

マイコン講習

—第6回—

K.Miyauchi

2021.10.07

目次

- ・センサの基礎
- ・増幅回路
- ・フィルタ回路
- ・インピーダンス変換
- ・CDSセルのフィルタ
- ・加速度センサ

本講習で使用するもの

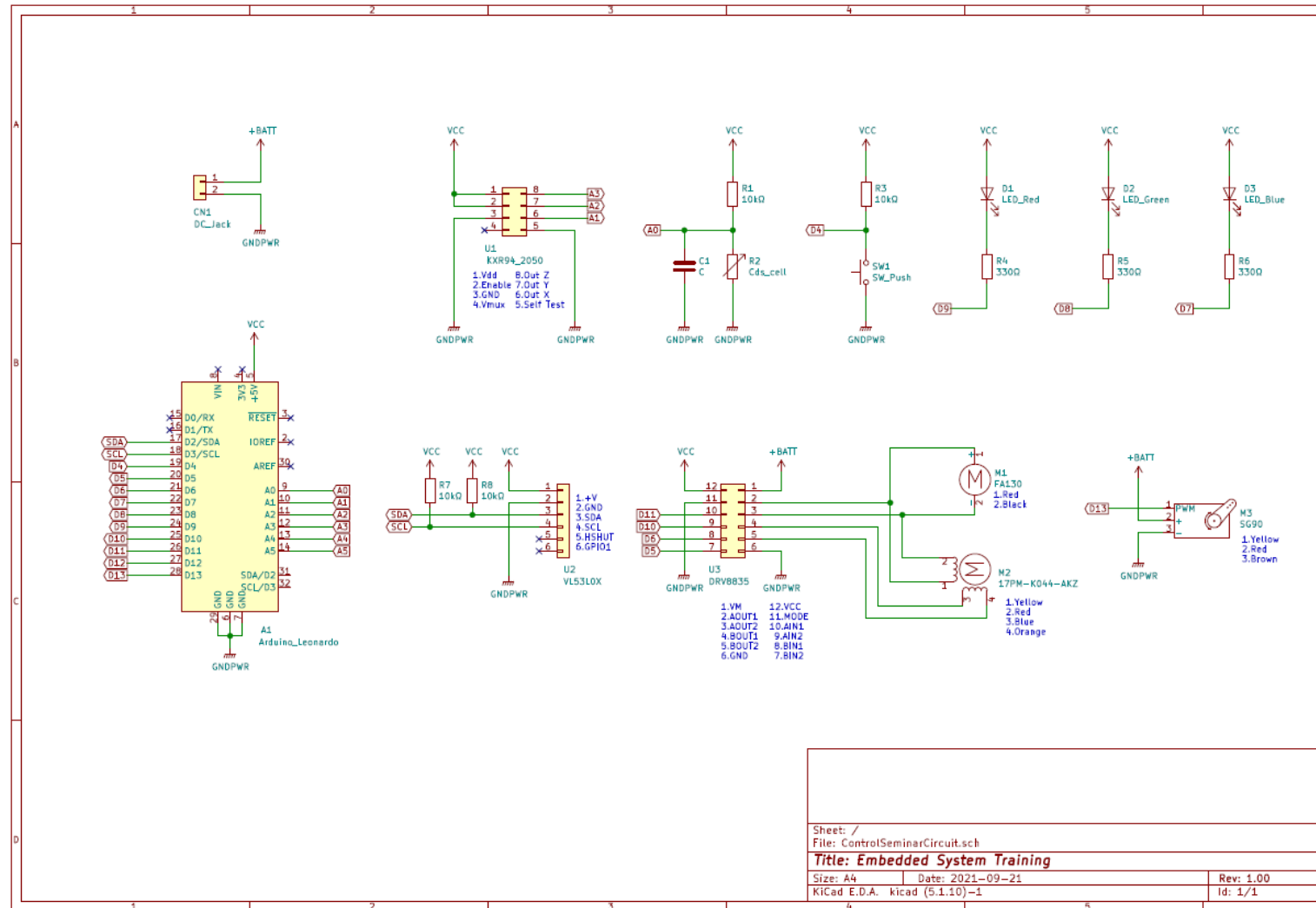
パソコン: Windows, Mac

開発環境: Arduino IDE

マイコン: Arduino Leonardo

その他, 回路周りの物品

回路図



センサの基礎

センサとは、物理量を電気信号として出力するデバイスのこと。

力や光，加速度，角速度を計測するためのセンサがある。

また，センサによっては，**アナログ信号**で出力されるものと**デジタル信号**で出力されるものがある。

センサの基礎

アナログ信号で出力されるセンサは、出力される電圧をADC等によって読み取ることができる。

しかし、電圧の変化が微弱であったり、ノイズが含まれていたりすることが多いため、増幅器やフィルタなどを組むことが必要になることもある。

センサの基礎

アナログ信号は連続時間であるが、それを離散時間の信号とすることを**サンプリング**(標本化)という。

離散時間にしたときに、1秒間に何回、データを取得したかを表すパラメータとして、**サンプリング周波数**がある。

このサンプリング周波数は、**サンプリング定理**より、原信号がもつ最高周波数の2倍以上とすることによって、原信号を完全に復元できる。

このことを満たさない場合、原信号と異なる信号が再現されることがあり、これを**エイリアシング**という。

センサの基礎

コンピュータで扱うために、標本化したデータを有限桁にする必要がある。この操作のことを量子化という。

この量子化した値をPCに転送するために、2進化する必要がある。この操作のことを符号化という。

AD変換器では、これらの操作をすべて行っている。

センサの基礎

デジタル信号で出力されるセンサは、様々な形式で出力される.

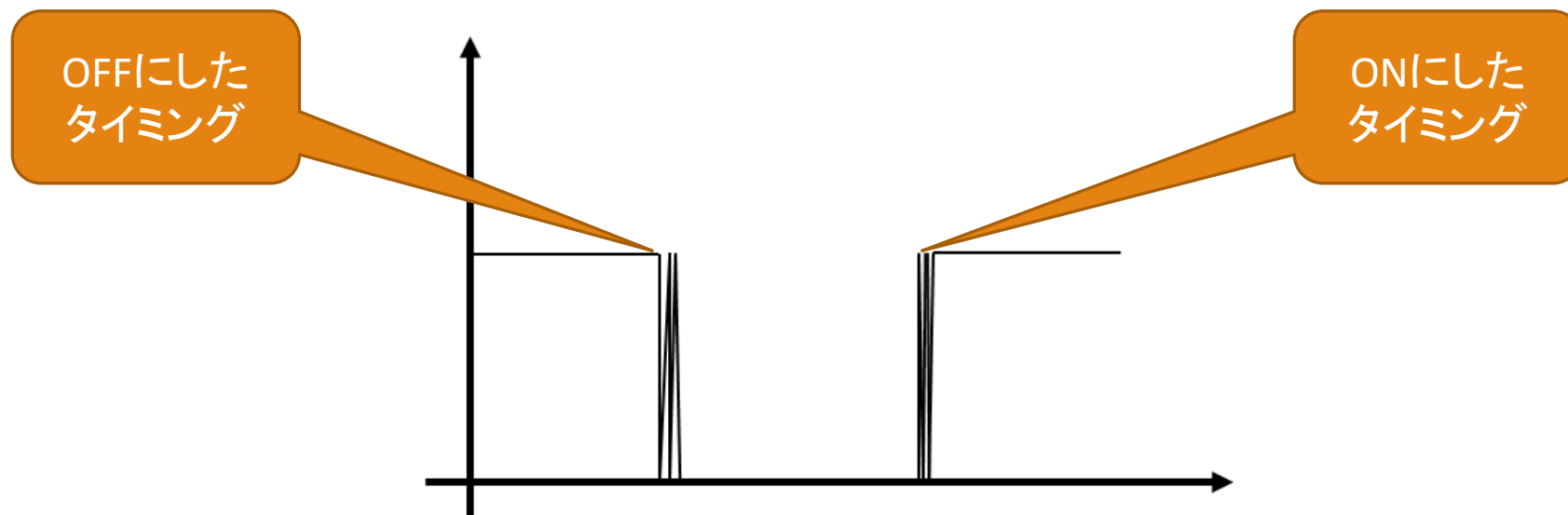
- バイナリコード
- グレイコード
- パルス
- PWM
- シリアル通信 etc...

センサの仕様書を見て、それに合わせてシステムを設計する必要がある.

センサの基礎

マイクロスイッチなどと呼ばれるような**機械式センサ**では、機械的接点がつながった時や放れたときにチャタリングと呼ばれる現象が生じる。

単純なカウンタやパルスカウントをするにしても、チャタリングを考慮してシステムを組む必要がある。



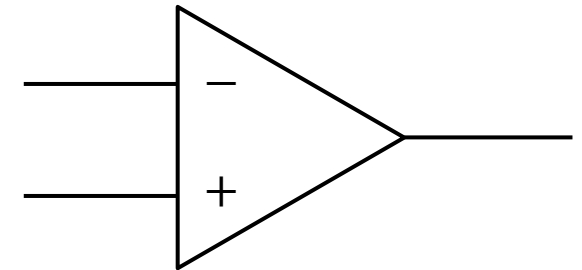
増幅回路

微弱な信号を大きな信号に変換する回路のことを増幅回路という。

増幅回路には、トランジスタを用いたものとオペアンプを用いたものがありますが、今回はオペアンプを用いたもののみの説明。

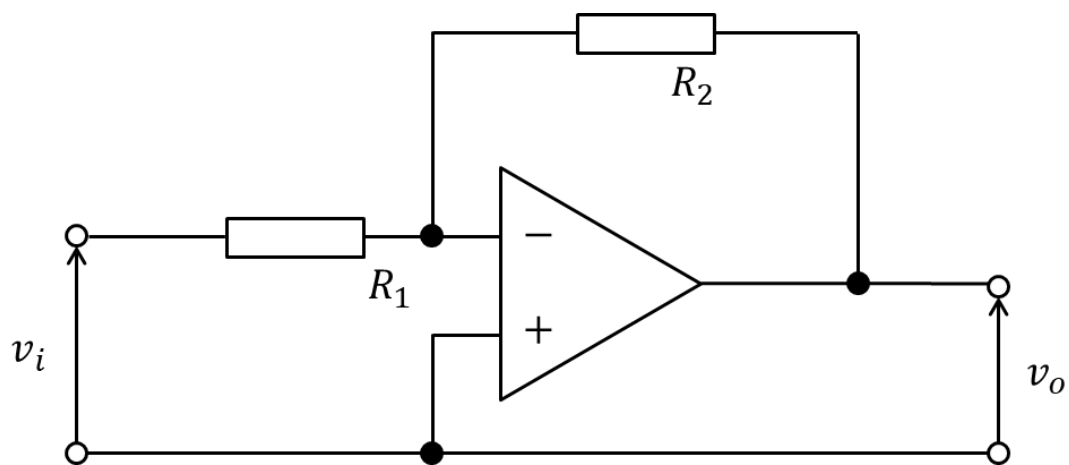
＜オペアンプの特徴＞

- ・反転入力端子(－)と非反転入力端子(＋)は入力インピーダンスが無限大と仮定してもよい。
⇒2端子間の電位がほぼ同じ(イマジナルショート)
- ・出力インピーダンスが小さい。

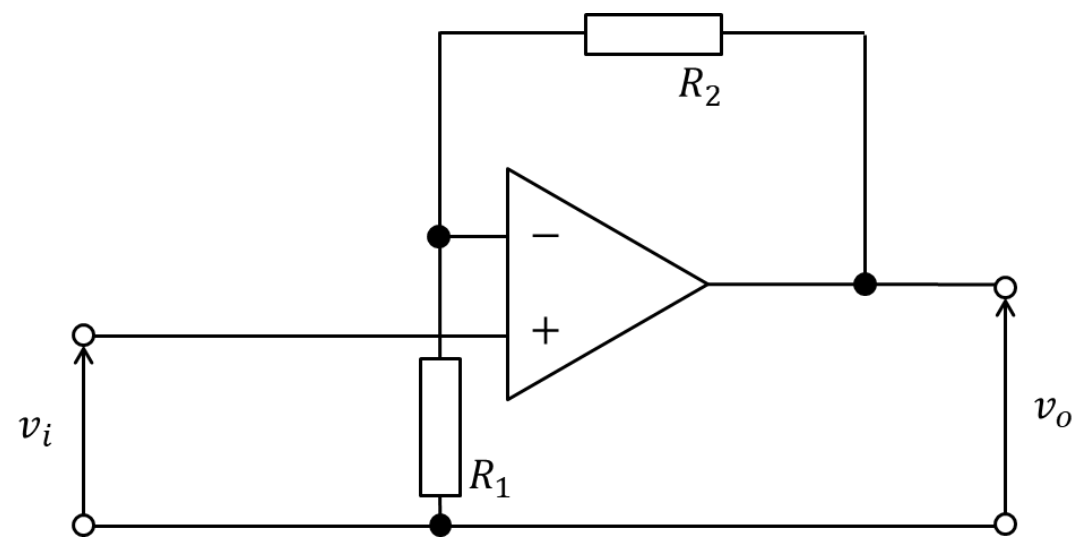


増幅回路

オペアンプを用いた増幅回路として、
反転増幅回路と非反転増幅回路がある。



反転増幅回路

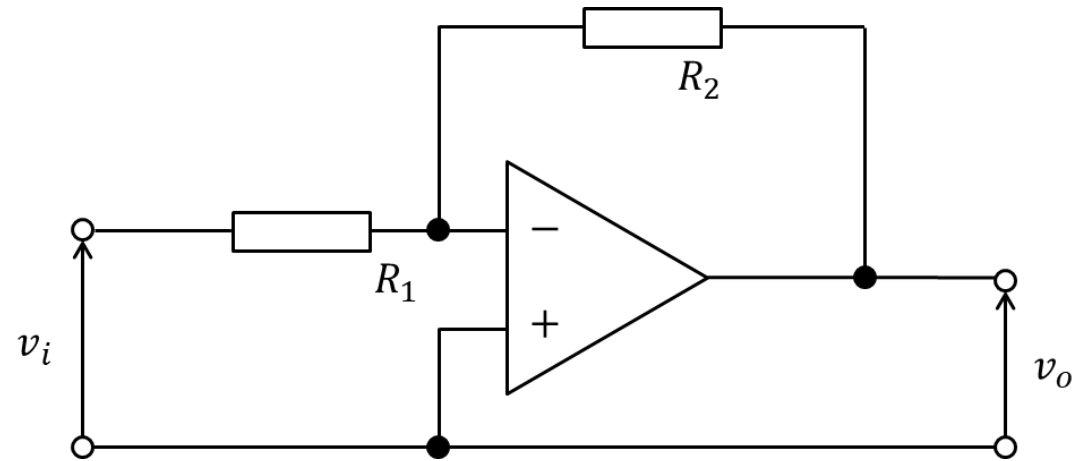


非反転増幅回路

増幅回路

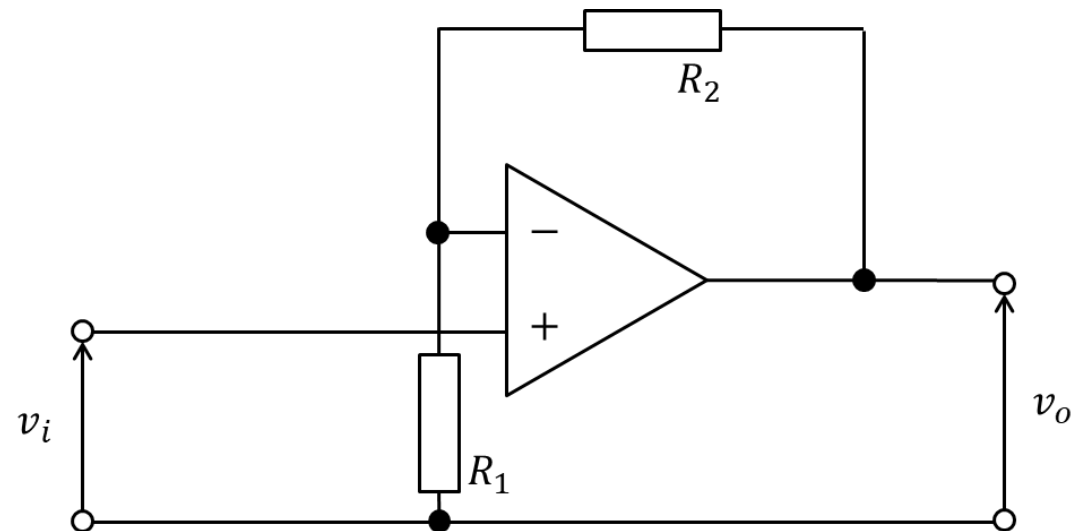
反転増幅回路はその名前の通り、
入力信号の位相を反転して信号
増幅する回路である。

$$\text{関係式: } v_o = -\frac{R_2}{R_1} v_i$$



非反転増幅回路は、入力信号の
位相を変えずに信号増幅する回
路である。

$$\text{関係式: } v_o = \left(1 + \frac{R_2}{R_1}\right) v_i$$



フィルタ回路

フィルタ回路とは、特定の範囲の周波数の信号のみカットする回路のことをいう。

フィルタ回路として、

- ・ハイパスフィルタ(低周波をカットする)
- ・ローパスフィルタ(高周波をカットする)
- ・バンドパスフィルタ(指定範囲外をカットする)

がある。

またカットし始める周波数のことを**カットオフ周波数**という。

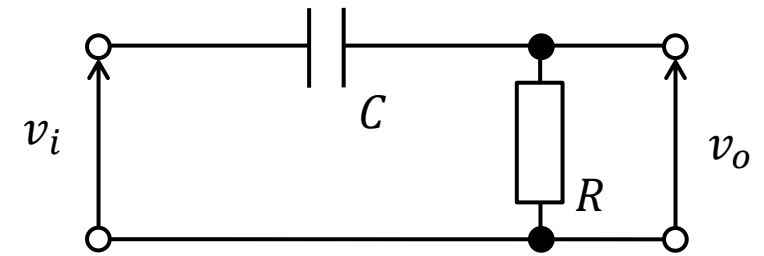
フィルタ回路

シンプルなフィルタ回路

＜ハイパスフィルタ＞

$$\text{関係式: } v_o = \frac{j\omega RC}{1+j\omega RC} v_i$$

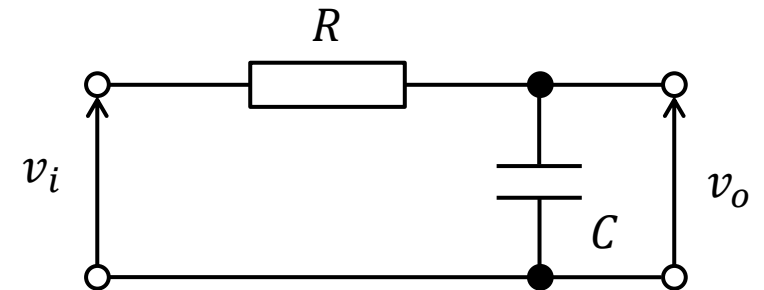
$$\text{カットオフ周波数: } f_c = \frac{1}{2\pi RC}$$



＜ローパスフィルタ＞

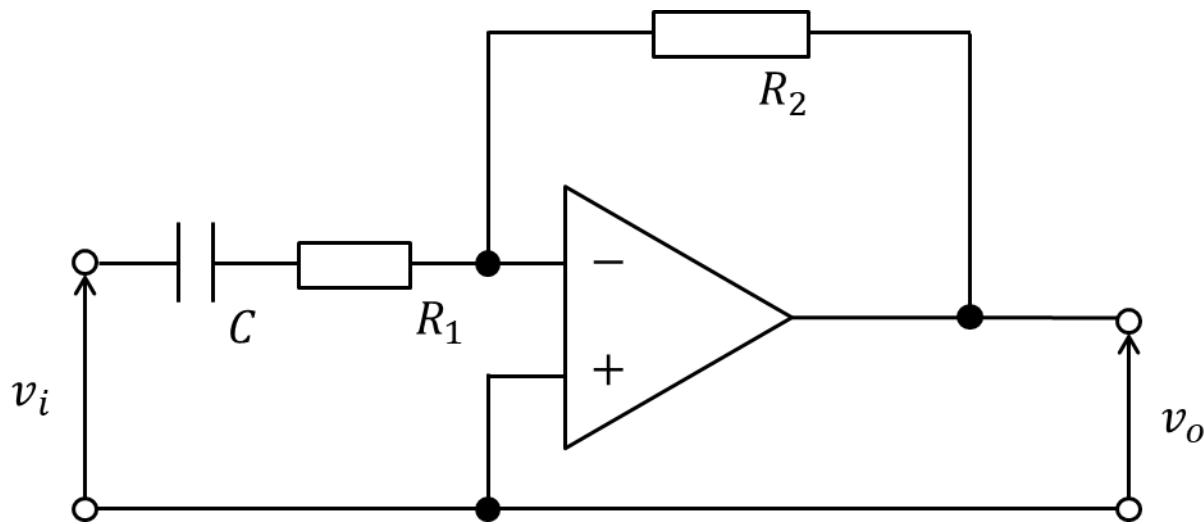
$$\text{関係式: } v_o = \frac{1}{1+j\omega RC} v_i$$

$$\text{カットオフ周波数: } f_c = \frac{1}{2\pi RC}$$

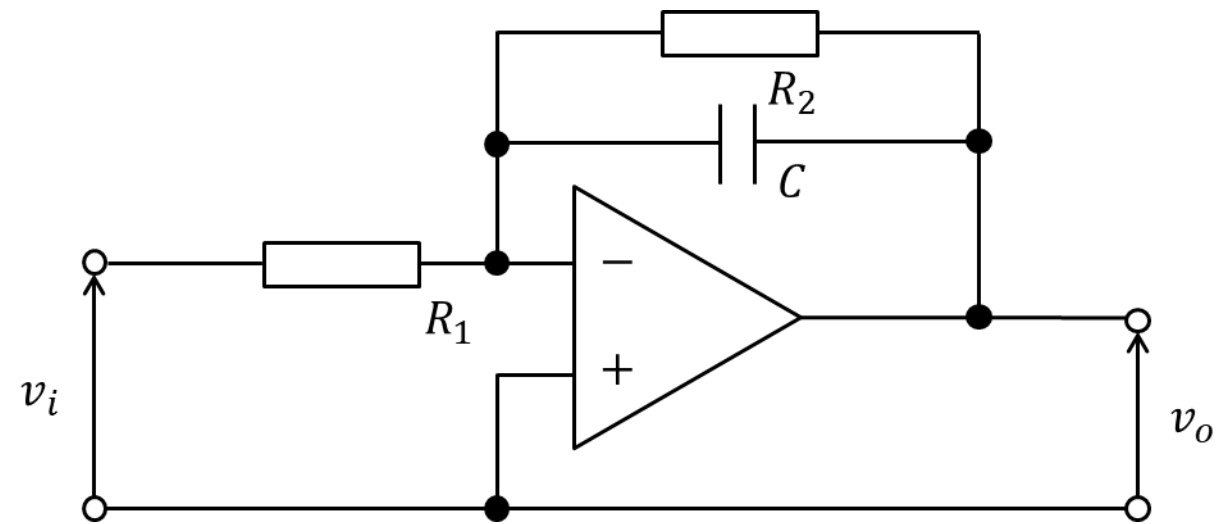


フィルタ回路

原信号が十分に計測可能であれば、さきほどのフィルタでほとんど解決することはできるが、微弱な信号であるときには、増幅器を含めたフィルタを組み込む必要がある。



ハイパスフィルタ



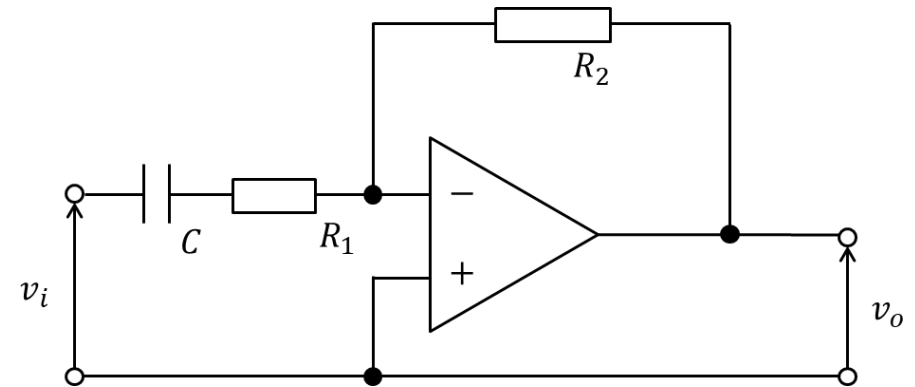
ローパスフィルタ

フィルタ回路

＜ハイパスフィルタ＞

$$\text{関係式: } v_o = -\frac{R_2}{R_1} \frac{j\omega R_1 C}{1+j\omega R_1 C} v_i$$

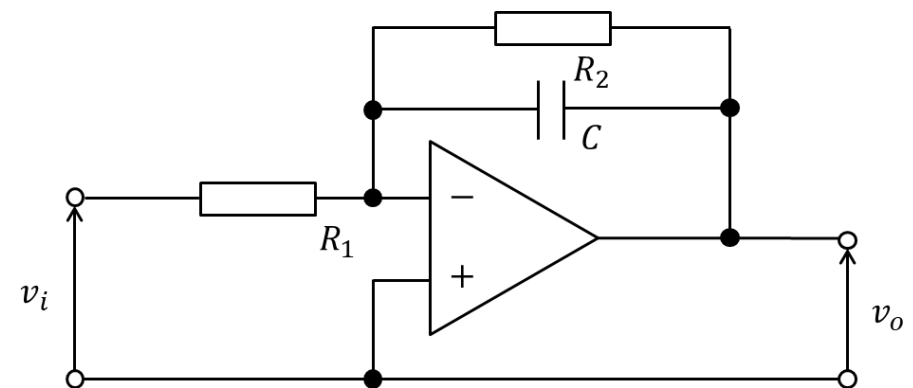
$$\text{カットオフ周波数: } f_c = \frac{1}{2\pi R_1 C}$$



＜ローパスフィルタ＞

$$\text{関係式: } v_o = -\frac{R_2}{R_1} \frac{1}{1+j\omega R_2 C} v_i$$

$$\text{カットオフ周波数: } f_c = \frac{1}{2\pi R_2 C}$$



インピーダンス変換

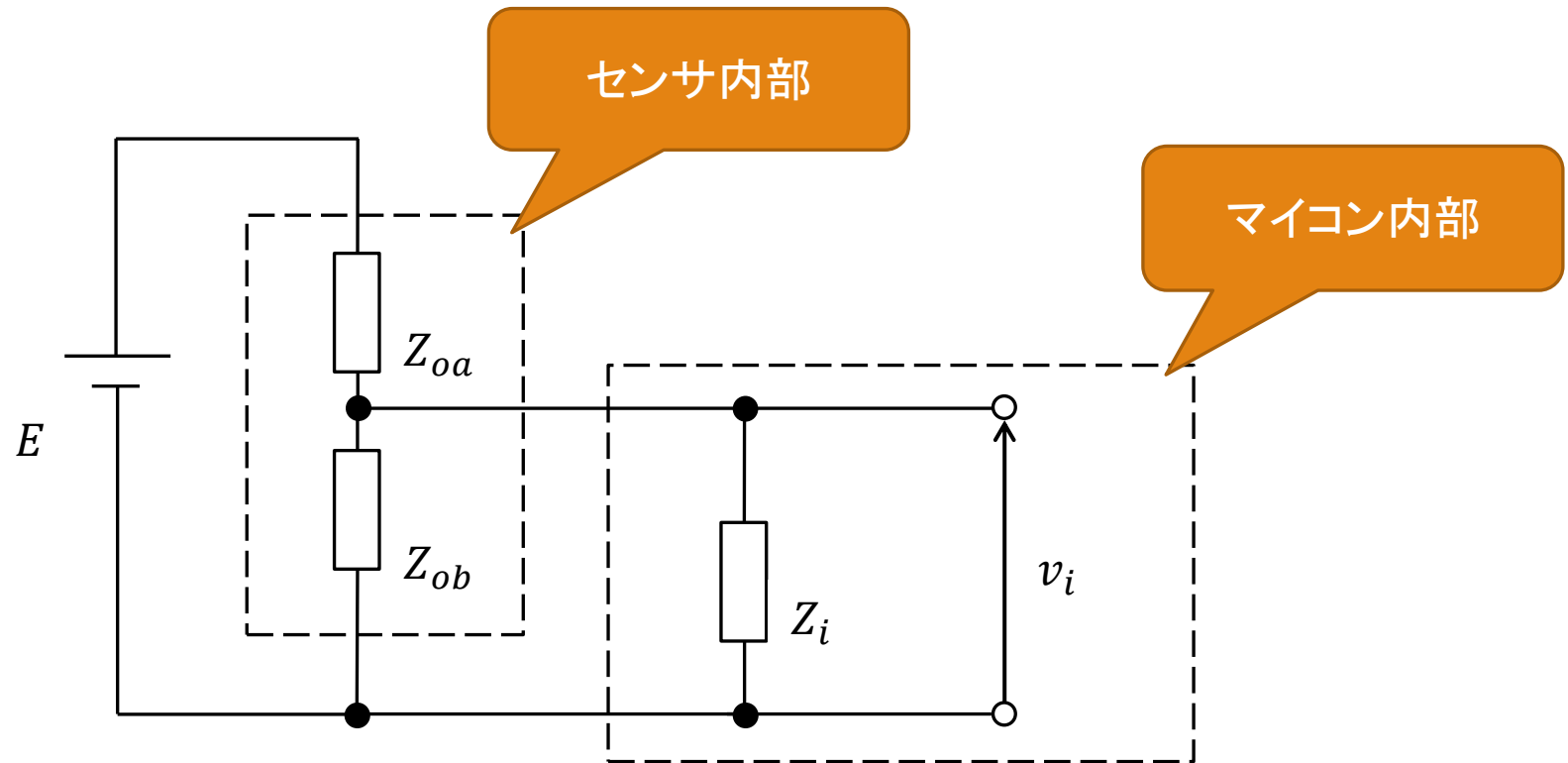
マイコンなど入力回路の入力インピーダンスが
センサの出力インピーダンスや配線等によるインピーダンスより
十分に大きければセンサの出力をほぼ正確に得ることができる。

しかし光学系のセンサ等では、
出力インピーダンスがとても大きく、
配線も長ければセンサの入力に影響を受ける。

そのような場合には、
インピーダンス変換を施さなければならない。

インピーダンス変換

イメージ

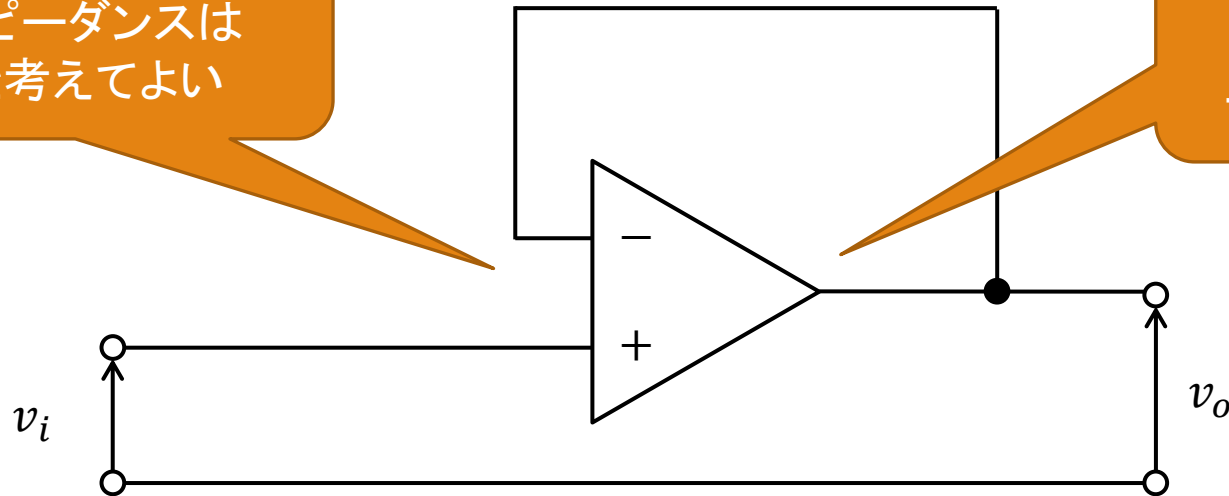


Z_i が Z_o より十分に大きければ、 v_i の影響が小さい

インピーダンス変換

インピーダンス変換を行う回路として、
ボルテージホロワと呼ばれる回路が存在する。

オペアンプの
入力インピーダンスは
無限大と考えてよい



オペアンプの
出力インピーダンスは
十分小さいと考えてよい

インピーダンス変換

センサからの出力を得るときには,

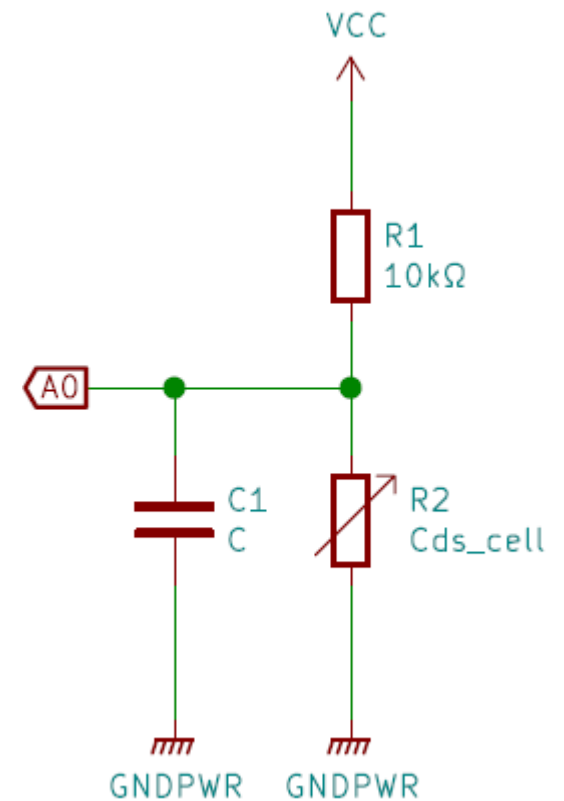
- ・センサの出力インピーダンスが十分に小さい
- ・計器(マイコン等)の入力インピーダンスが十分に大きい

ようにする必要がある.

CDSセルのフィルタ

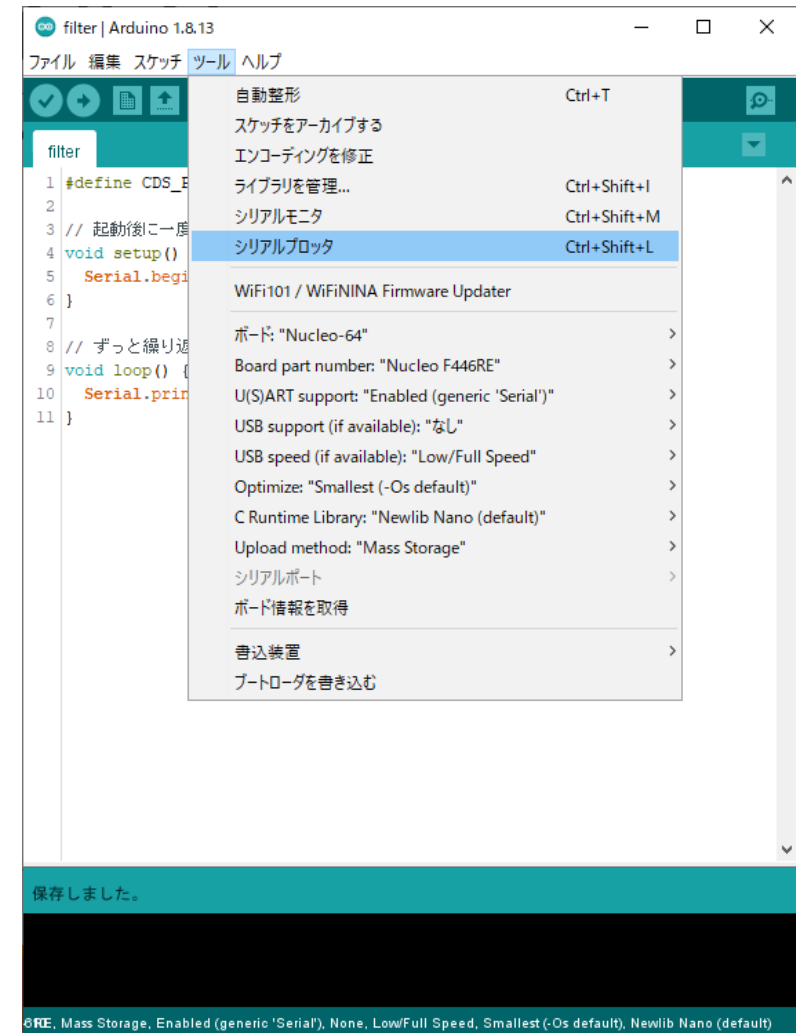
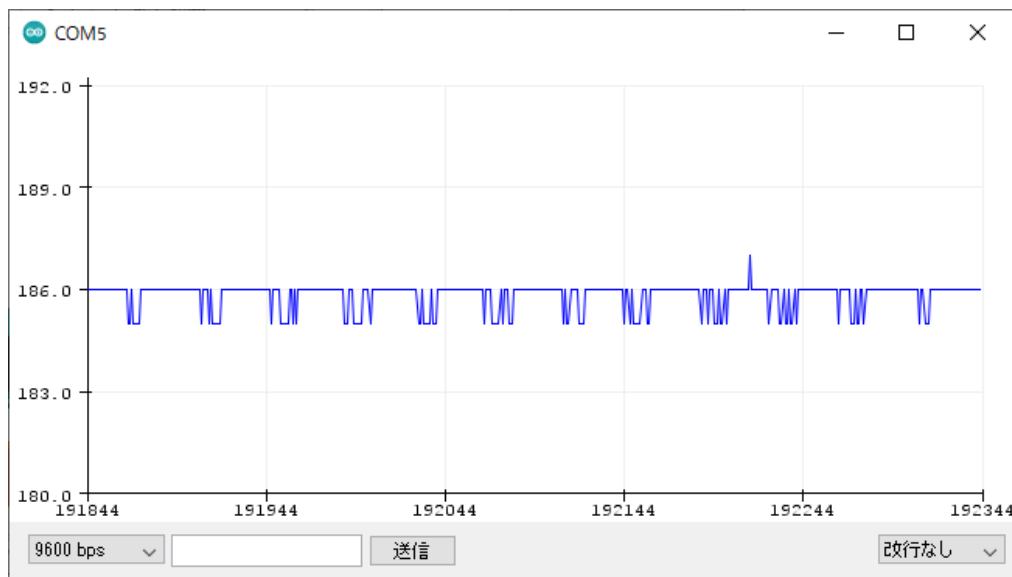
右の回路を実装して、出力を得てみましょう。
コンデンサを変更したり、外してみても、
出力がどのように変化するか観察してみましょう。

```
1 #define CDS_PIN A0           // Cdsセルが接続されているピン番号
2
3 // 起動後に一度だけ実行される関数
4 void setup() {
5     Serial.begin(9600);
6 }
7
8 // ずっと繰り返される関数
9 void loop() {
10     Serial.println(analogRead(CDS_PIN)); // Cdsセルの出力値を送信
11 }
```



CDSセルのフィルタ

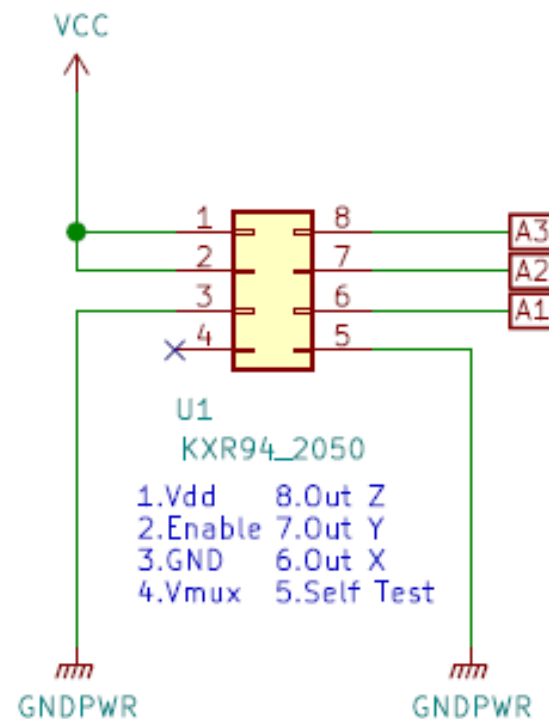
メニューバーの「ツール」⇒「シリアルプロッタ」を選択して、出力を可視化できる。



加速度センサ

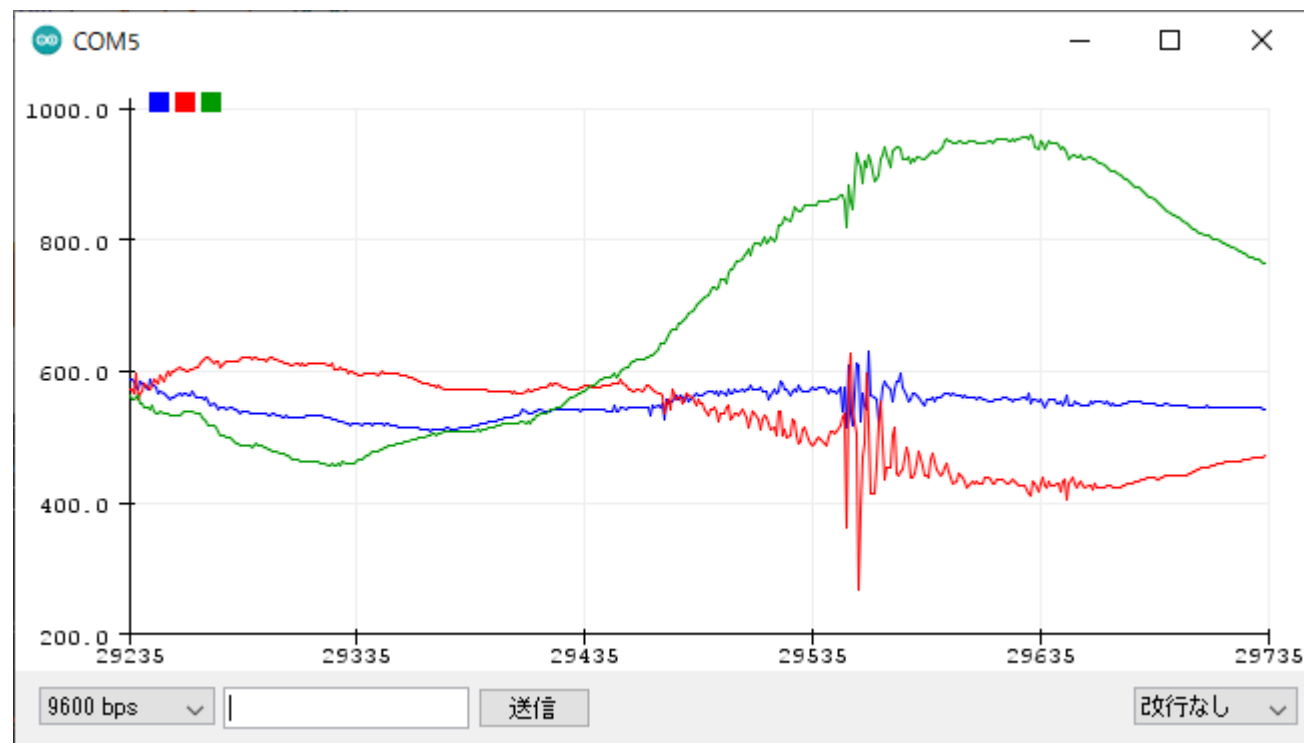
加速度センサは，力センサ等で慣性力を計測することによって，加速度を算出している．右の回路を実装して，プロットしてみましょう．

```
1 #define ACCELE_X_PIN A1 // 加速度センサ X軸出力ピン
2 #define ACCELE_Y_PIN A2 // 加速度センサ Y軸出力ピン
3 #define ACCELE_Z_PIN A3 // 加速度センサ Z軸出力ピン
4
5 void setup() {
6   Serial.begin(9600);
7
8   // 凡例表示
9   Serial.print(analogRead("ACCELE_X"));
10  Serial.print(",");
11  Serial.print(analogRead("ACCELE_Y"));
12  Serial.print(",");
13  Serial.print(analogRead("ACCELE_Z"));
14  Serial.println();
15 }
16
17 void loop() {
18   // 加速度センサの出力値を取得して送信
19   Serial.print(analogRead(ACCELE_X_PIN));
20   Serial.print(",");
21   Serial.print(analogRead(ACCELE_Y_PIN));
22   Serial.print(",");
23   Serial.print(analogRead(ACCELE_Z_PIN));
24   Serial.println();
25 }
```



加速度センサ

こんな感じに出力が得られればOK.
(振り回している状態)



演習

- ・ノートPCによっては、落下した時にハードディスクドライブのモータを停止させるようなシステムが組まれているものがある。一度、ボタンを押すとDCモータが回転し続けるようにし、重力方向に加速度センサが1秒以上動いたときにモータを停止させよ。なお、停止後にボタンを押すと復帰するものとする。
- ・PCに現在の加速度センサの3軸の速度[m/s]をスペース区切りでx軸, y軸, z軸の順に送信するシステムを作成せよ。ただし、PCから"V"を受信したタイミングで送信するものとする。

例) V \Rightarrow 0.00 1.14 1.14