数より高いノイズ、成分は落とす。

## ステップ3: AD変換の準備

### ○ マイコン周りのハード設計とソフト

#### ◇とにかく、アナログ電圧を取得できるように

#### ◇ハード面

- ・入力端子の適切な処理
- ・基準電圧の与え方  
→ AD変換のフルスケールを決定  
[マイコン電源流用、専用に用意]



## ステップ3: AD変換の準備

### ○ マイコン周りのハード設計とソフト

#### ◇ソフト面

- ・AD変換機能そのもののテスト
- ・入力の切り替え方法 (多チャンネルの場合)  
[チャンネル毎別レジスタ、DMAによるメモリ格納等]
- ・変換速度の設定  
[フリーラン、割込駆動、ソフト指令]



## ステップ3: AD変換の準備

### ○ マイコン周りのハード設計とソフト

#### ◇注意点

- ・システムを全て作ってしまう前に  
(なるべく、詳細設計に入る前に)  
AD変換部の単独動作を確認する。
- ※ マイコンによっては最難関ポイント
- ※ 他の工夫では性能改善が困難
- ※ 仕様書の読み間違いを避ける

## 今回の目的

### ○ アナログセンサの使用法

- ステップ0: なにを測定するか
- ステップ1: センサとマイコンの選択
- ステップ2: 処理の検討
- ステップ3: AD変換の準備
- ステップ4: センサ信号処理回路の用意
- ステップ5: 変換値の確認
- ステップ6: 信号処理の実装
- ステップ7: キャリブレーション/校正

## ステップ4: センサ信号処理回路の用意

### ○ センサとAD変換器入力の中継

- ◇センサ出力の電圧信号への変換
  - ・電圧出力以外のセンサの場合
- ◇電圧レンジのマッチング
  - ・増幅、オフセット調整
- ◇ローパスフィルタ (たまにハイパス)
  - ・ノイズの除去、周波数帯域の選択



## ステップ4: センサ信号処理回路の用意

### ○ センサ出力の電圧信号への変換

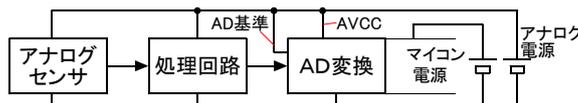
#### ◇センサの出力

- ・電圧出力
  - 内部に処理回路を含むセンサ: 扱い楽(注意あり)
  - 原理的に電圧出力: 受ける回路に要注意
- ・電流出力
  - 光センサ (光量に比例した電流): 電流電圧変換
- ・抵抗変化
  - 測定対象によって抵抗値が変化する。
  - 電流を流して電圧にする。

## ステップ4: センサ信号処理回路の用意

### ○ センサ出力の電圧信号への変換

- ◇電圧出力センサ (回路を含むセンサ)
  - ・出力強く、後の回路が自由(AD直結も)。
  - ・一般に電源の供給が必要。
  - ・出力が電源電圧比例の場合あり。
    - = 電源電圧の正確さ必要(レシオメトリック)



## ステップ4: センサ信号処理回路の用意

### ○ センサ出力の電圧信号への変換

#### ◇電圧出力センサ (回路を含むセンサ)

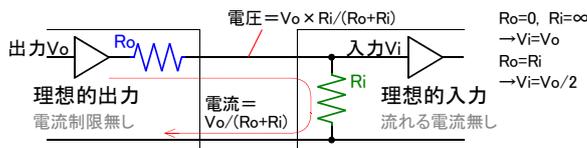
- ・データシートを読むときの注意
  - ・出力電圧が電源に依存するか。
    - (単に出力00Vでは確認できず)
    - < 大抵は標準的な電源のときの出力値が記載)
  - ・電源のノイズがどの程度影響するか。
    - 電源周りのコンデンサ外付けなど
- ・出カインピーダンス

## ステップ4: センサ信号処理回路の用意

### ○ センサ出力の電圧信号への変換

#### ◇出カインピーダンスと入力インピーダンス

- ・内部に直列/並列に入っていると見なす。
- ・出カインピーダンス  $R_o$ : 低いほど良い
- ・入力インピーダンス  $R_i$ : 高いほどよい

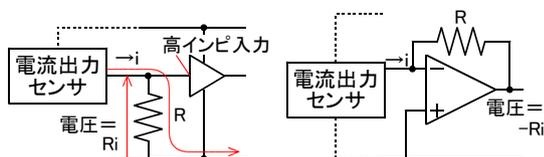


## ステップ4: センサ信号処理回路の用意

### ○ センサ出力の電圧信号への変換

#### ◇電流出力センサ

- ・抵抗にその電流を流して電圧に
- ・オペアンプによる電流-電圧変換回路

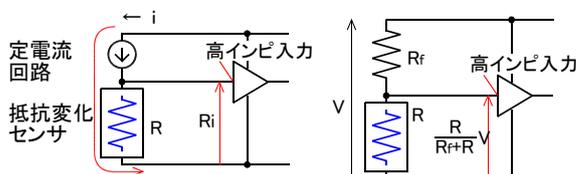


## ステップ4: センサ信号処理回路の用意

### ○ センサ出力の電圧信号への変換

#### ◇抵抗変化型センサ

- ・抵抗に(一定の)電流を流して電圧に
- ・固定抵抗と分圧回路を構成



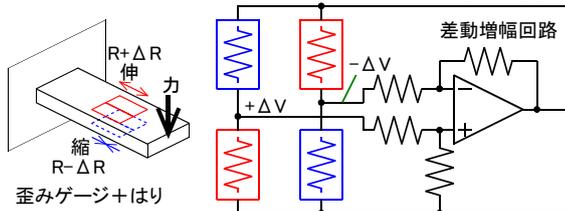
### ステップ4: センサ信号処理回路の用意

#### ○ センサ出力の電圧信号への変換

##### ◇ 抵抗変化型センサ (ブリッジ)

・ 微小な変化を抽出

※正式にはインストゥルメンテーションアンプ



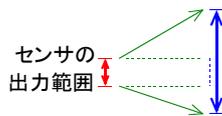
### ステップ4: センサ信号処理回路の用意

#### ○ 電圧レンジのマッチング

##### ◇ センサ、変換回路 と AD変換の整合

・ 電圧範囲が異なる場合

・ AD範囲の有効活用のために増幅 + 範囲の上下



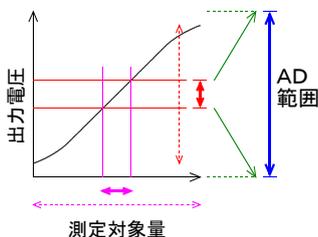
### ステップ4: センサ信号処理回路の用意

#### ○ 電圧レンジのマッチング

##### ◇ センサ出力の一部拡大→高感度化

・ センサの出力範囲はADに一致している。が、一部を大きく増幅することで、測定対象の範囲を絞る。  
→ 高感度(高分解能)に

※精度は低下する可能性あり

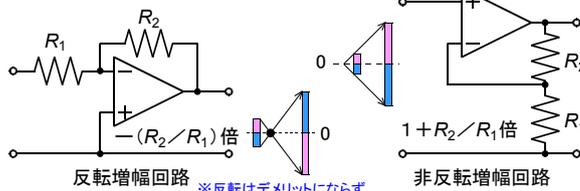


### ステップ4: センサ信号処理回路の用意

#### ○ 電圧レンジの変更: 増幅

##### ◇ 基本的なオペアンプ増幅回路 (単純)

・ 反転増幅 : 基本的、正負反転、入力低め  
・ 非反転増幅 : 入力インピーダンス高

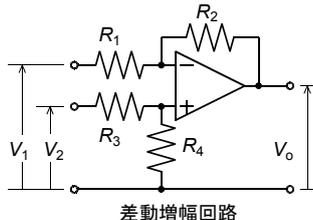


### ステップ4: センサ信号処理回路の用意

#### ○ 電圧レンジの変更: 増幅

##### ◇ 電圧の差の増幅

・ 差動増幅、インストゥルメンテーションアンプ



$$R_3 = R_1, R_4 = R_2 \text{ として、} \\ V_o = (R_2/R_1) (V_2 - V_1)$$

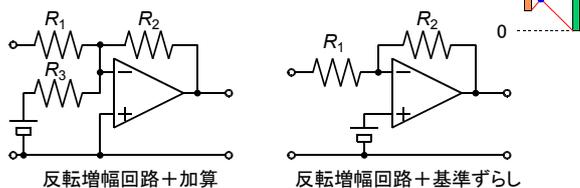
インストゥルメンテーションアンプ (計装アンプ) は、入力インピーダンスを高めた差動増幅回路の一種。

### ステップ4: センサ信号処理回路の用意

#### ○ 電圧レンジの変更: オフセット

##### ◇ 電圧の範囲を上下させる

・ 別の電圧を加算  
・ 増幅の基準点をずらす



### ステップ4: センサ信号処理回路の用意

#### ○ アナログフィルタ回路

##### ◇ ローパスフィルタ

・ 高周波ノイズを除去 (増幅中、前後)  
・ AD変換前のアンチエイリアシング  
※高い周波数成分が化けるのを防ぐ→C05

##### ◇ ハイパスフィルタ (計測ではあまり多くない)

・ 直流分、低周波の変動を除去

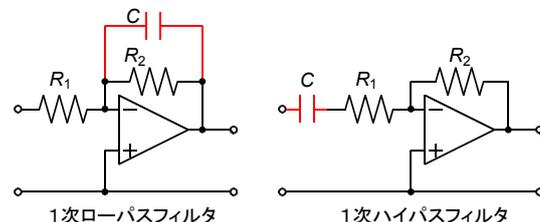


### ステップ4: センサ信号処理回路の用意

#### ○ アナログフィルタ回路

##### ◇ ローパス、ハイパスの基本回路(一次)

・ 反転増幅回路にコンデンサを追加するだけ

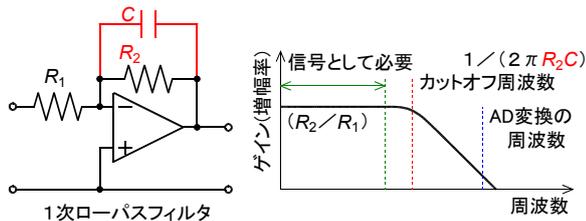


## ステップ4: センサ信号処理回路の用意

### ○ ローパスフィルタと信号の周波数

#### ◇周波数の関係

必要な信号応答 < ローパス < AD変換



## ステップ4: センサ信号処理回路の用意

### ○ 回路設計の手順

- 4-1: センサ出力を電圧に直す回路の検討  
※出力インピーが高い電圧出力の対処含む
- 4-2: 上記出力とADの入力範囲の確認
- 4-3: 間の増幅回路を検討  
・オフセット付き、単電源、場合によって多段
- 4-4: ローパス用コンデンサの検討



## 今回の目的

### ○ アナログセンサの使用法

- ステップ0: なにを測定するか
- ステップ1: センサとマイコンの選択
- ステップ2: 処理の検討
- ステップ3: AD変換の準備
- ステップ4: センサ信号処理回路の用意
- ステップ5: 変換値の確認
- ステップ6: 信号処理の実装
- ステップ7: キャリブレーション/校正

## ステップ5: 変換値の確認

### ○ ここまでの動作の妥当性検証

#### ◇「そこそこ」適切に値が出ているか

- ・測定対象の変動に対して値が変わるか。
- ・測定対象の範囲で、値が全域に渡るか。
- ・想定外のノイズが乗っていないか。
- ・他の入力の影響を受けていないか。



## ステップ5: 変換値の確認

### ○ ここまでの動作の妥当性検証

#### ◇「そこそこ」適切に値が出ているか

- ・このチェックには「マイコン内の値を出す」手段が必要。(事前準備)
- 例) シリアルポート経由 (低~中速)
- 液晶表示器などに出力(低速)
- DA変換から出力→オシロ (~高速)



## ステップ6: 信号処理の実装

### ○ 生のAD変換値を情報に

#### ◇処理内容の確認

- ・想定データ、可能ならAD後の実データを表計算、Cなどを使って処理の確認する。
- ・わかりやすい手段で十分に検証し、処理方法を練っておく(マイコン上で困難)。



## ステップ6: 信号処理の実装

### ○ 生のAD変換値を情報に

#### ◇マイコンであるがゆえに

- ・基本的に整数計算  
加減乗算(除算)+シフト( $\times 2^{\pm n}$ )  
※整数演算の確認もPC上で済ます
- ・変数の検討は「有効桁数」+小数点位置  
16bit: 10進数で5桁程度 (場合によって不足)  
32bit: 10進数で9桁程度 (足りること多い)
- ・オフライン可なら無理にマイコンにしない。

## ステップ6: 信号処理の実装

### ○ 信号処理の一例

#### ◇微分(→速度) 積分(速度→)

入力データを  $u[i]$  ( $i=0,1,2,\dots$ )、出力  $y[i]$ 、ADの時間間隔(サンプリング周期)を  $T$  として:

- ・微分  
 $y[i] = (u[i] - u[i-1]) / T$  (今回と前回の差)  
 $y[i] = (u[i] - u[i-1])$  ( $T$ が一定→省く)
- ・積分  
 $y[i] = y[i-1] + u[i] \times T$  ( $y[i-1] + u[i]$ )

## ステップ6: 信号処理の実装

### ○ 信号処理の一例

#### ◇ローパスフィルタ

$$y[i] = (1-r) \times y[i-1] + r \times u[i]$$

- ・r はフィルタの程度を決める小さな定数。  
例) 0.01      0.99y + 0.01u

- ・前回の値(y[i-1])に、少し(r)だけ、  
今回の入力(u[i])を混ぜる。  
= 入力の変化が徐々に反映される

- ・rが小さいほどカットオフ周波数がさがる。

## ステップ6: 信号処理の実装

### ○ 信号処理の一例

#### ◇オフセットの除去

##### ○ オフセット計測モード

- ・N回にわたって、u[i]を平均化  
ofs=0 → ofs=ofs+u[i] (N回) → ofs=ofs/N

##### ○ 通常時

- ・s[i]=u[i]-ofs  
s[i]はオフセット除去済みの入力値

※オフセット計測は「0の状態」(基準位置など)で行う

## 今回の目的

### ○ アナログセンサの使用法

- ステップ0: なにを測定するか
- ステップ1: センサとマイコンの選択
- ステップ2: 処理の検討
- ステップ3: AD変換の準備
- ステップ4: センサ信号処理回路の用意
- ステップ5: 変換値の確認
- ステップ6: 信号処理の実装
- ステップ7: キャリブレーション/校正

## ステップ7: キャリブレーション

### ○ センサや回路のずれなどを吸収

#### ◇センシング回路の実態

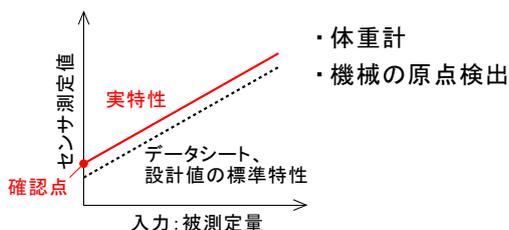
- ・センサのゼロ点や感度にはばらつき。
- ・センサの取り付け方の影響も受ける。
- ・回路の抵抗値の誤差による増幅率誤差。
- ・ただし、個体ごとの再現性高し。  
→ センサと回路の誤差を反映させる



## ステップ7: キャリブレーション

### ○ キャリブレーションの一例 (ゼロ点のみ)

「ゼロ」に対する出力を測定 +  
被測定値に対する感度(傾き固定)

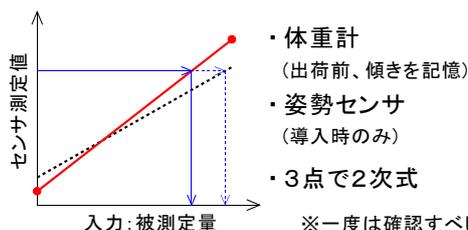


- ・体重計
- ・機械の原点検出

## ステップ7: キャリブレーション

### ○ キャリブレーションの一例 (直線的)

測定対象区間で2カ所の測定を行い、  
その間を一次式(y=ax+b, x=(y-b)/a)で求める。



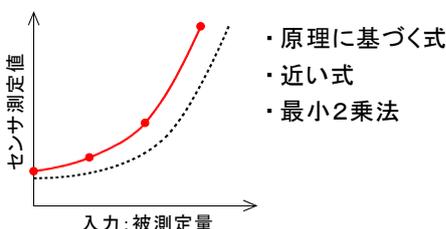
- ・体重計 (出荷前、傾きを記憶)
- ・姿勢センサ (導入時のみ)
- ・3点で2次式

※一度は確認すべし

## ステップ7: キャリブレーション

### ○ キャリブレーションの一例 (非直線的)

何点かで対応関係を測定する  
→ その点(の近く)を通る曲線で近似する。



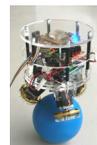
- ・原理に基づく式
- ・近い式
- ・最小2乗法

## 開発の実例: 玉乗りロボットの姿勢センサ

### ○ 概要

#### ◇バランスを取るロボット

- ・本体の傾斜角度、傾斜角速度をもとに、車輪を回してバランス



#### ◇要求仕様

- ・長時間の安定性、応答性50Hz程度
- ・角度および角速度分解能: 高いほどよい
- ・測定範囲: 鉛直から±10度程度  
2.30度/秒程度の速度

## 開発の実例：玉乗りロボットの姿勢センサ

### ○ センサとマイコンの選定



#### ◇センサ

- ・角速度：アナデバ ADXRS401  
75deg/s, 5V動作, 2.5V基準 2kHz
- ・加速度：アナデバ ADXL203  
1.7g, 5V動作, 2.5V基準

#### ◇マイコン

- ・ルネサス H8/3052, 16bit  
AD: 10bit 8ch

## 開発の実例：玉乗りロボットの姿勢センサ

### ○ 処理の検討



#### ◇センサごとの処理

- ・AD変換: 16kHz →ソフトでローパス
- ・角速度センサ →感度向上のため増幅
- ・加速度センサ →感度向上のため増幅

#### ◇センサ融合処理 (合成フィルタ)

- ・角速度 ←角速度センサ
- ・角度 応答性(高周波数)←角速度積分
- 合成 安定性(低周波数)←加速度傾斜

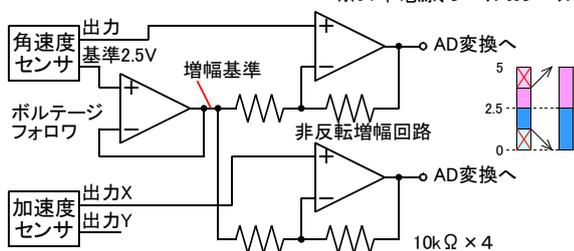
## 開発の実例：玉乗りロボットの姿勢センサ

### ○ 回路の概要



#### ◇2.5V基準、2倍増幅、非反転

※5V単電源、レールtoレール



## 開発の実例：玉乗りロボットの姿勢センサ

### ○ キャリブレーション方法



#### ◇方針

- ・そもそも固定小数点でぎりぎりの実装なので、SI単位系やdegなどは使わず。
- ・角速度センサを基準とした単位に  
角速度: AD変換値そのもの  
角度: AD変換値の積分(×Tは省く)
- ・加速度の測定値(AD値)に係数をかけて角速度側に合わせてから、合成。

## 開発の実例：玉乗りロボットの姿勢センサ

### ○ キャリブレーション方法



#### ◇ゼロ点のキャリブレーション

- ・水平、静止の状態ですら1秒程度(256回)AD変換値を測定して平均。

#### ◇角度のキャリブレーション

- ・ロボットを5度程度傾け、角速度センサの積分値、加速度センサの値を比較し、その感度の係数を求める。

## まとめ：これまでのメカトロセミナー関連

### ○ 検索:[ロボット開発工学]→メカトロセミナー

- S0: なにを測定するか →C06,C07
- S1: センサとマイコンの選択 →C06,C05
- S2: 処理の検討 →C07
- S3: AD変換の準備 →C05,C02
- S4: センサ信号処理回路の用意 →C03
- S5: 変換値の確認 (→C11)
- S6: 信号処理の実装 →C07
- S7: キャリブレーション/校正 →C07

## まとめ

### ○ アナログセンサをマイコンにつなぐ

- ・重要な選定ポイントは  
センサ本体  
AD変換器  
処理のためのソフトと電子回路  
である。必要な精度、分解能、応答性に  
応じた、これらの要素が求められる。
- ・アナログ部分をなるべく少なくすることが  
性能やコストの点で有利である。

## まとめ

### ○ 開発の手順

- ・開発手順はブロック図の通りではない。
- ・主要箇所を選定し、その仕様を確認して、**仕様と仕様をつなぐ**ための設計を行う。  
センサ～AD変換器  
AD変換器～実際に必要な値
- ・調整を行う箇所を予め想定しておく。  
ソフトの調整は容易であるが、ハードで調整すべきことも多い。