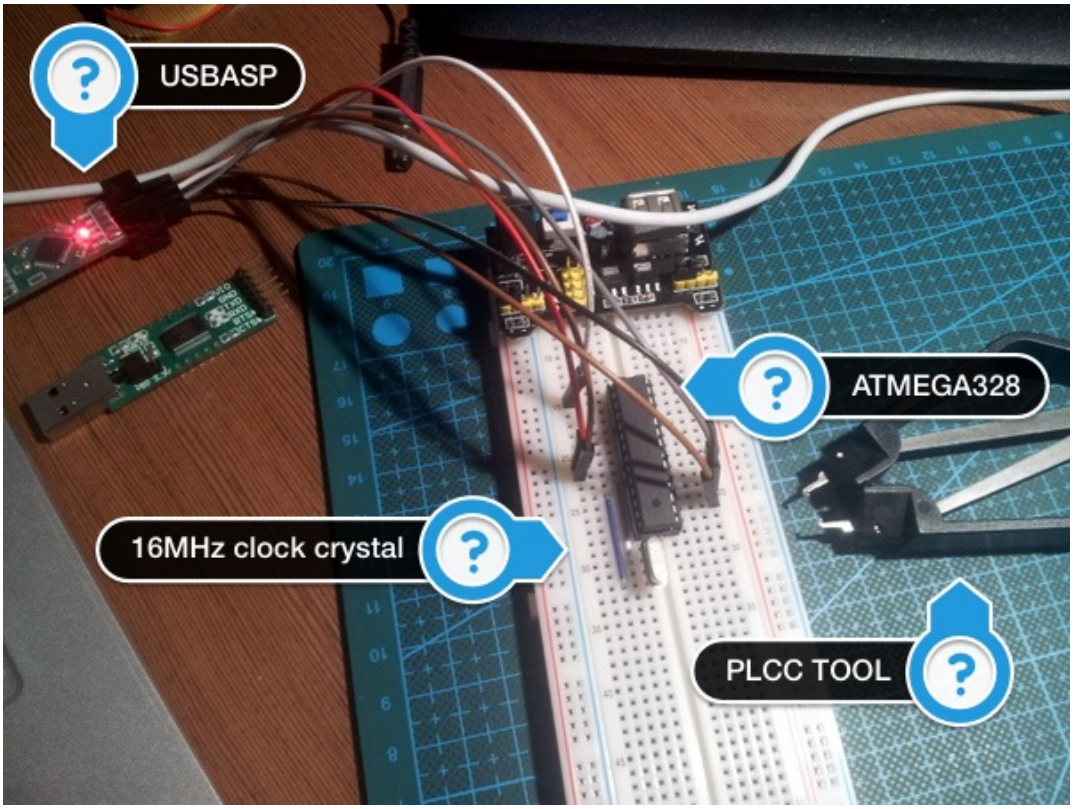
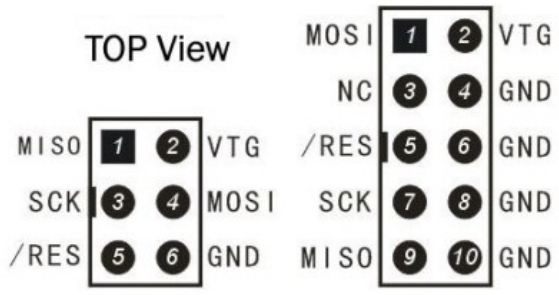
voltage display

2014-12-07 XnnYygn

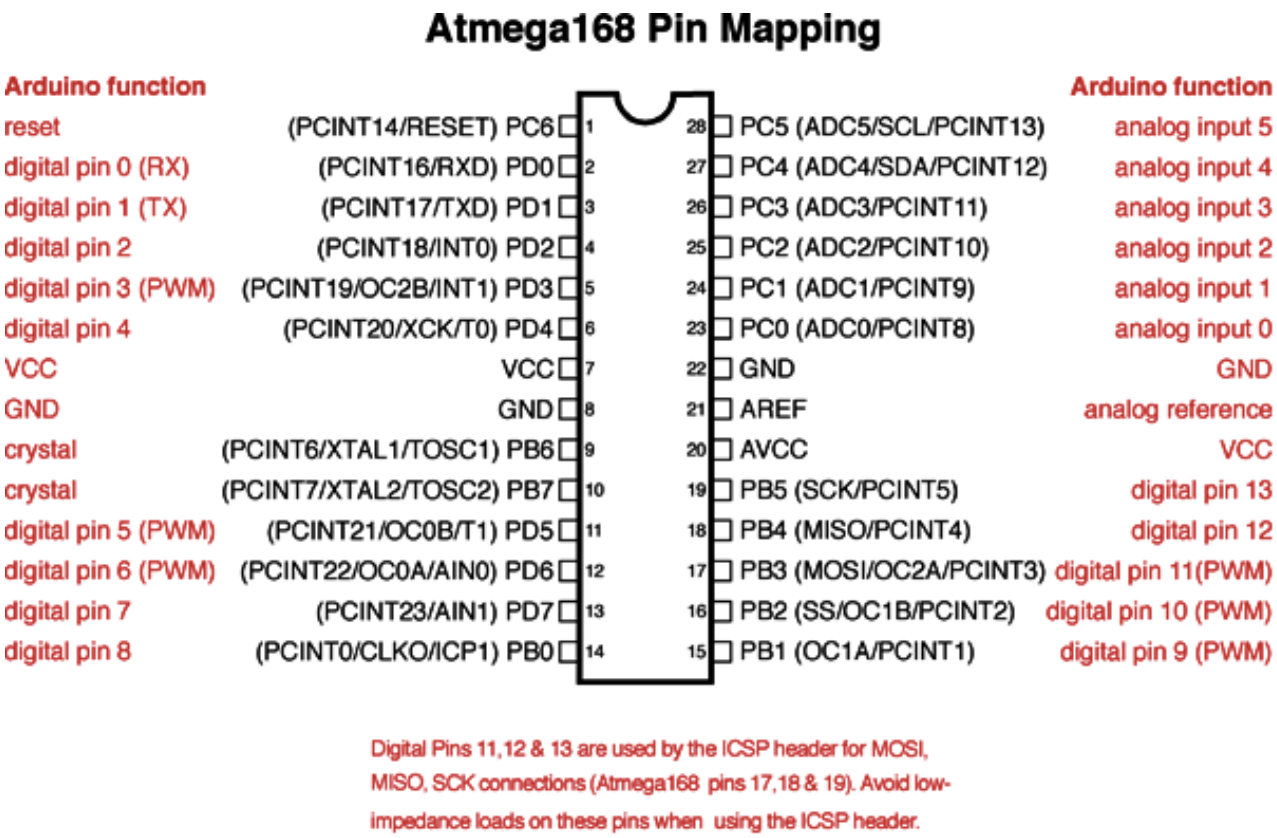
ここまでは、一般的な腕があるんでしょう。今回Arudino抜きで直接AVRの集成電路を利用しました。
そう、元々電子工作と言うのはArdunioだけではないでしょう。Arduinoのサイトでも自分がAVR開発ボードを言いました。(http://playground.arduino.cc/Main/AVR)
Arduino UNOの中心はATMega328P-PUです。今回使ったのはDIPの方です。（SMDの方に比べて）



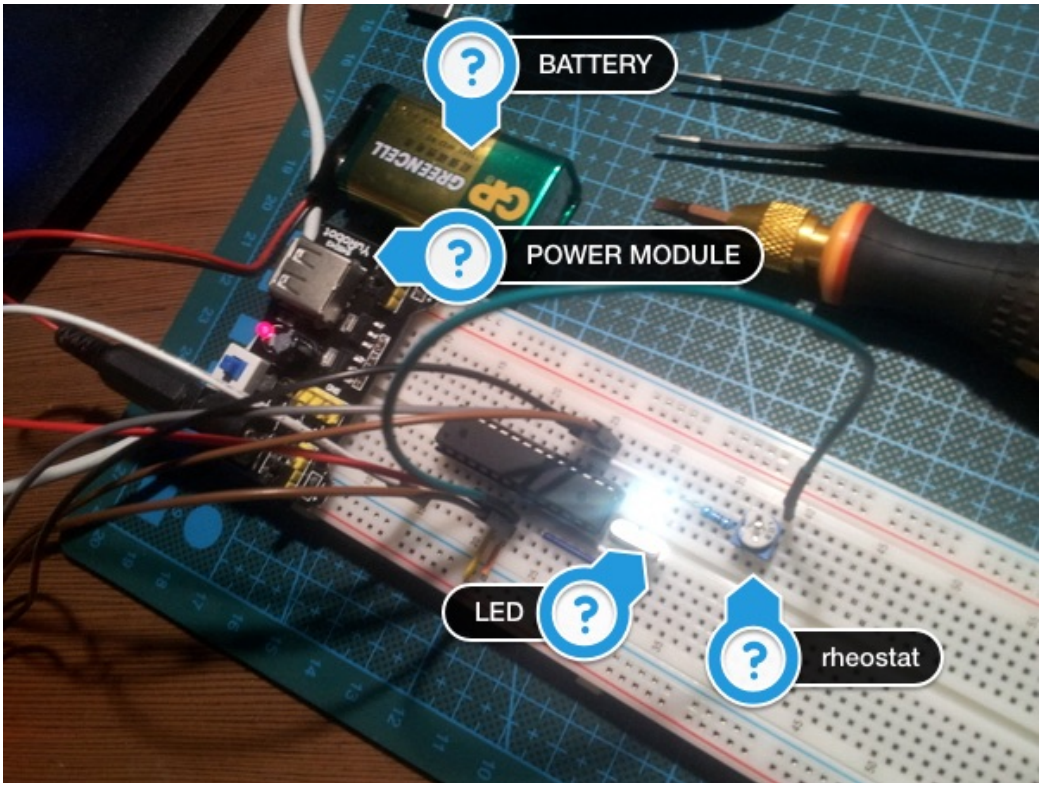
USBASPを使って、BOOTLOADERをArudino IDEでアップロードしてみました。
clock crystalは必要です。ないならアップロードできません。



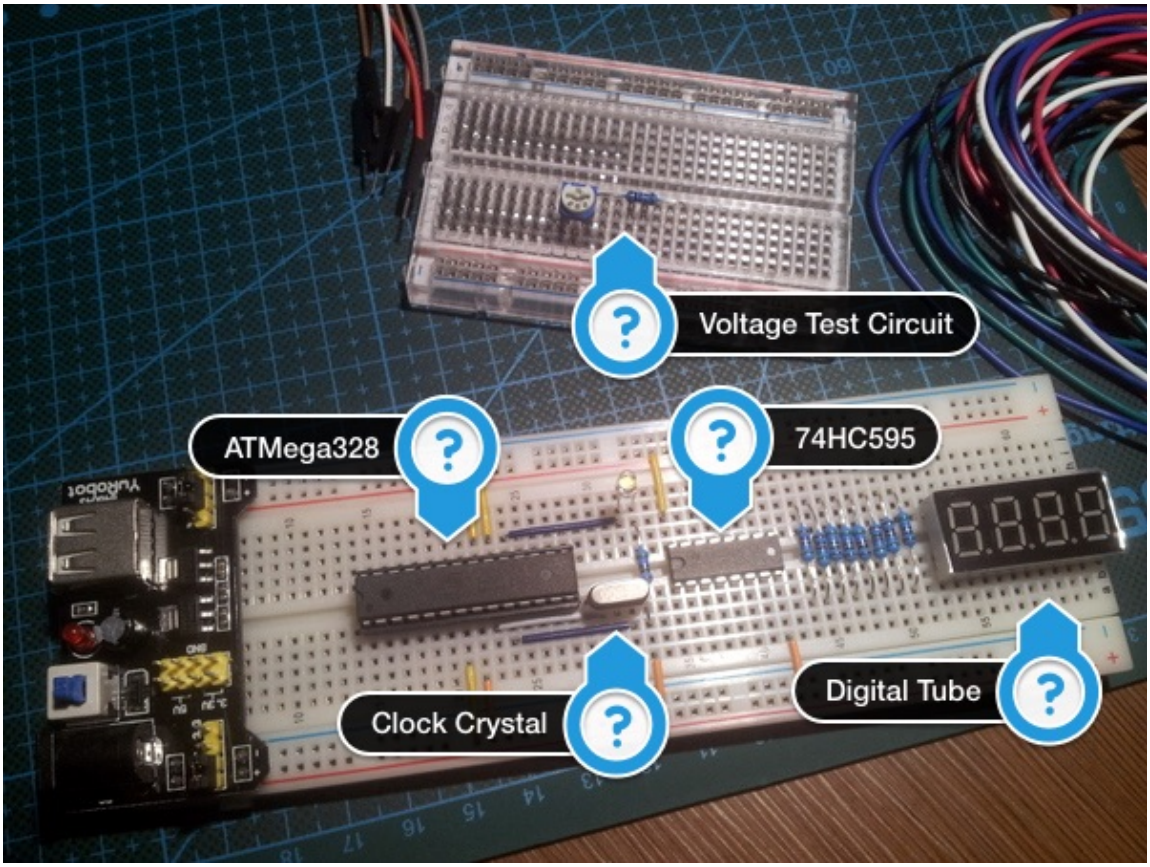
これはUSBASPのPINです（右）。使ったのは左と同じ六のPINしかありません。



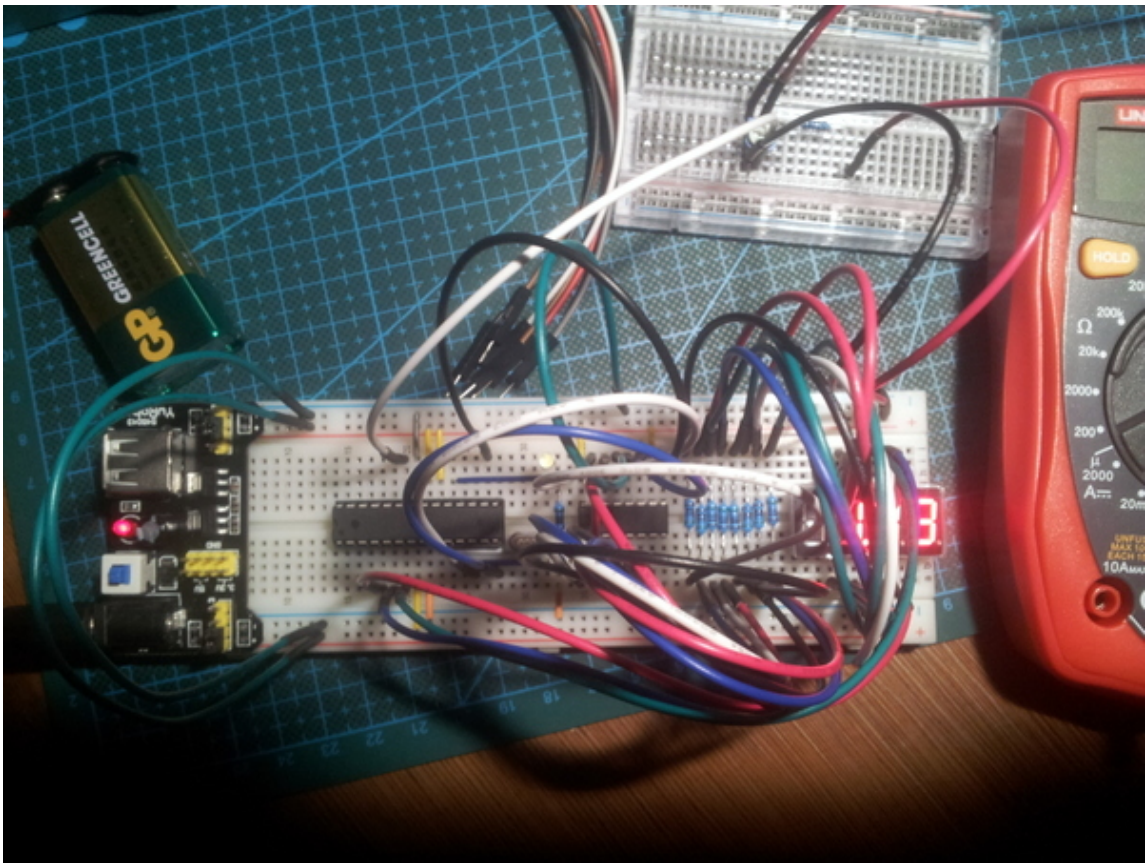
これはATMega168の図ですが、ATMega328とほぼ同じです。
BOOTLOADERをアップロードしたら、後でFT232RLなどでアップロードするんですか。いいえ、今回は全部直接USBASPでアップロードします。もしFT232RLでアップロードすると、RESETボタンは必要です。「アップロードと同時RESETボタンを押す」とかちょっと面倒です。これもArudinoが流行ってるわけの一つでしょう。
D13のLEDのテスト。なんか眩るしい。



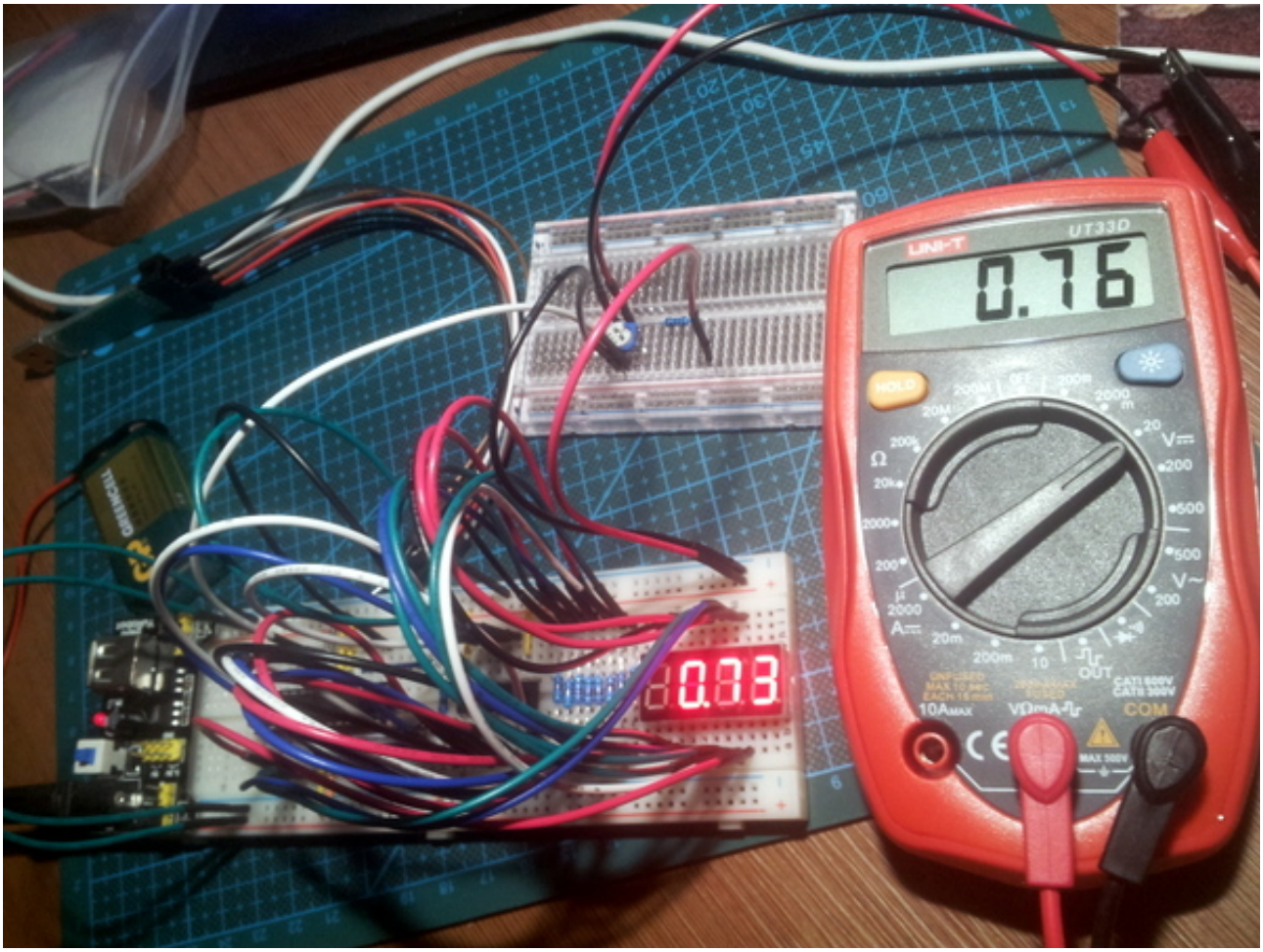
これからは作品を作る時間です。



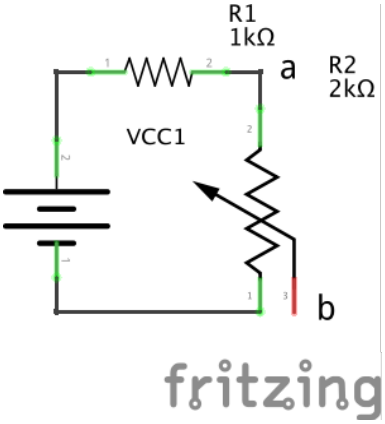
これは主な部品です。次はラインつきの写真です。



製品です。



まず上の電圧の電路を紹介します。



測るのはaとbの間の電圧です。

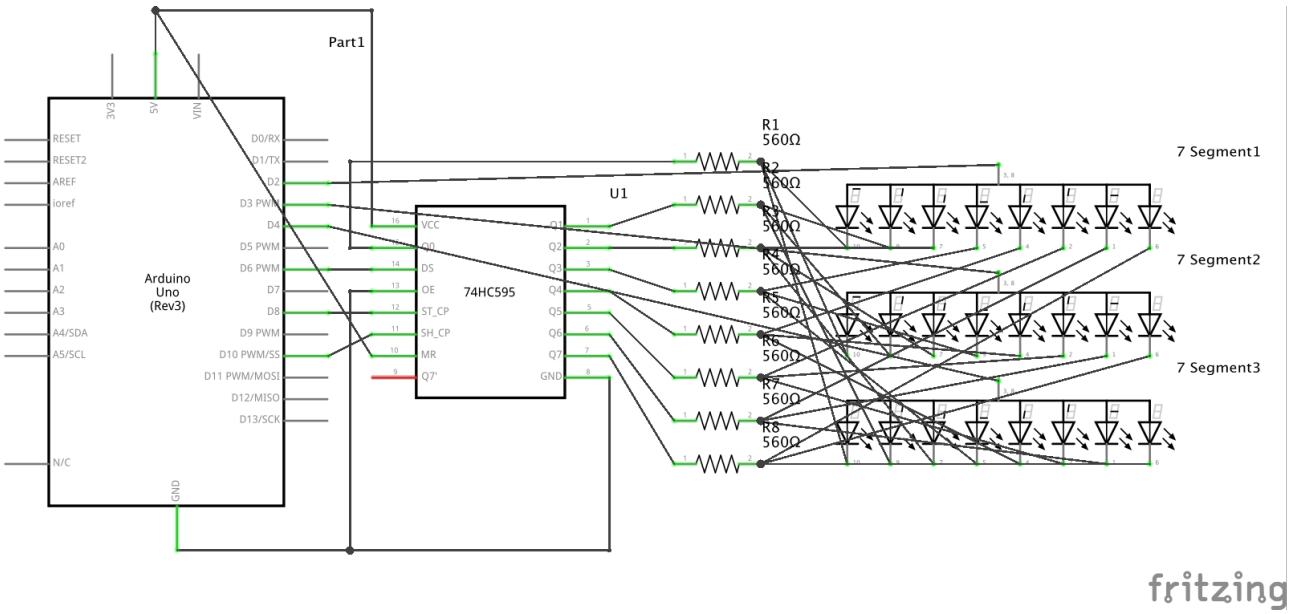
次は数字を表す部分です。普通の8-segment LED displayはハのPINがあります。でもこの部品は４つの数字はあるのに、ただ12のPINがあります。直接全部のLEDをコントロールするのはできません、4 x 8, 32のPINが必要です。でも何で12なの? 実は8 + 4で計算されました。４つのPINはどの数字をコントロールし、残った８つのPINはこの数字のLEDをコントロールします。実際12のPINそんなに多いではないが、毎回12のPINをコントロールのは面倒なので、74HC595という集成電路を使いました。でも、数字を選ぶ４つのPINは直接コントロールします。
次はどのようで表示できます。毎回一つの数字を選んで、この数字をコントロールします。でも、毎回４つの数字を表すんでしょう。だから、ちょっとネタみたいなものの出番です。

確かに毎回４つの数字を表して、そしてこの作品は実際毎回一つの数字しかコントロールできません。では、４つの数字を別々で表します。これ、すぐばれるんでしょう。でも、もしとても早いならどうでしょう。実際別々の数字を表すのは見られません。コードはこちらです。

```
28 void loop() {
29   // read voltage
30   int analogValue = analogRead(0);
31   float voltage = 0.0048 * analogValue;
32   byte numbers = voltage * 100;
33
34   for(byte i = 0; i < 20; i++) {
35     displayNumber(numbers / 100, true, PIN_D_2);
36     displayNumber((numbers % 100) / 10, false, PIN_D_3);
37     displayNumber(numbers % 10, false, PIN_D_4);
38   }
39 }
40
41 void displayNumber(byte n, bool withPoint, byte pin) {
42   digitalWrite(pin, LOW);
43   digitalWrite(PIN_LATCH, LOW);
44   shiftOut(PIN_DATA, PIN_CLOCK, LSBFIRST, digits[n] + (withPoint ? 1 : 0));
45   digitalWrite(PIN_LATCH, HIGH);
46   delay(10);
47   digitalWrite(pin, HIGH);
48 }
```

ここの delay(10)は調整した結果です。うえのLOOPを気にしないでください。それはただ電圧の読むのはスピードを遅くされたいです。
digitsは計算した結果です。

byte digits[] = {252, 96, 218, 242, 102, 182, 190, 224, 246, 238};



これは電路です。

実際AVRの集成電路でものを作ると直接Arudinoで作るは別に大きな違うところはないと思います。でもArudinoはほかの機能たとえば安全措置はあります。直接より人を安心させるんでしょう。最初はArudinoでテストして、後で直接ATMega328など集成電路にアップロードするとはいいと思います。これも一般的に研究から実際応用する流れでしょう。