

Héli-message défilant

Afficheur rotatif à LED à contrôleur AVR



Steffen Sorge

Ce « bidule » astucieux a son origine dans un « mémorial » commémorant l'obtention du baccalauréat. Il permet de faire pivoter sur 360 degrés un message défilant en n'utilisant que 8 LED bleues. Un afficheur de grande taille vraiment réduit à l'essentiel...

L'idée de cet afficheur est née au cours d'une séance de « remue-méninges » sur la réalisation d'un « bidule » commémorant l'obtention du baccalauréat. Il doit s'agir d'un objet original. Dans l'établissement d'enseignement fréquenté par l'auteur, cette tâche était traditionnellement confiée aux élèves de la filière électronique.

Des réveils basés sur un principe similaire ont servi de point de départ. Ces produits d'Extrême-Orient peu coûteux n'affichent toutefois que l'heure et pas de texte.

Le premier prototype a été mis au point en quelques jours de la façon suivante : 8 LED tirées de la caisse à bricolage, un commutateur de synchronisation de début du texte, un AT90S2313, une pile monobloc 9V et quelques autres babioles ont été montés sur une petite latte de bois. La latte a été fixée à l'axe d'un petit moteur à courant continu qui la faisait pivoter. Un peu de programmation a permis d'afficher un mot.

Ce premier prototype (NDLR : tout a fait dans le style de l'atelier ELEKTOR)

présentait encore quelques défauts :

- LED pas assez lumineuses ;
- durée de vie limitée du commutateur (commutation à chaque tour du moteur) ;
- la pile rotative ne constituait pas non plus la panacée...

Mais les problèmes sont faits pour être résolus. On s'est tout d'abord procuré des LED bleues 3 500 mCd à haute luminosité. Puis on a résolu le problème du commutateur en le remplaçant par son équivalent à effet Hall récupéré dans une vieille imprimante. La pile a fait place à une alimentation par bague de frottement. Pour terminer, le AT90S2313 a dû céder la place à un ATmega8...

Un contrôleur pour 8 LED

Le schéma (**figure 1**) montre que le circuit ne comprend en gros qu'un microcontrôleur (IC1) qui commande 8 LED

et est raccordé à un capteur de Hall (IC3). Les composants externes requis par le contrôleur se limitent à un quartz 16 MHz (X1) avec 2 condensateurs de charge (C4 et C5). Un bon vieux 7805 (IC2) sert de régulateur de tension.

Le microcontrôleur est incapable de fournir un courant suffisant aux LED fortement pulsées. Elles sont donc excitées par les étages à transistors T1 à T8.

Étant donné que les 8 LED tournantes doivent « écrire » une image de 8 lignes (toutefois pas de gauche à droite mais selon une rotation de 360 degrés), il faut que le microcontrôleur soit synchronisé par une « impulsion d'image ». C'est là que le commutateur de Hall (IC3) entre en jeu. Son niveau de sortie change dès qu'il se trouve dans un champ magnétique (donc près d'un aimant). Sa sortie, « haute » en l'absence de champ, passe à « bas » lorsqu'il s'approche d'un aimant.

Ce changement de niveau entraîne la création de l'image par le logiciel du microcontrôleur.

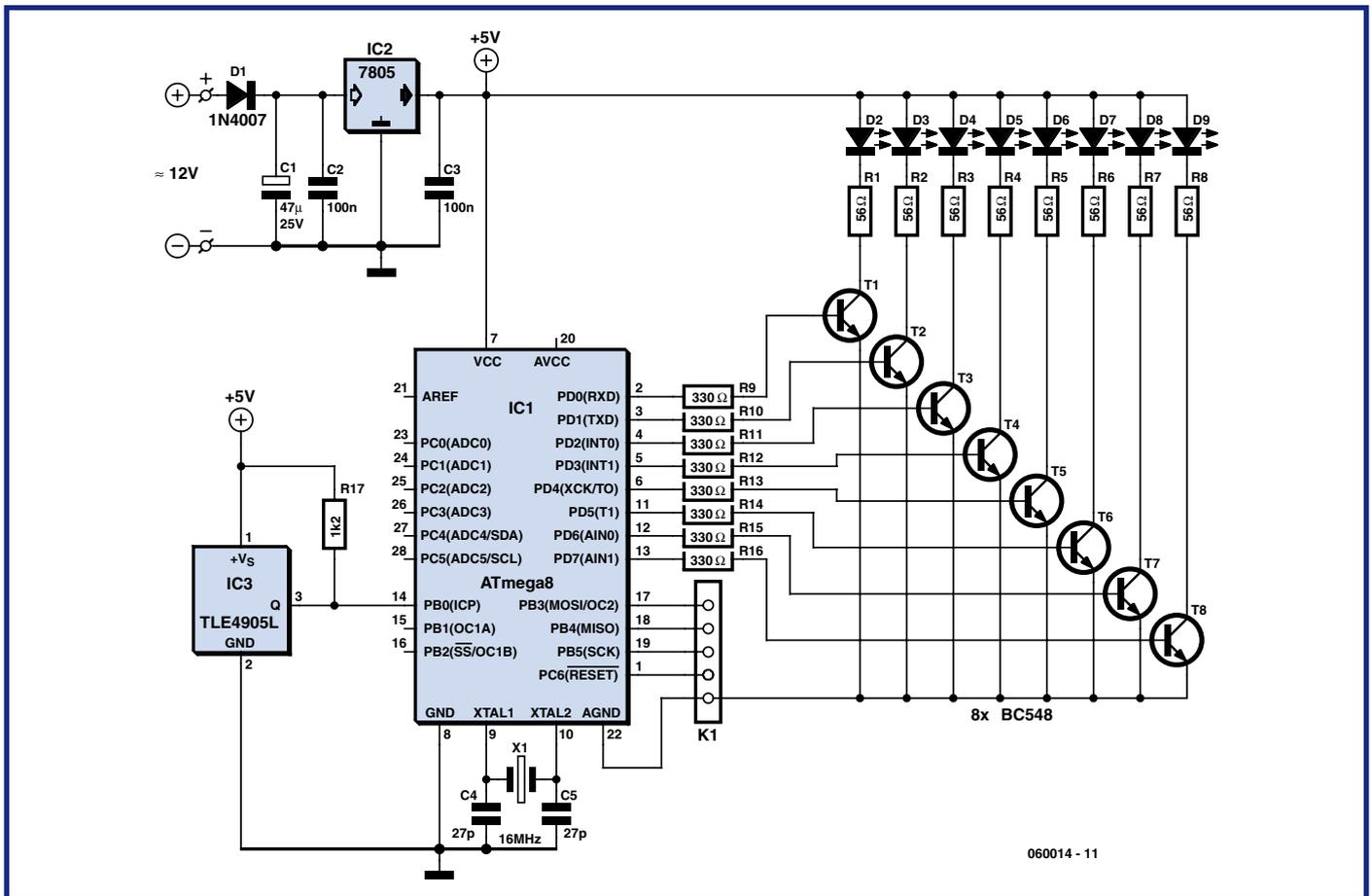


Figure 1. Le circuit se compose pour l'essentiel d'un contrôleur AVR commandant 8 LEDs et relié à un capteur de Hall (IC3).

Logiciel

Le ATmega8 est un contrôleur AVR de Atmel appartenant à la famille des contrôleurs RISC (à jeu d'instructions réduit). Le programme du message défilant n'a toutefois pas été écrit en langage assembleur mais en Basic BASCOM-AVR.

Le logiciel, généreusement commenté, peut être téléchargé gratuitement à partir du site Web ELEKTOR (fichier **060014-11.zip**).

Le programme engendre de petites images individuelles de fragments du texte mémorisé dans le contrôleur. Ces fragments de texte sont ensuite convertis en colonnes et transmis aux LED à l'instant adéquat. La rotation du moteur donne l'impression d'un texte écrit sur un cylindre.

Le contrôleur ne peut mémoriser qu'un message défilant de 255 caractères. Ces caractères sont mémorisés dans le tableau « Instring » contenant 3 éléments (3 segments de texte) qui peuvent être affichés successivement.

La sortie des caractères est synchronisée par le commutateur à effet Hall déjà mentionné. Il signale aussi la rotation du moteur (toutes les LED sont éteintes quand le moteur est immobile). La longueur du texte défilant sur un « afficheur » de ce type dépend du rayon du cercle des LED pivotantes, donc de la longueur du bras rotatif. Le montage dont la photo est reproduite ici fonctionne avec un texte de 30 caractères. Le programme tourne en une boucle sans fin synchronisée par le commutateur à effet Hall et envoie 30 caractères en continu.

Le programme peut générer les caractères suivants : les majuscules de A à Z, ci-inclus les voyelles infléchies (Ä, Ö, Ü), les espaces, les points d'interrogation et d'exclamation, le point, la virgule et le trait d'union.

Chaque caractère a droit à son sous-programme. Il est donc facile d'ajouter ou de modifier des caractères. Un sous-programme n'est ici rien d'autre qu'une succession d'instructions d'écriture pour le port D commandant les LED. Chaque instruction d'écriture

est suivie d'une courte pause produite par une boucle de délai.

Chaque instruction d'écriture crée une colonne du caractère. Celui-ci est composé de plusieurs colonnes, chacune contenant 8 points d'image. On peut donc « dessiner » très facilement un caractère sur du papier quadrillé et convertir les colonnes en octets de gauche à droite. Supprimons les boucles de délai (sous-programme 'Wa') dans le code source... et le caractère apparaît. Par exemple :

```
Sub A
Portd = &B11000000
Portd = &B00110000
Portd = &B00101110
Portd = &B00100001
Portd = &B00101110
Portd = &B00110000
Portd = &B11000000
Portd = &B00000000
Wa
Wa
End Sub
```

La dernière instruction d'écriture est toujours '0' pour éteindre les LED.

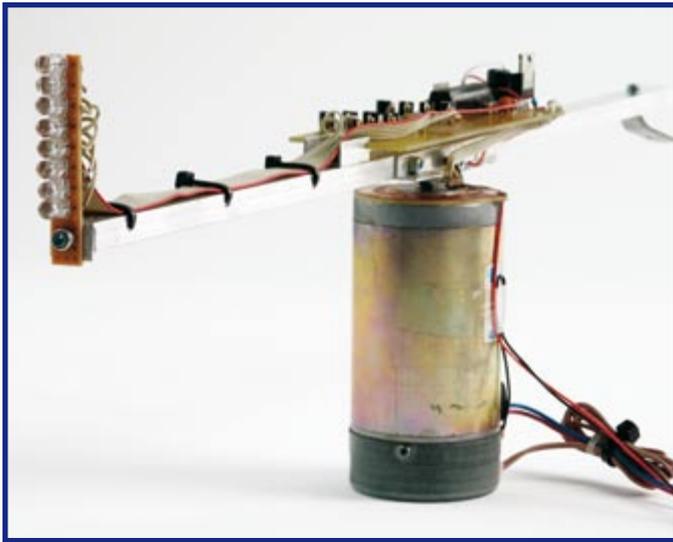


Figure 2. Coup d'œil sur le bras pivotant avec la carte du contrôleur.

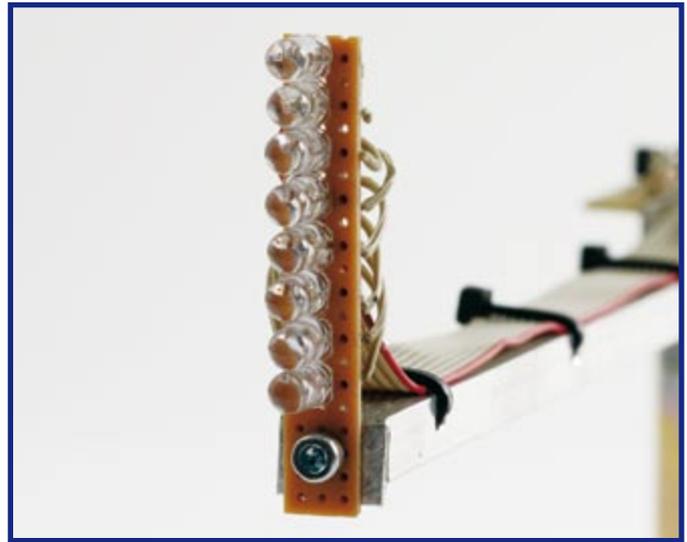


Figure 3. Disposition des 8 LED à l'extrémité du bras pivotant.

Le délai du sous-programme 'Wa' doit être ajusté au nombre de tours du moteur. Il faut procéder par tâtonnements pour déterminer la valeur idoine.

La reproduction du texte commence par les 30 caractères du premier segment mémorisé dans Instring. Le texte est décalé d'un caractère (la position initiale du texte est surveillée dans la variable « Laufer ») après 4 rotations du moteur (surveillée dans la variable « Laufer1 »). Lorsque la fin du premier segment de texte est atteinte, le segment suivant est choisi (variable « Strz »). La fin du dernier segment marque aussi le recommencement du cycle.

Une modification importante des paramètres comme le nombre de tours et le rayon du cercle des LED tournantes n'entraîne que quelques petites modifications dans le programme :

Une modification du rayon (longueur du bras) entraîne l'adaptation en deux endroits du nombre de caractères (différent de 30) apparaissant sur « l'afficheur ».

La modification du nombre de tours entraîne celle de la longueur des pixels (sous-programme 'Wa'). Il est possible d'ajuster la vitesse de

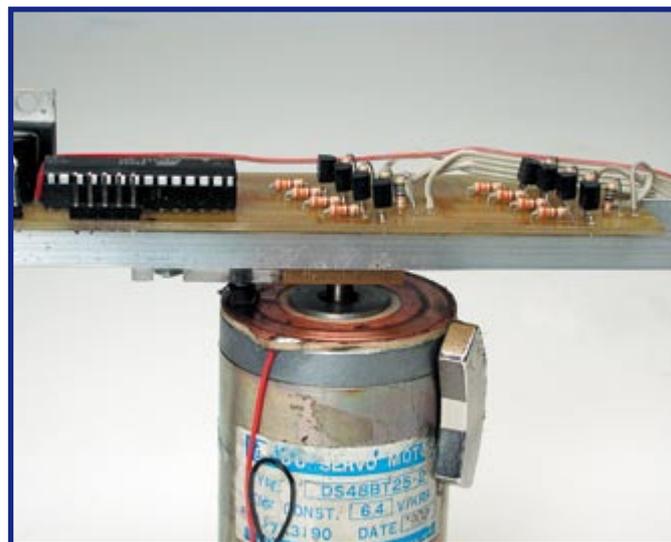


Figure 4. L'alimentation du circuit pivotant est assurée par une bague de frottement.

défilement du texte (celle à laquelle le texte se déplace). Le volume de texte peut être augmenté (si on dispose d'un contrôleur avec plus de mémoire).

Construction du circuit

Le circuit peut être monté sans trop de peine en suivant le schéma. Comme le ATmega8 utilisé est aussi en version DIL (ATmega8-16PU), il ne faut pas craindre de problèmes de CMS ou autres. Il faut choisir les résistances R9 à R16 en fonction du courant nominal des LED. On peut simplement les calculer comme résistances

de protection à 5 V. La valeur de 56 Ω a été choisie pour le prototype.

Outre la liste de pièces, le dessin des pistes et le plan de montage utilisés par l'auteur peuvent être téléchargés à partir du site Web ELEKTOR. Il faut toutefois procéder avec circonspection : les désignations des connexions dans le plan de montage ne figurent pas dans le schéma (figure 1) ou présentent des différences. Le schéma correspond toutefois électriquement à la carte et permet de la contrôler en cas de doute.

La carte, de forme allongée, est adaptée aux dimensions du bras pivotant. Les LED à l'avant du bras sont montées l'une au-dessus de l'autre sur une carte plus petite (voir photos). La carte à espacement normal de 2,54 mm s'accommode mal du montage en rang des LED 5 mm. Il faut aplatir quelque peu leur boîtier des deux côtés opposés. Les 2 cartes sont reliées par un câble plat 9 conducteurs. Les anodes des LED sont interconnectées et reliées à +5 V ; chaque cathode est reliée à sa résistance de protection (R9 à R16). D2 (reliée à R1) est la LED supérieure sur la carte, D9 (reliée à R8) est la LED inférieure. Si on inverse le raccordement de la ligne de LED (D9 en haut, D2 en bas), l'écriture a aussi « la tête en bas ».

Programmation

Le téléchargement **060014-11.zip** contient 3 fichiers : un .bas, un .hex et un .bin. Le fichier .bas contient le code source en « Bascom ». Le fichier .hex ou .bin, directement flashé dans le AT-Mega 8, sera alors prêt à être exécuté. Le texte affiché fera défiler le message de démonstration programmé par l'auteur.

Il faut disposer de Bascom pour programmer son propre texte. Ouvrir le fichier Schrift.bas et modifier le contenu du tableau INSTRING. Il ne reste plus qu'à presser F7 pour compiler le code source.

Appliquer la tension de fonctionnement (9...12 V) au circuit pour programmer le contrôleur. Raccorder les connexions de programmation (K1 dans le schéma) au PC par l'intermédiaire d'un programmeur adéquat (par exemple PonyProg). Flasher le fichier Schrift.bin ou Schrift.hex dans la puce (Positionner les Fusebits sur 16 MHz externe !). La version démo gratuite BASCOM-AVR DEMO 1.11.8.3 ne compile pas

plus de 4 Ko de code. Il faut limiter la longueur du texte pour l'utiliser, donc renoncer à placer 3 textes dans le tableau INSTRING.

Mécanique

Le « Rotor » est constitué par un bras d'environ 50 cm en tôle d'aluminium. On fixera au milieu du bras une petite prise reliée électriquement à l'axe du moteur. Relier ensuite la connexion de masse à la carcasse du moteur (!). Une bague de frottement faite du matériau de base d'une carte (sans revêtement de laque photosensible) est montée à la surface de la carcasse et reliée au « + » de l'alimentation. Le bras du rotor est muni d'un petit balai de charbon frottant contre la bague. Ce balai doit être isolé électriquement de la tôle d'aluminium du bras. Le champ magnétique d'un aimant, fixé également à la carcasse du moteur, « effleure » et donc active à chaque tour le commutateur de Hall monté à l'emplacement approprié du bras.

Le moment est venu de monter la carte sur le bras (éviter tout déséquilibre).

Pour alimenter la carte, raccorder la masse à la prise (reliée électriquement à l'axe du moteur) et le « + » au balai de charbon. Rééquilibrer s'il subsiste encore un déséquilibre du bras.

Relier la source de courant (bloc d'alimentation secteur 9 à 12 V) à la bague de frottement (plus) et à la carcasse du moteur (masse). Le circuit consomme en gros la somme des courants des LED, à quoi vient s'ajouter le courant du moteur, si celui-ci est alimenté par la même source.

Dès que le courant est appliqué et que le bras pivote, le texte rotatif devrait apparaître sur « l'afficheur » – exactement comme sur la photo en début d'article...

P.S. Le succès de l'auteur ne se borne pas au « Mémorial commémorant l'obtention du baccalauréat ». Il a aussi obtenu ce titre et étudie entre-temps l'électrotechnique et les techniques de l'information à l'I.U.T. de la Côte Ouest à Heide, en Allemagne.

(060014-1)

Publicité