

vivre vite, mourir vieux

prolongateur de durée de vie de batterie

Trop de fabricants ne font aucun cas de la santé des batteries qu'ils mettent dans leurs appareils ménagers ou autres ustensiles de cuisine. D'un point de vue électrique, l'ensemble n'est souvent rien de plus qu'un bloc-batterie, un interrupteur et un moteur. Parfois ces batteries s'épuisent vite, elles meurent trop jeunes. Ce projet montre comment un ATtiny45V, un MOSFET de puissance et une poignée de composants peuvent allonger leur durée de vie. Greffé sur une vieille râpe à fromage électrique, ce circuit en a amélioré le fonctionnement ; désormais, sa batterie n'en durera que plus longtemps. Vous pourrez bien sûr appliquer ce circuit de Jouvence sur d'autres appareils, p. ex. un aspirateur portatif.

Volker Schmidt
(Allemagne)

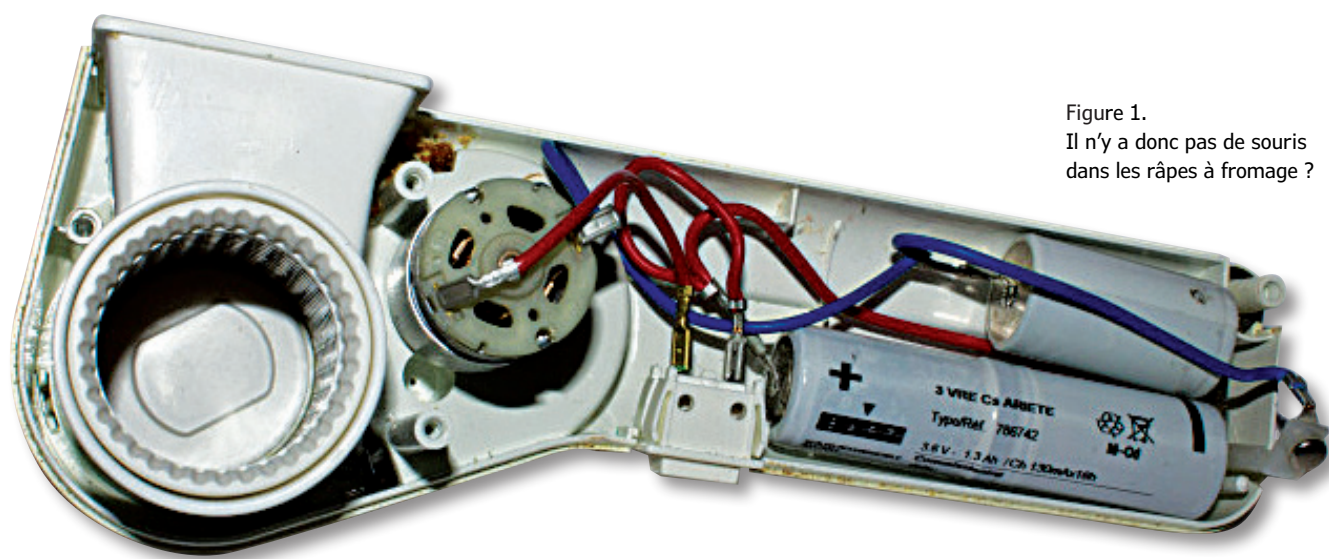


Figure 1.
Il n'y a donc pas de souris
dans les râpes à fromage ?

Un jour, pour le fromage c'était râpé
J'ai aussitôt cherché la source du problème. Ça n'a pas été long, il a suffi d'ouvrir la râpe à fromage (**fig. 1**) : les trois accumulateurs NiCd de 1300 mAh étaient fichus. Le responsable était la conception elle-même : le bloc-batterie était relié directement au moteur via un interrupteur (**fig. 2**), entraînant le risque d'une décharge complète. Or, avec ce genre d'alimentation, c'est un risque qui devrait être zéro.

J'ai d'abord dû acheter de nouveaux accus. Le fournisseur proposait encore le bloc-batterie d'origine sur Internet, mais chez un distributeur en ligne j'ai trouvé des accus Sub-C NiMH de 2400 mAh parfaitement adaptés à ma râpe. Pour son prototype, l'équipe d'Elektor a toutefois choisi un accumulateur lithium-ion qui fournit une tension nominale de 3,6 V (V_{CC}).

Quoi qu'il en soit, le simple remplacement des accus ne pouvait résoudre le problème

de la décharge profonde, une surveillance de la batterie était nécessaire. Une conception analogique aurait été trop complexe pour une tension si basse, et la carte aurait été trop grande pour équiper toutes sortes d'appareils (portatifs). Une solution s'imposa donc : un microcontrôleur.

Matériel

Le microcontrôleur ATtiny45V d'Atmel [1] a été choisi pour les tâches de commande car il possède un convertisseur analogique-numérique (CAN) de 10 bits, un boîtier à 8 broches (suffisantes ici), ainsi qu'une large plage de tension démarrant à 1,8 V minimum. En pratique ce minimum est toutefois de 2,1 V pour obtenir des conversions analogiques-numériques correctes (avec une référence interne de 1,1 V). La consommation totale de l'ATtiny45V est en outre très faible, le courant ajouté est d'environ 0,3 mA – et il est disponible en boîtier traversant et CMS. Notez bien que le circuit peut également fonctionner avec un ATtiny25V ou un ATtiny85V.

L'ATtiny est alimenté directement sous V_{CC} par la batterie Li-ion. Il est cadencé à 4 MHz par son oscillateur RC interne et n'a donc pas besoin de quartz externe.

D'après le schéma (fig. 3), le MOSFET T1 est effectivement placé entre le pôle négatif de la batterie (GND) et la charge (ici un moteur). Il est commandé via R1 par la ligne du port PB3 du μC . Le diviseur de tension R2/R3 fournit une tension $V_{CC}/4$ à une entrée CAN sur PB4. Cette division est nécessaire, car le convertisseur travaille avec une tension de référence de 1,1 V. Comme nous allons le voir, le microcontrôleur se sert de ce niveau de tension pour les tâches de surveillance. La batterie est shuntée par C2 pour prévenir tout bruit susceptible de nuire à l'ATtiny. L'entrée Reset est forcée au niveau V_{CC} via R4 et D1.

Programme

Dans la fonction *Main* du micrologiciel, la tension sur PB4 est mesurée et stockée dans un tableau de 10 éléments à chaque itération de boucle. Comme nous l'avons vu, ces tensions sont proportionnelles à V_{CC} , et une simple conversion permet de déduire la tension réelle du bloc-batterie. Une fois le tableau rempli, le programme compare les valeurs du

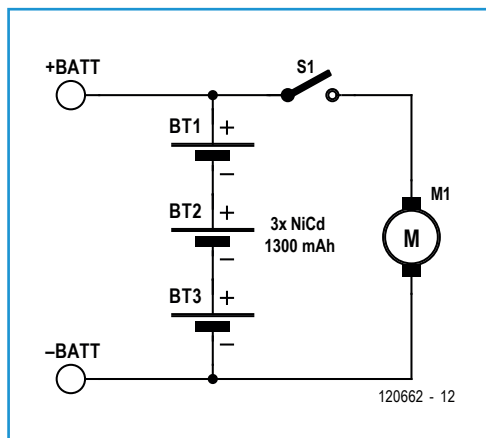
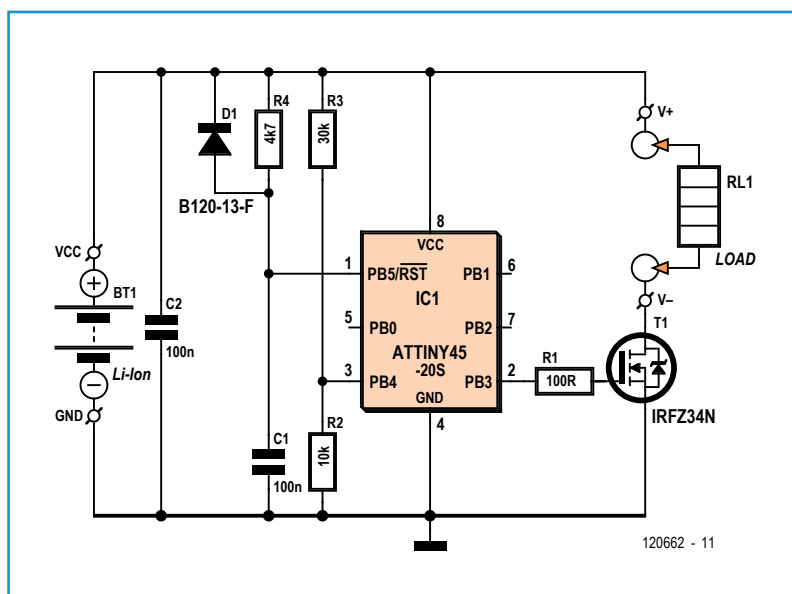


Figure 2. Le (sobre) circuit d'origine de la râpe.

tableau avec la tension minimum. Si toutes les valeurs sont inférieures au minimum, la sortie PB3 est mise au niveau haut et le MOSFET passe à l'état bloqué, ce qui ouvre le circuit à courant fort et empêche la batterie de subir une décharge profonde. Si au contraire toutes les valeurs ne sont pas inférieures au minimum, PB3 garde son niveau bas et le MOSFET reste à l'état conducteur (et le circuit reste fermé). La tension minimum est stockée dans la variable *MinVolt* (en mV) ; vous devrez adapter cette valeur au type de batterie utilisée.

Le programme est écrit en C et riche en commentaires. Vous pouvez télécharger le code source sur la page Elektor.LABS [2] liée à ce projet. Vous y trouverez aussi le projet pour *Atmel Studio 5*, ainsi qu'une brève explication

Figure 3. Schéma du prolongateur de durée de vie de batterie.



sur la façon dont sont converties et comparées les valeurs du CAN. Pour la configuration des fusibles, assurez-vous que CKDIV8 soit mis à *True* (= horloge sur 4 MHz).

Ça sentira le fromage avant le sapin

J'ai construit mon prototype sur un morceau de plaque d'essai, me suis servi de tube thermo-rétractible pour l'isoler, puis l'ai installé près de l'accu du haut (la figure 1 montre la configuration d'origine de la râpe). Par commodité, l'équipe d'Elektor a préféré une conception sur circuit imprimé, téléchargeable [2] comme le schéma, les fichiers PDF et Gerber, ainsi que ceux du projet *Eagle*. Si *Eagle* ne vous est pas familier, ou si vous souhaitez vous perfectionner avec ce programme, nous vous recommandons la lecture du *Guide de démarrage pour Eagle V6 — Prenez les commandes d'Eagle*, livre (en anglais) publié par Elektor [3].

En attendant, plus d'excuses pour ne pas ressortir de vos placards ces appareils alimentés par batterie et depuis longtemps oubliés !

(120662 - version française : Hervé Moreau)

LISTE DES COMPOSANTS

Résistances

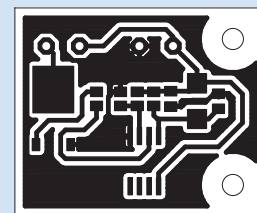
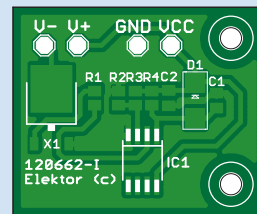
- R1 = 100 Ω, 0,25 W, 5 %
- R2 = 10 kΩ, 0,25 W, 1 %
- R3 = 30 kΩ, 0,25 W, 1 %
- R4 = 4,7 kΩ, 0,25 W, 1 %

Condensateurs

- C1,C2 = 100 nF, 16 V min., céramique

Semi-conducteurs

- T1 = IRFZ34, MOSFET, canal N, 60 V
- D1 = B120-13-F, diode Schottky, 1 A, 20 V
- IC1 = Attiny45V-10SU à 8 bits, boîtier SOIC, programmé



Liens

- [1] www.atmel.com/devices/ATTINY45.aspx
- [2] www.elektor-labs.com/120662
- [3] www.elektor.fr/products/books/english/eagle-v6.2476942.lynx

