

実用アナログ回路

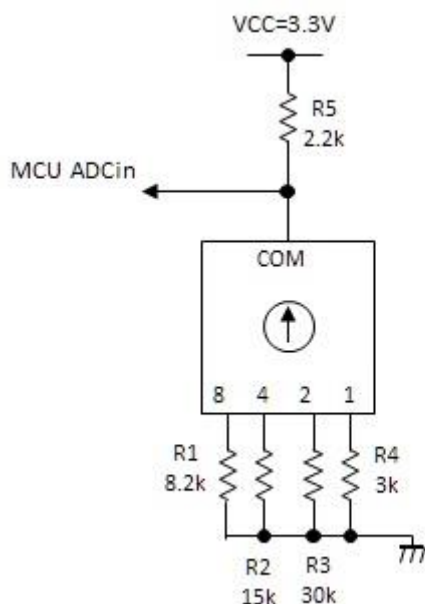
アナログ多値入力回路 2

前回は、アナログによる多値入力の基本例を紹介しました。本稿では、基本回路を応用発展して[ロータリーエンコーダ](#)のスイッチ回路を用いた任意に4桁とか5桁の桁の整数値を入力する例を紹介します。

桁数の大きな整数値の入力をするには前述したアナログ入力判定回路を応用として、10進数のBCDロータリーエンコーダを使用して桁数の大きな整数値の入力を行う例を紹介します。

ロータリーエンコーダのCOM端子とVCCの間にTop抵抗R5を接続し、4bitの各端子にR1,R2,R3,R4の抵抗を対GNDへ接続します。

これで、前述のラダー[抵抗](#) DACと同じ考え方でアナログ判定が可能になります。



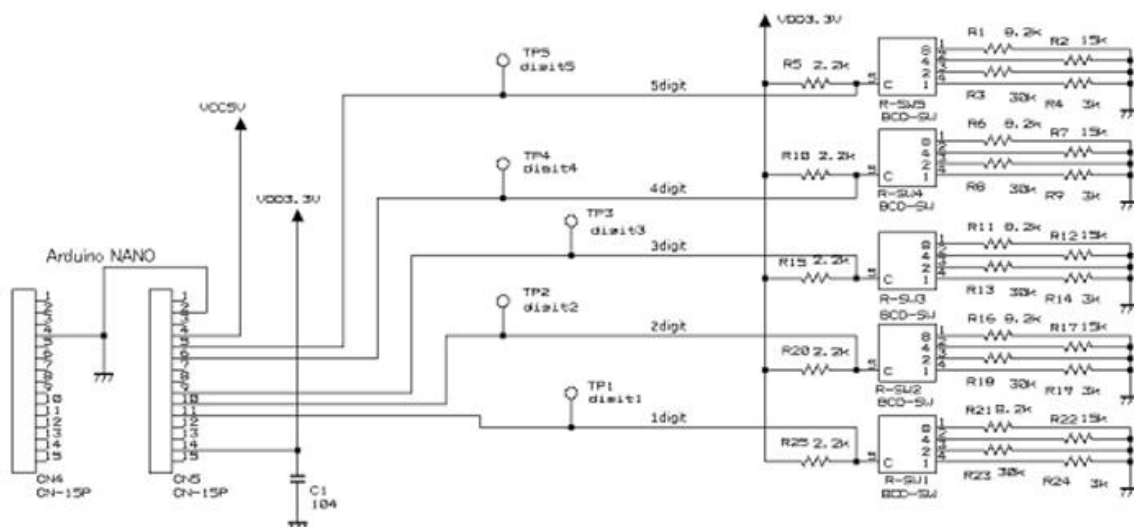
下表に 10 進数に応じたロータリーエンコーダの各ポジション位置での ADC へ入力される電圧値を示します。

	TopR (kΩ)	1	2	4	8	合成R (kΩ)	減衰比	期待値(V)	LLimit(V)	H.Limit(V)
		R4 (kΩ)	R3 (kΩ)	R2 (kΩ)	R1 (kΩ)					
	2.2	3	30	15	8.2	(kΩ)				
0	2.2						1.00	3.30	3.27	3.33
1	2.2	3				3.00	0.58	1.90	1.89	1.92
2	2.2		30			30.00	0.93	3.07	3.07	3.09
3	2.2	3	30			2.73	0.55	1.83	1.81	1.84
4	2.2			15		15.00	0.87	2.88	2.87	2.9
5	2.2	3		15		2.50	0.53	1.76	1.73	1.78
6	2.2		30	15		10.00	0.82	2.70	2.69	2.73
7	2.2	3	30	15		2.31	0.51	1.69	1.67	1.71
8	2.2				8.2	8.20	0.79	2.60	2.58	1.62
9	2.2	3			8.2	2.20	0.50	1.65	1.635	1.66

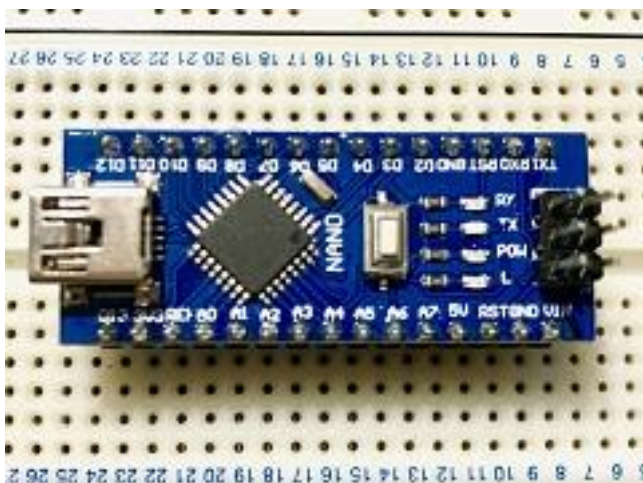
例えば、4 桁 (0~9999) の整数値入力回路を形成するには、回路を 4 個用意し、各々を 1 の位、10 の位、100 の位、1000 の位用に使用します。この場合、各桁用に ADC 入力が必要になりますので 4 本の ADC 入力端子を使用することになります。

BCD ロータリーエンコーダと MCU を使用した 5 桁数入力回路の実例

[MCU](#) に小型マイコンボードとして普及している [Arduino](#) シリーズの Arduino Nano を使用した回路例を示します。

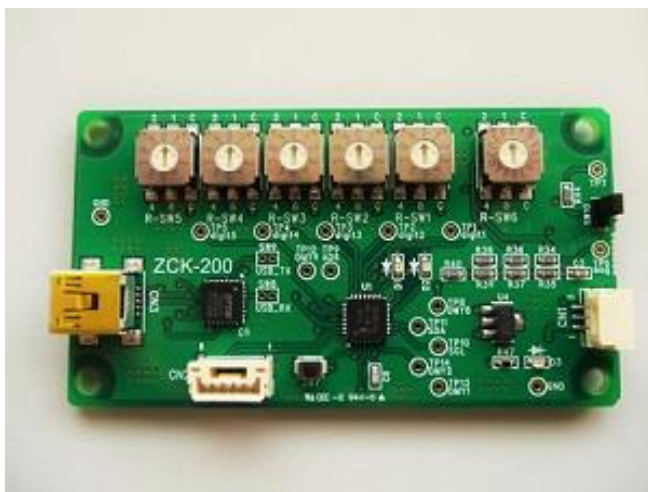


- 小型 AVR マイコンボード Arduino Nano



同回路を実際に組み込んだ例

●INEDLaboratry 社製 ZCK-200 基板



この 5 桁の数値化をソフトウェアで行う処理は、下記のようになります。

- 1 ロータリーエンコーダでの DAC 値を ADC で計測し 10 進数の値にします。
- 2 各桁毎の桁処理を行います。
この場合、0.1 の桁、1 の桁、10 の桁、100 の桁、1000 の桁と 5 つの数値を求め変数値に代入しておきます。
- 3 最終的に 5 つの変数値を総加算して、5 桁の整数値を求めます。

Arduino のプログラム例

参考例として上記回路で Arduino Nano で実行処理するプログラム例を下記に記載します。

==(↓ここから)=====

```
/**
 * 5digit Analog input
 * use Arduino Nano
 **/

int analogPin1 = 1;    // digit1 R-SW1 で生成の analog 値を読む端子 A1
int vin1 = 0;         // digit1 読み取った値を格納する変数

int analogPin2 = 2;    // digit2 R-SW2 で生成の analog 値を読む端子 A2
int vin2 = 0;         // digit2 読み取った値を格納する変数

int analogPin3 = 3;    // digit3 R-SW3 で生成の analog 値を読む端子 A3
int vin3 = 0;         // digit3 読み取った値を格納する変数

int analogPin4 = 6;    // digit4 R-SW4 で生成の analog 値を読む端子 A6
int vin4 = 0;         // digit4 読み取った値を格納する変数

int analogPin5 = 7;    // digit5 R-SW5 で生成の analog 値を読む端子 A7
int vin5 = 0;         // digit5 読み取った値を格納する変数

float ain1;
float ain2;
float ain3;
float ain4;
float ain5;

float dig1;
```

```
float dig2;
float dig3;
float dig4;
float dig5;
float dig;

float tout;
float duty;

void setup() {
}

void loop() {

    //digit1 1000 の桁
    vin1 = analogRead(analogPin1); // digit1 1000 の桁を読み取る
    ain1 = vin1*0.004714;

    if ((ain1 > 3.27)&&(ain1 < 3.33)) {
        dig1 = 0;
    }
    if ((ain1 > 1.89)&&(ain1 < 1.92)) {
        dig1 = 1000;
    }
    if ((ain1 > 3.07)&&(ain1 < 3.09)) {
        dig1 = 2000;
    }
}
```

```
if ((ain1 > 1.81)&&(ain1 < 1.84)) {
  dig1 = 3000;
}
if ((ain1 > 2.87)&&(ain1 < 2.9)) {
  dig1 = 4000;
}
if ((ain1 > 1.73)&&(ain1 < 1.78)) {
  dig1 = 5000;
}
if ((ain1 > 2.69)&&(ain1 < 2.73)) {
  dig1 = 6000;
}
if ((ain1 > 1.67)&&(ain1 < 1.71)) {
  dig1 = 7000;
}
if ((ain1 > 2.6)&&(ain1 < 2.62)) {
  dig1 = 8000;
}
if ((ain1 > 1.635)&&(ain1 < 1.66)) {
  dig1 = 9000;
}

//digit2 100 の桁
vin2 = analogRead(analogPin2); // digit2 100 の桁を読み取る
ain2 = vin2*0.004714;

if ((ain2 > 3.27)&&(ain2 < 3.33)) {
  dig1 = 0;
}
```

```
if ((ain2 > 1.89)&&(ain2 < 1.92)) {
    dig2 = 100;
}
if ((ain2 > 3.07)&&(ain2 < 3.09)) {
    dig2 = 200;
}
if ((ain2 > 1.81)&&(ain2 < 1.84)) {
    dig2 = 300;
}
if ((ain2 > 2.87)&&(ain2 < 2.9)) {
    dig2 = 400;
}
if ((ain2 > 1.73)&&(ain2 < 1.78)) {
    dig2 = 500;
}
if ((ain2 > 2.69)&&(ain2 < 2.73)) {
    dig2 = 600;
}
if ((ain2 > 1.67)&&(ain2 < 1.71)) {
    dig2 = 700;
}
if ((ain2 > 2.6)&&(ain2 < 2.62)) {
    dig2 = 800;
}
if ((ain2 > 1.635)&&(ain2 < 1.66)) {
    dig2 = 900;
}

//digit3 10 の桁
```



```
vin3 = analogRead(analogPin3); // digit3 10 の桁を読み取る
ain3 = vin3*0.004714;

if ((ain3 > 3.27)&&(ain3 < 3.33)) {
  dig3 = 0;
}
if ((ain3 > 1.89)&&(ain3 < 1.92)) {
  dig3 = 10;
}
if ((ain3 > 3.07)&&(ain3 < 3.09)) {
  dig3 = 20;
}
if ((ain3 > 1.81)&&(ain3 < 1.84)) {
  dig3 = 30;
}
if ((ain3 > 2.87)&&(ain3 < 2.9)) {
  dig3 = 40;
}
if ((ain3 > 1.73)&&(ain3 < 1.78)) {
  dig3 = 50;
}
if ((ain3 > 2.69)&&(ain3 < 2.73)) {
  dig3 = 60;
}
if ((ain3 > 1.67)&&(ain3 < 1.71)) {
  dig3 = 70;
}
if ((ain3 > 2.6)&&(ain3 < 2.62)) {
  dig3 = 80;
}
```

```
}  
if ((ain3 > 1.635)&&(ain3 < 1.66)) {  
    dig3 = 90;  
}  
  
//digit4 1 の桁  
vin4 = analogRead(analogPin4); // digit4 1 の桁を読み取る  
ain4 = vin4*0.004714;  
  
if ((ain4 > 3.27)&&(ain4 < 3.33)) {  
    dig4 = 0;  
}  
if ((ain4 > 1.89)&&(ain4 < 1.92)) {  
    dig4 = 1;  
}  
if ((ain4 > 3.07)&&(ain4 < 3.09)) {  
    dig4 = 2;  
}  
if ((ain4 > 1.81)&&(ain4 < 1.84)) {  
    dig4 = 3;  
}  
if ((ain4 > 2.87)&&(ain4 < 2.9)) {  
    dig4 = 4;  
}  
if ((ain4 > 1.73)&&(ain4 < 1.78)) {  
    dig4 = 5;  
}  
if ((ain4 > 2.69)&&(ain4 < 2.73)) {  
    dig4 = 6;  
}
```

```
}  
if ((ain4 > 1.67)&&(ain4 < 1.71)) {  
    dig4 = 7;  
}  
if ((ain4 > 2.6)&&(ain4 < 2.62)) {  
    dig4 = 8;  
}  
if ((ain4 > 1.635)&&(ain4 < 1.66)) {  
    dig4 = 9;  
}  
  
//digit5 0.1 の桁  
vin5 = analogRead(analogPin5); // digit5 0.1 の桁を読み取る  
ain5 = vin5*0.004714;  
  
if ((ain5 > 3.27)&&(ain5 < 3.33)) {  
    dig5 = 0;  
}  
if ((ain5 > 1.89)&&(ain5 < 1.92)) {  
    dig5 = 0.1;  
}  
if ((ain5 > 3.07)&&(ain5 < 3.09)) {  
    dig5 = 0.2;  
}  
if ((ain5 > 1.81)&&(ain5 < 1.84)) {  
    dig5 = 0.3;  
}  
if ((ain5 > 2.87)&&(ain5 < 2.9)) {  
    dig5 = 0.4;
```

```
}  
if ((ain5 > 1.73)&&(ain5 < 1.78)) {  
    dig5 = 0.5;  
}  
if ((ain5 > 2.69)&&(ain5 < 2.73)) {  
    dig5 = 0.6;  
}  
if ((ain5 > 1.67)&&(ain5 < 1.71)) {  
    dig5 = 0.7;  
}  
if ((ain5 > 2.6)&&(ain5 < 2.62)) {  
    dig5 = 0.8;  
}  
if ((ain5 > 1.635)&&(ain5 < 1.66)) {  
    dig5 = 0.9;  
}  
  
dig = dig1 + dig2 + dig3 + dig4 + dig5;  
  
}  
  
==(↑ここまで)=====
```