

## Une alimentation numérique CC



par Guido Socher ([homepage](#))

### L'auteur:

Guido adore Linux parce qu'il est un vraiment bon système pour développer votre propre matériel.

*Traduit en Français par:*  
Jean-Etienne Poirrier  
([homepage](#))



### Résumé:

Une alimentation électrique en courant continu (CC) fiable est un périphérique important pour une passion comme l'électronique. Une alimentation électrique stabilisée électroniquement peut facilement coûter plus de 150 Euros.

En utilisant un plan basé sur un microcontrôleur intelligent, nous pouvons construire une alimentation électrique qui a plus de fonctionnalités et qui est beaucoup moins chère.

Les lecteurs, pour qui tout ceci serait nouveau, devraient d'abord lire [Programmer le microcontrôleur AVR avec GCC, libc 1.0.4](#).

Ne vous inquiétez pas si le circuit imprimé semble trop compliqué pour votre laboratoire à la maison. Toutes les pièces et des PCB pré-imprimés seront, encore une fois, disponibles sur [shop.tuxgraphics.org](http://shop.tuxgraphics.org).

## Introduction

En 2002, j'ai écrit un article pour [linuxfocus.org](http://linuxfocus.org) sur une alimentation électrique se basant sur un microcontrôleur ([article251 du numéro de novembre 2002](#)). L'article a reçu beaucoup d'intérêt, comme j'ai pu le remarquer des courriels que j'ai reçu à son sujet. Le plan de cette alimentation électrique était cependant difficile pour les fans, dont l'électronique reste une passion, dû à la complexité du circuit.

Ainsi, voici maintenant un nouveau plan, moins complexe, mais qui possède plus de fonctionnalités.

1. Seulement des composants pas chers et standards sont utilisés
2. Seulement une source de courant est nécessaire (pas de tension négative pour des amplificateurs opérationnels)
3. L'affichage montre les valeurs prédéfinies et actuelles de la tension et du courant en même temps.

4. Vous pouvez complètement contrôler l'alimentation électrique via votre ordinateur ou vous pouvez l'utiliser seule.
5. Elle est très petite et très puissante.

Comment a-t-il été possible d'enlever des composants et d'ajouter plus de fonctionnalités ? Le truc est de déplacer les fonctionnalités qui étaient précédemment basées sur des composants analogiques (comme les amplificateurs opérationnels) dans le microcontrôleur. En d'autres termes, la complexité du logiciel et des algorithmes est plus élevée mais la complexité du matériel est réduite. Cela réduit la complexité de l'ensemble pour vous puisque le logiciel peut être juste copié. Le fait que l'Atmega8 puisse faire beaucoup plus que le microcontrôleur AT90S4433, maintenant dépassé, a aidé également.

Dans cet article, vous apprendrez aussi de nouvelles choses sur l'Atmega8. Vous avez vu comment utiliser le convertisseur analogique-digital. Nous allons ajouter un convertisseur digital-analogique très rapide qui pourra être utile pour beaucoup d'autres circuits.

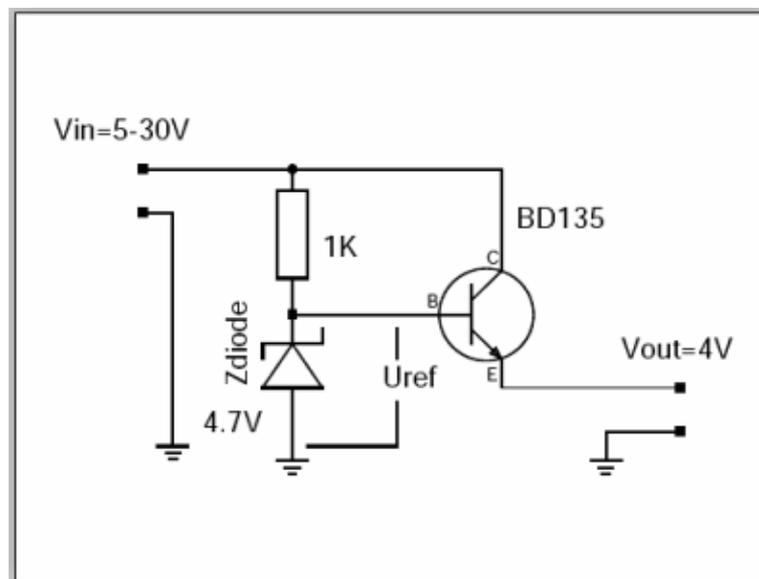
Dans cet article, nous allons discuter du plan matériel et électrique. Il y aura ensuite deux articles supplémentaires où j'expliquerai le logiciel. Nous allons aussi ajouter plus de fonctionnalités. Cependant, les ajouter ne nécessitera que des changements logiciels.

## L'idée de base du plan électrique

Je pourrais bien sûr juste vous donner le diagramme du circuit et quelques instructions sur la manière de le construire. Cela fonctionnera probablement mais vous ne saurez pas comment cela fonctionne. La plupart des kits d'électronique de hobby sont comme ça.

L'idée du magasin en ligne [shop.tuxgraphics.org](http://shop.tuxgraphics.org) est cependant de fournir de la bonne documentation gratuite accompagnant les kits de développement. Dans la documentation (par exemple, cet article), nous expliquons réellement comment les choses fonctionnent et pourquoi elles ont été conçues de cette manière. Je trouve beaucoup plus amusant de construire quelque chose que je comprends et j'espère que vous l'appréciez aussi.

Commençons par l'alimentation électrique stabilisée électroniquement la plus simple possible. Elle consiste en 2 parties de base : un transistor et une référence de tension générée avec une diode Z.



La tension de sortie de ce circuit est  $U_{ref} - 0.7V$ . Les  $0.7V$  sont approximativement la chute de tension que vous avez entre B et E sur le transistor. La diode Z et la résistance génèrent une tension de référence qui est stable, même si l'entrée fluctue et est bruitée. Le transistor est nécessaire pour gérer des courants plus élevés que la diode Z et la résistance seuls ne peuvent fournir. Dans cette configuration, le transistor amplifie juste le courant. Le courant que la résistance et la diode Z doivent fournir est le courant de sortie divisé par  $h_{fe}$  ( $h_{fe}$  est un nombre que vous pouvez trouver en regardant dans la feuille de spécifications du transistor).

Quels sont les problèmes avec ce circuit ?

- Le transistor va mourir lorsqu'il y aura un court-circuit dans la sortie
- Il ne fournit qu'une tension de sortie fixe

Ce sont des limitations assez importantes qui rendent ce circuit inutilisable mais ce circuit est toujours le bloc de construction de base de toutes les alimentations électriques régulées électroniquement.

Pour surmonter ces problèmes, vous avez besoin d'une certaine « intelligence » qui va réguler le courant sur la sortie et une tension de référence variable. C'est tout (... et cela rend le circuit beaucoup plus complexe).

Pendant les dernières décennies, les gens ont utilisé des amplificateurs opérationnels pour fournir cette intelligence. Les amplificateurs opérationnels peuvent, au fond, être utilisés comme des calculateurs analogiques pour additionner, soustraire, multiplier ou faire des « ou » logiques sur des tensions et des courants.

De nos jours, les microcontrôleurs sont si rapides que tout cela peut être réalisé facilement par logiciel. La beauté est que vous obtenez également un voltmètre et un ampèremètre gratuitement, comme effet secondaire. La boucle de contrôle dans les microcontrôleurs doit connaître les valeurs de tension et de courant, d'une façon ou d'une autre. Vous n'avez qu'à l'afficher. Ce dont vous avez besoin de la part du microcontrôleur, c'est :

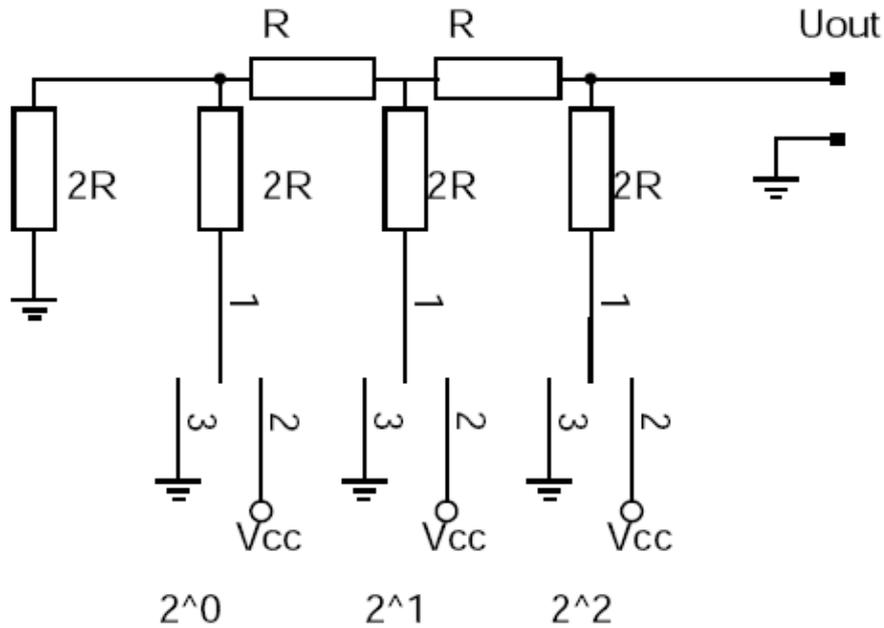
- un convertisseur A-D pour mesurer la tension
- un convertisseur D-A pour envoyer les commandes à notre transistor de puissance (qui fournit la tension de référence)

Le problème est que le convertisseur D-A a besoin d'être très rapide. S'il y a un court-circuit détecté dans la sortie, alors il doit immédiatement réduire la tension à la base du transistor sinon il mourra. Rapide signifie en quelques millisecondes (aussi rapide qu'un amplificateur opérationnel).

L'Atmega8 possède un convertisseur A-D qui est plus que rapide mais il n'a pas, à première vue, de convertisseur D-A. Il est possible d'utiliser une modulation en largeur d'impulsion et un filtre passe-bas analogique pour obtenir un convertisseur D-A mais elle reste beaucoup trop lente pour implémenter la protection contre les court-circuits dans le logiciel. Comment construire un convertisseur D-A rapide ?

## L'échelle R-2R

Il y a de nombreuses manières de construire un convertisseur digital-analogique mais nous en avons besoin d'un, rapide, pas cher et qui peut facilement s'interfacer avec notre microcontrôleur. Il existe un convertisseur D-A connu sous le nom « d'échelle R-2R ». Il consiste en des résistances et des interrupteurs seulement. Il y a deux types de résistances. Une avec R et une avec deux fois la valeur de R.



$$U_{out} = Z * (V_{cc}/(Z_{max} + 1))$$

La figure ci-dessus montre un convertisseur D-A R2R 3 bits. La logique de contrôle déplace les interrupteurs entre GND et Vcc. Un « 1 » numérique connecte l'interrupteur à Vcc et un « zéro » numérique à GND. Que fait ce circuit ? Il fournit des tensions par pas de  $V_{cc}/8$ . En général, la tension de sortie est  $Z * (V_{cc}/(Z_{max}+1))$  où Z est le nombre numérique. Dans le cas du convertisseur A-D 3 bits, c'est 0-7.

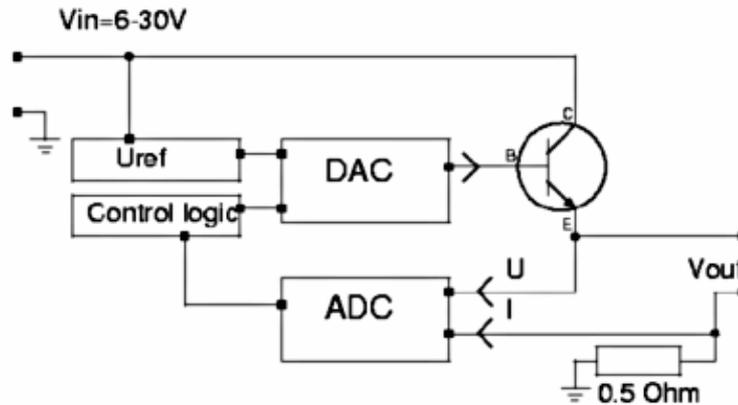
La résistance interne du circuit est vue comme la sortie de R.

Au lieu d'utiliser des interrupteurs séparés, nous pouvons connecter l'échelle R-2R aux lignes de sortie du microcontrôleur. Une broche de sortie de l'Atmega8 peut fournir environ 10mA mais, à ce courant, vous notez déjà une chute de tension. Nous souhaiterions utiliser la gamme totale de sortie 0-5V de manière à ce que la charge sur la sortie soit moindre qu'1 mA. En d'autres termes, nous construisons une échelle R-2R avec des résistances de 5K et 10K.

Le convertisseur A-D de l'Atmega8 a une résolution de 10 bits. Pour utiliser complètement cette résolution, nous avons également besoin de 10 bits pour le convertisseur D-A. En d'autres termes, nous avons besoin de trouver 10 broches de sortie qui ne sont pas utilisées par rien d'autre. C'est un petit défi puisque nous souhaitons également avoir un clavier, un affichage LCD et une interface sérielle I2C vers le PC mais l'Atmega8 est assez bon. Tout entre dedans.

## Un plan plus détaillé

Ainsi, voici un plan plus détaillé du circuit ci-dessus.



Je ne veux pas vous décourager mais c'est malheureusement toujours un circuit qui n'est pas utilisable. Cependant, il est très intéressant pour comprendre l'idée qui se trouve derrière le circuit final. Mais qu'est-ce qui ne va pas avec ce circuit ? Il y a principalement deux choses :

- Le DAC (digital to analog converter, convertisseur digital vers analogique) ne peut pas fournir le courant pour conduire le transistor de puissance.
- Le microcontrôleur opère à 5V ; donc, la sortie maximum du DAC est 5V, ce qui signifie que la tension de sortie maximale derrière le transistor de puissance sera  $5 - 0.7 = 4.3V$ .

Pour résoudre cela, nous devons ajouter des amplificateurs pour le courant et la tension.

## Le circuit final

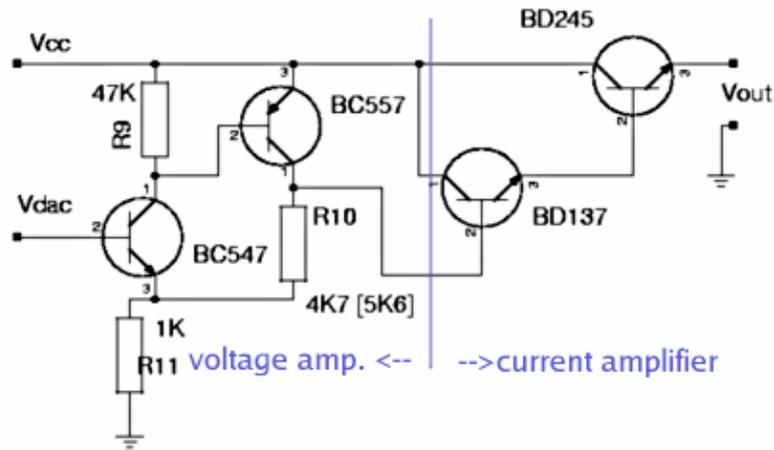
Lorsqu'on ajoute des amplificateurs, nous devons garder à l'esprit qu'ils doivent fonctionner avec des signaux larges. La plupart des plans d'amplificateurs (par exemple, pour l'audio) sont fait avec la supposition que les signaux seront petits comparés à la tension fournie. Donc, oubliez tous les livres classiques sur la création de transistor amplificateurs.

Nous pourrions utiliser les amplificateurs opérationnels mais ces derniers requièrent qu'on fournisse des tensions positive et négatives supplémentaires, ce que nous cherchons à éviter.

L'amplificateur nous impose également la condition supplémentaire de partir d'une tension fournie de zéro jusqu'à un état stable sans générer de pic de sortie. En d'autres termes, il ne doit y avoir aucune oscillation courte ou pic de sortie lorsque vous allumer l'alimentation électrique.

Que faire ? Eh bien, ce n'est pas une tâche facile du tout et elle requièrent une petite expérience. Je vais parcourir avec vous le circuit final et effectuer les calculs nécessaires.

Nous partons avec le transistor de puissance. Selon la feuille de spécifications, le BD245B a un  $h_{fe} = 20$  à 3A de sortie. Il va ainsi tirer environ 15mA sur la Base. Pour amplifier le courant (réduire le courant nécessaire pour conduire le circuit), nous utilisons une configuration connue sous le nom de « Darlington transistor ». Pour cela, nous plaçons un BD137 devant. Il a une valeur  $h_{fe}$  de 50–100. Cela réduira le courant nécessaire à moins de 3mA ( $150mA / 50$ ). 3 mA, c'est gérable avec des petits transistors de signal comme le BC547/BC557. Il existe aussi des transistors pour petite tension qui sont très bons pour construire des amplificateurs de tension.



Pour une sortie de 30V, nous devons au-moins amplifier les 5V du DAC d'un facteur de 6. Pour cela, nous combinons un transistor PNP et un NPN comme montré ci-dessous. L'amplification de tension de ce circuit est :

$$V_{amp1} = (R10 + R11) / R11$$

L'alimentation devrait être disponible en 2 versions : maximum 30V de sortie et maximum 22V de sortie. Une combinaison de 1K et 5.6K donne un facteur de 6.6, ce qui est bon pour la version 30V. Pour la version 22V, nous utilisons 1K et 4.7K.

La résistance interne du circuit comme vu sur la base du BC547 est :

$$R_{in} = h_{fe1} * S1 * R11 * R9 = 100 * 50 * 1K * 47K = 235 \text{ MOhm}$$

- hfe est approximativement 100 à 200 pour un transistor BC547
- S est la pente de la courbe d'amplification d'un transistor et est approximativement de 50 [unit = 1/Ohm]

C'est plus que suffisamment haut pour la connection à notre DAC qui a une résistance interne de 5K.

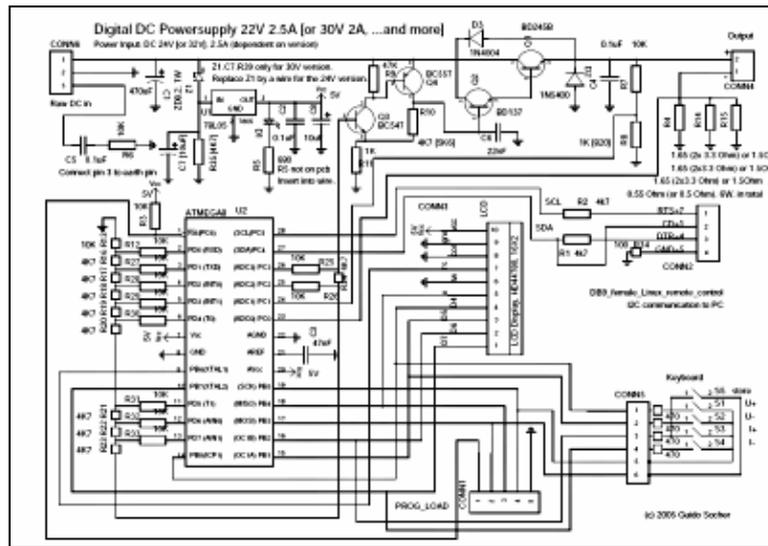
La résistance équivalente de notre résistance de sortie est :

$$R_{out} = (R10 + R11) / (S1 + S2 * R9 * R11) = \text{approximativement } 2 \text{ Ohm}$$

C'est suffisamment bas pour conduire le BD137 suivant.

R9 lie la base de BC557 à l'émetteur, ce qui signifie « off » pour le transistor jusqu'à ce que le DAC et BC547 arrivent. R11 et R10 lient la base du BD137 initialement à la terre qui coupe la sortie de l'étage Darlington. En d'autres mots, chaque composant dans cet étage d'amplification est éteint, initialement. Cela signifie que nous n'obtiendrons aucune oscillation, ni pics de sortie de ces transistors à la mise sous tension.

Maintenant que vous avez vu comment un tel circuit est conçu, nous devons ajouter quelques diodes de protection et des capacités pour stabiliser le circuit et ce sera fini.



Le circuit final est grand comme : [\[PNG\]](#) [\[PDF\]](#)

Les valeurs entre crochets droits sont pour la version de 30V.

Le diagramme du circuit et le fichier de conception de la platine (format pcb) sont aussi inclus dans le paquet tar.gz qui vous pourrez télécharger à la fin de cet article.

Le connecteur nommé « Raw DC in » (= CONN6) possède 3 broches. La broche 1 est le « plus », la 2 est le « moins » (= GND) et la broche 3 devrait être connectée à la broche de la terre. En d'autres termes, la « terre » n'est ni directement connectée à GND ni à « plus » non plus.

## Les limites

D'expériences précédents, nous savons que la plupart des lecteurs souhaiteraient « modifier » un peu le circuit. Voici une liste des limitations du matériel et de la manière de les surmonter :

78L05: la tension maximale d'entrée est de 30V, le ZD8.2 étend cette limite à environ 38V. En utilisant une diode Z différente, vous pouvez étendre cela un peu. Cela n'ira pas au-delà de 45V.  
La tension maximale de sortie devrait être au moins 2V plus basse que la tension d'entrée.

BD245B: 10A 80W, Les 80W sont cependant à une température de 25°C. En d'autres termes, ajoutez une marge de sécurité et calculez avec 60W :  
(tension maximale de sortie \* courant maximal < 60W

Vous pouvez ajouter un second BD245B pour aller jusqu'à 120W. Pour s'assurer que le courant se distribue de manière égale, ajoutez une résistance de 0.22 Ohms dans la ligne d'Emetteur de chaque BD245B. Le même circuit et la même carte peuvent être utilisés. Montez les transistors sur un refroidisseur en aluminium propre et connectez-les avec de courts fils à la carte.

Shunt de mesure du courant :

Il y a une résistance de 0.5 ou 0.55 Ohm avec 6W. Cela est suffisant pour environ 3A de sortie ( $I_{out}^2 * 0.55 \leq 6W$ ). Utilisez une résistance plus puissante pour des courants plus élevés.

# Les sources de courant

Vous pouvez soit utiliser un transformateur, un redresseur et de grands condensateurs ou vous pouvez essayer d'obtenir une alimentation électrique 32/24V d'ordinateur portable. Je me suis dirigé vers la seconde option. Ces « briques » d'alimentation électrique pour ordinateur portable sont parfois vendues très bon marché (fin de stock) et quelques unes fournissent 70W à 24V ou même 32V DC.

La plupart des gens iront probablement vers la solution du transformateur parce qu'ils sont très faciles à obtenir.

version 22V 2.5A : vous avez besoin d'un transformateur 18V 2.5A, un redresseur et un condensateur de 3000 $\mu$ F (raison :  $18 * 1.4 = 2.5V$ , au moins 1000 $\mu$ F par ampère)

version 30V 2A : vous avez besoin d'un transformateur 24V 2A, redresseur et un condensateur de 2200 $\mu$ F (raison :  $24 * 1.4 = 33.6V$ , au moins 1000 $\mu$ F par ampère)

Cela ne fait pas de tort d'acheter un transformateur qui peut fournir plus d'ampères.

Vérifiez que votre circuit possède une isolation correcte. Soyez sûr qu'il n'est **pas** possible de toucher aucune partie qui pourrait transporter 110V/230V, même quand la boîte est ouverte. Connectez toutes les parties métalliques du châssis à la terre (pas au GND du circuit).

## Les tests

Quand vous avez soudé tout ensemble, ne connectez pas encore le microcontrôleur au secteur. Effectuez, d'abord, quelques tests de base.

Test 1 : Connectez une alimentation électrique (au-moins 15V) à l'entrée de courant du circuit et vérifiez que vous obtenez 5V DC derrière le régulateur de tension.

Test 2 : Mesurez la tension de sortie. Elle devrait être de 0V.

Test 3 : Connectez les broches 7 et 26 sur le microcontrôleur avec un fil (le microcontrôleur n'étant pas raccordé au secteur). Vous devriez obtenir, en sortie, quelques volts de moins que ce qui est utilisé en entrée DC. Enlevez le fil et la sortie devrait retourner à zéro. Assurez-vous de ne pas provoquer de court-circuit sur la sortie.

Insérez le microcontrôleur et chargez le logiciel de test LCD en lançant les commandes suivantes dans le répertoire du paquet `digitaldcpower.tar.gz` décompressé :

```
make lcdtest
make lcdtestload
```

Vous devriez voir « ok works » sur l'affichage.

Maintenant, vous pouvez charger le logiciel final comme décrit ci-dessus.

# Le logiciel

Pour les curieux, j'ai inclus une description assez longue de chaque fichier .c dans le fichier README du paquet digitaldcpower.tar.gz. Le code contient également de nombreux commentaires. Sinon, j'ai l'intention d'écrire d'autres articles qui expliqueront juste la partie logiciel. Cet article est déjà assez long et vous devez construire le matériel, tout d'abord.

J'ajouterai aussi graduellement plus de fonctionnalités avec les prochains articles :

- Vous serez capables de stocker les paramètres (limite de tension et de courant) de manière à ce que l'alimentation électrique s'allume avec les mêmes paramètres au prochain allumage
- Vous serez capables de contrôler l'alimentation électrique à partir de votre PC sous Linux via le connecteur rs232.

Maintenant, vous voulez savoir comment utiliser le logiciel et comment le charger.

Décompressez le paquet digitaldcpower.tar.gz et entrez (« cd ») dans le répertoire qui est créé.

Editez le fichier hardware\_settings.h et ajustez-le en fonction du matériel. Ici, vous pouvez aussi effectuer les calibrations du voltmètre et de l'ampèremètre. Le fichier est bien commenté.

```
gedit hardware_settings.h
```

Connectez le câble de programmation et allumez le circuit. Ensuite, exécutez :

```
make wrfuse4mhz
```

Cela va positionner la fréquence d'horloge du microcontrôleur à 4MHz. Le logiciel est conçu pour cette fréquence.

```
make
```

Cela va compiler le logiciel.

```
make load
```

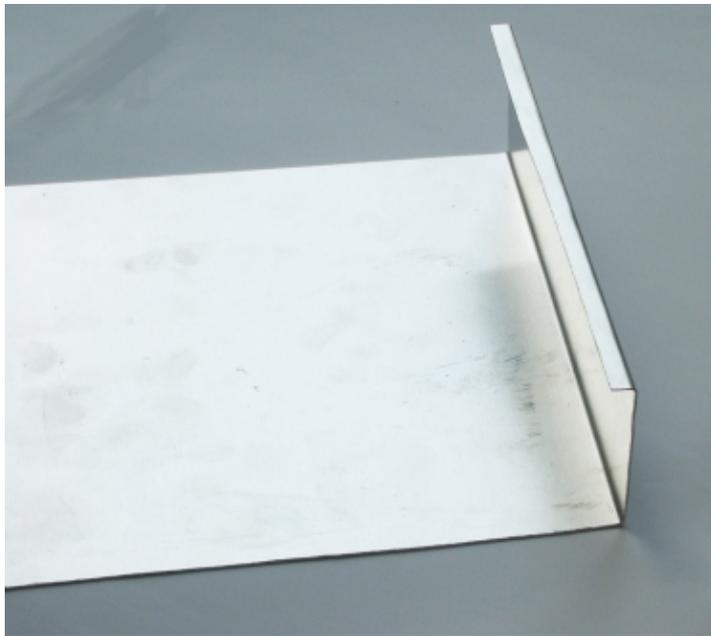
Cela va charger le logiciel

## Quelques photos et idées

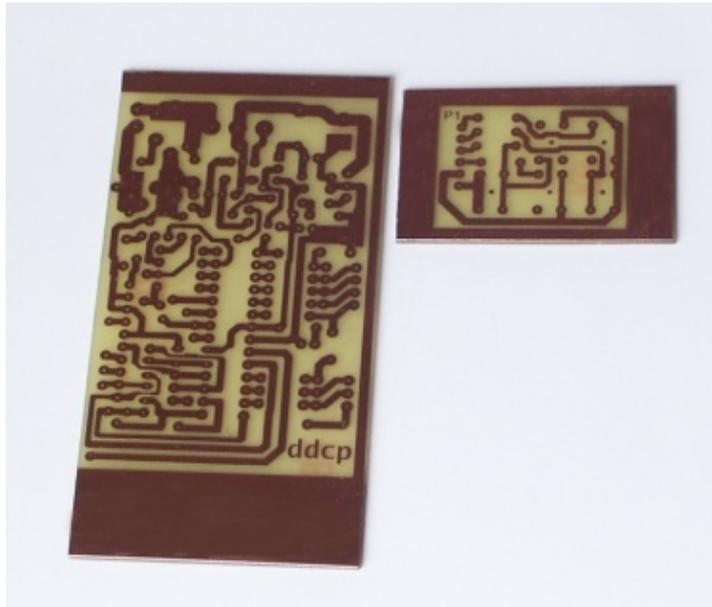
Voici quelques photos de l'alimentation électrique que j'ai construite. Les photos sont parfois meilleures qu'une longue description, spécialement lorsqu'il s'agit de matériel et de parties mécaniques.



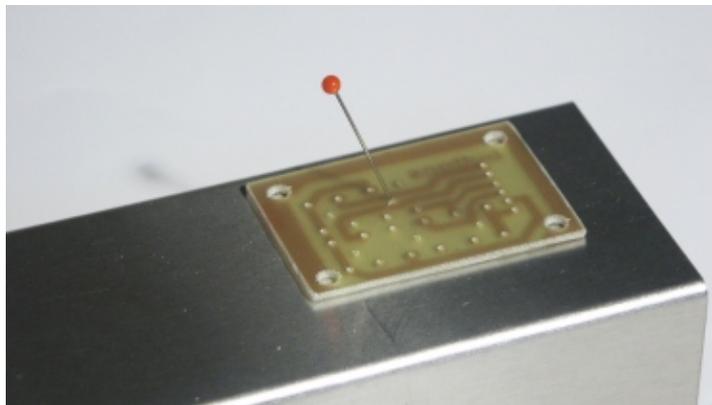
Vous pouvez construire un boîtier simple mais ayant l'air professionnel à partir d'une feuille d'aluminium et deux planches en bois.



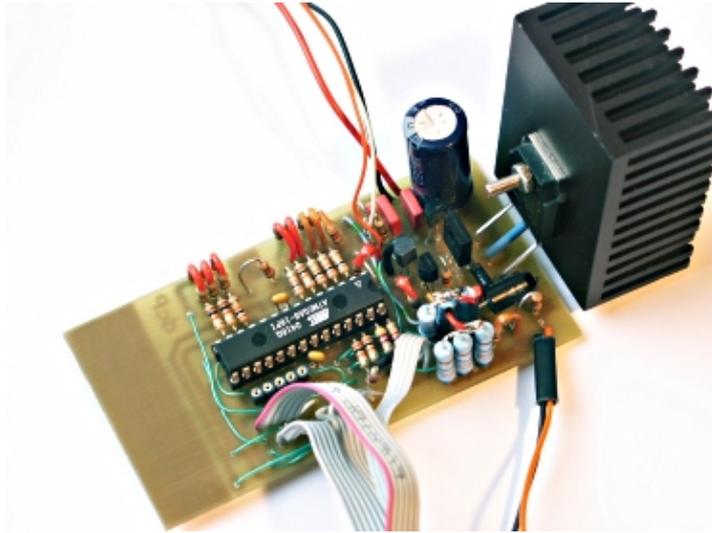
Pliez simplement l'aluminium deux fois. Cela est facile à faire, même sans équipement professionnel.



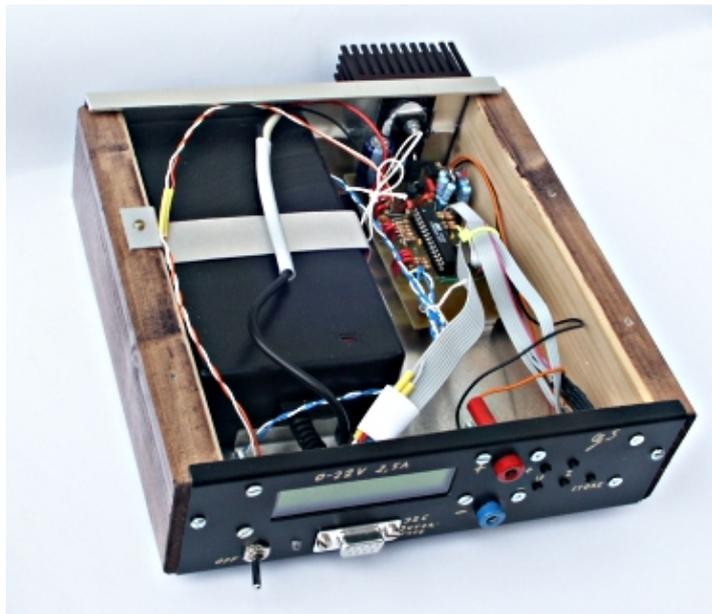
Les deux circuits imprimés. Le plus petit est pour le clavier.



Marquez les positions des boutons poussoirs sur le boîtier avant de souder les boutons sur la carte. Cela rend plus facile le perçage des trous à la bonne position



Le circuit. Très petit, riches en fonctionnalités et puissant.



L'alimentation électrique finale.

Amusez-vous ! ... Et il y aura d'autant plus d'amusement que ce matériel pourra faire beaucoup plus ...

## Références/Téléchargement

- [Page de téléchargement](#) de cet article (les mises à jour et corrections y seront également disponibles).
- Programmer le microcontrôleur AVR avec GCC, libc 1.0.4 : [article 352 de novembre 2004](#)
- Feuille de spécifications pour l'Atmega8 : allez sur <http://www.atmel.com/> et sélectionnez products->Microcontrollers ->AVR-8 bit RISC->Documentation->datasheets ([copie locale, pdf, 2479982 bytes](#))
- [section électronique de Tuxgraphics](#), une collection de tous les articles dans cette série.

- magasin en ligne Tuxgraphics, section microcontrôleurs, vous pouvez y commander tous les composants (transistors, composants passifs, affichage LCD, PCB, microcontrôleur, ...).

---

<p><u>Site Web maintenu par l'équipe d'édition LinuxFocus</u> © <u>Guido Socher</u> "some rights reserved" see <a href="http://linuxfocus.org/license/">linuxfocus.org/license/</a> <a href="http://www.LinuxFocus.org">http://www.LinuxFocus.org</a></p>	<p>Translation information: en --&gt; -- : Guido Socher (<a href="#">homepage</a>) en --&gt; fr: Jean-Etienne Poirrier (<a href="#">homepage</a>)</p>
---	---

2005-07-08, generated by lfparsr\_pdf version 2.51