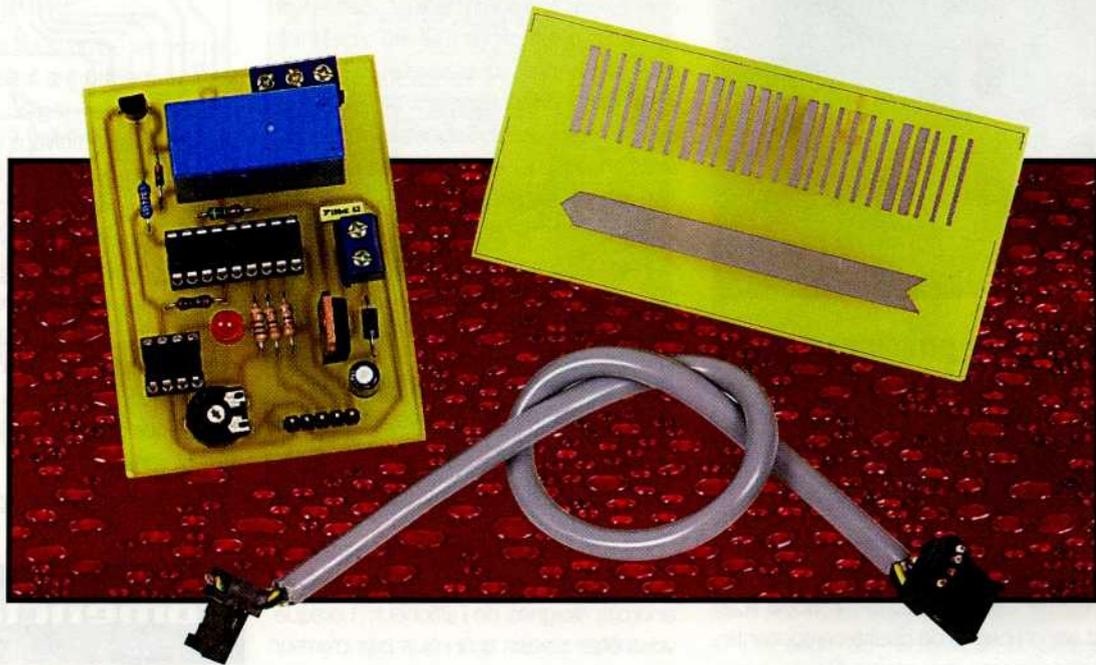


Serrure électrique à code-barres



Les serrures traditionnelles à clés ont encore de beaux jours devant elles. Il n'empêche, de nombreux dispositifs existent déjà ou s'inventent chaque année pour "moderniser" ce système. Parmi ceux-ci, on peut citer les télécommandes radio ou infrarouges, les transpondeurs, les iButtons, les cartes à puces, les cartes magnétiques et les code-barres. C'est ce dernier moyen qui est illustré dans ce montage.

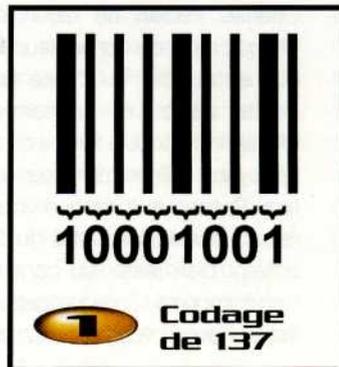
Notre lecteur de code-barres est un simple capteur CNY70 associé à un microcontrôleur PIC16F84, la clé à code-barres étant quant à elle gravée sur un morceau d'époxy. Le programme du microcontrôleur tient en quelques lignes écrites en BASIC F84, langage proche de l'assembleur mais simple à comprendre et à utiliser : le montage est donc idéal pour s'initier tout à la fois à l'électronique en général et à ce microcontrôleur, en particulier.

Dessin de la clé graphique, le code-barres

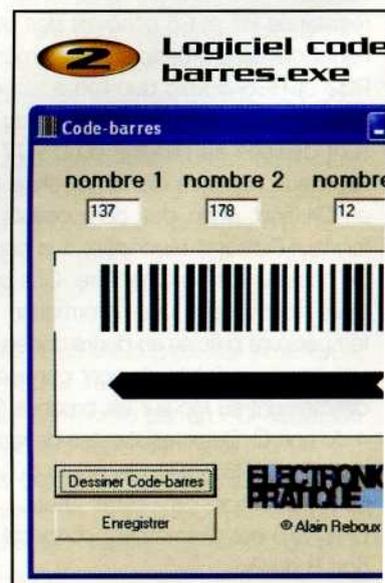
Un code-barres est une représentation graphique d'une information, représentation optimisée pour une lecture optique. A cet effet, les barres doivent contraster avec les espaces, ce qui explique qu'elles sont souvent noires sur fond blanc. Le principe d'un lecteur de code-barres est rapidement déduit : un rayon émis par une source lumineuse est envoyé sur le code-barre, la lumière plus ou moins réfléchiée par les barres et les espaces est alors captée par un récepteur optique puis interprétée par un décodeur.

Comme vous l'avez certainement remarqué lors de vos achats, il n'y a pas de standard unique pour le

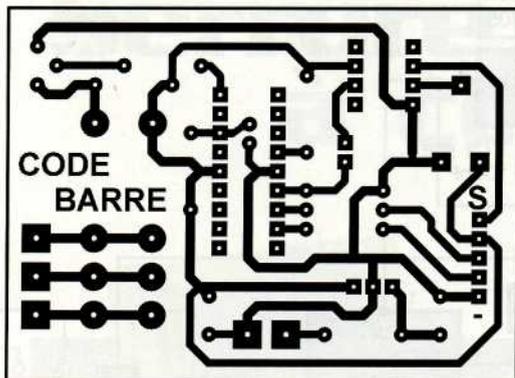
graphisme des code-barres : chaque industrie ou organisme a son propre standard (EAN13, CODE 39, etc.) alors pourquoi pas nous ! La **figure 1** décrit le principe que nous avons utilisé pour coder un nombre compris entre 0 et 255. Ce nombre s'écrit en binaire à l'aide de 8 bits de valeur 0 ou 1. Dans l'exemple, le nombre à convertir en code-barres est 137. Ce nombre s'écrit en notation binaire 10001001. Notre représentation graphique va consister à tracer une large barre noire suivie d'un mince espace blanc pour coder un 1 et à tracer une mince barre noire suivie d'un large espace blanc pour coder un 0. Pour préciser les termes "large" et "mince", "large" représente une grandeur métrique deux fois plus grande que "mince". Un tel codage présente



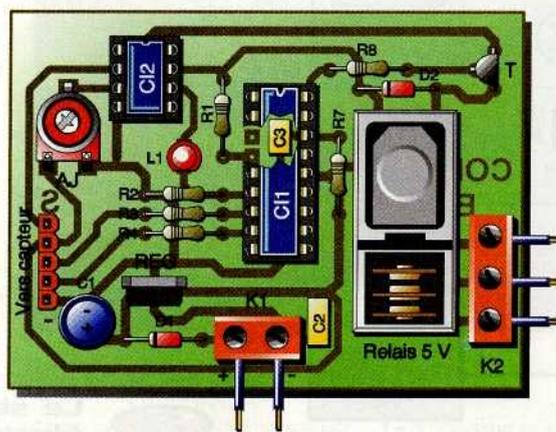
pour nous plusieurs intérêts : le tracé d'un tel code-barres est à la portée de n'importe quel programmeur (donc moi !) avec un algorithme simple et surtout, sa lecture est aisée par un capteur optique quelconque sans nécessiter la



finesse d'un rayon laser. Le logiciel CODE-BARRES.exe, créé à l'occasion de ce montage et présenté **figure 2**, permet de dessiner des clés d'après trois nombres de huit bits. Le nombre de clés différentes possible est donc de 256 x 256



4 Tracé du circuit imprimé de la carte à PIC



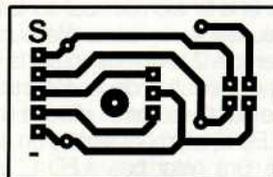
6 Implantation des éléments de la carte à PIC

la couleur rouge. Pour cela, l'anode de la LED rouge reliée à RB1 est à l'état haut alors que l'anode de la LED verte reliée à RB2 est à l'état bas. Lorsqu'une clé est passée devant le capteur et reconnue, les états de RB1 et RB2 sont inversés et la LED devient alors verte pendant dix secondes. Pendant ce même délai, le relais commandant une serrure électrique est activé par un passage à l'état haut de RA1 et donc la conduction du transistor T.

Le programme du PIC

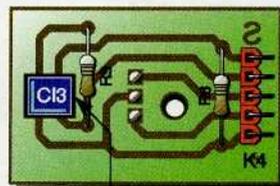
Le programme CODE_BAR du PIC est disponible sur le site Internet d'ELECTRONIQUE PRATIQUE sous deux formes : la première est le listing en BASIC F84 présenté dans cet article, la seconde est son fichier hexadécimal. Les lecteurs ne possédant pas le BASIC pourront ainsi charger directement le fichier hexadécimal à partir d'un des programmeurs proposés par la revue, les lecteurs possédant le BASIC pourront, plus tard, modifier plus facilement le programme source selon leurs envies et surtout selon leur propre clé !

L'écriture du programme se fait sans difficulté majeure. Les lecteurs fidèles

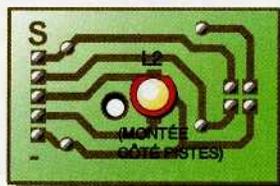


5 Tracé du circuit imprimé du capteur

7 Implantation des composants de la carte capteur



FACE MARQUÉE



(MONTÉE CÔTÉ PISTES)

;(1)déclaration des variables et des tableaux

```
VAR IMAGE
VAR FOIS
VAR LIGNE
DATA AFFI
```

;(2)initialisation

```
ORG 0
CONFIG PORTA,0
CONFIG PORTB,0
OUT PORTB,128
OUT PORTA,31
```

;(3)Affichage

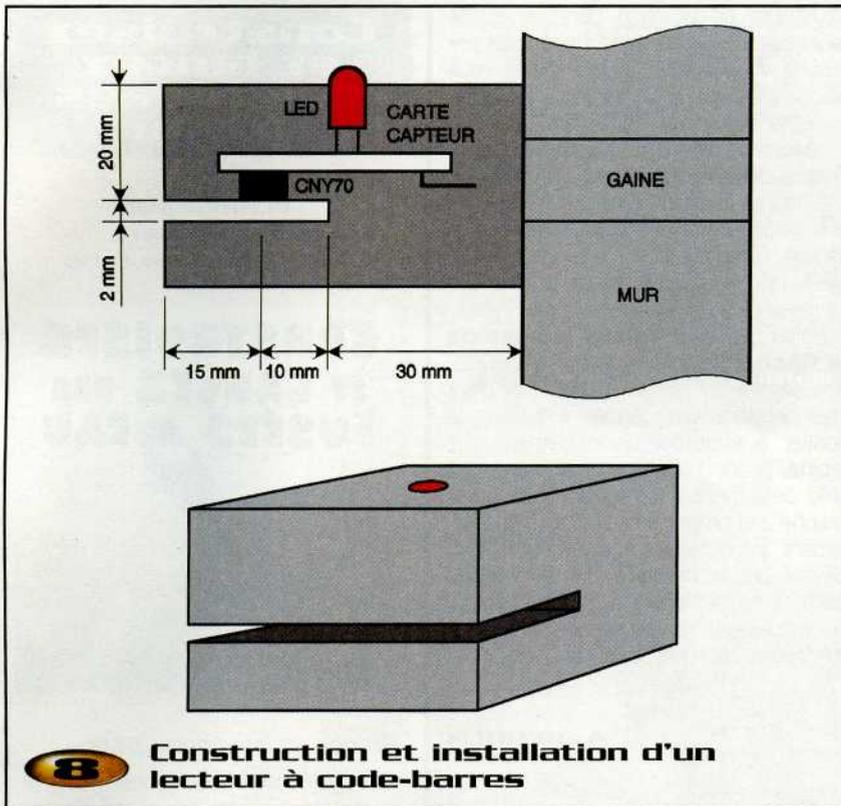
```
DEBUT FOR IMAGE=0 TO 4
FOR FOIS=1 TO 10
LIGNE=6*IMAGE
OUT PORTB, AFFI[LIGNE]
BCF PORTA,4
PAUSE 1
BSF PORTA,4
INCF LIGNE,1
OUT PORTB, AFFI[LIGNE]
BCF PORTA,3
PAUSE 1
BSF PORTA,3
INCF LIGNE,1
OUT PORTB, AFFI[LIGNE]
BCF PORTA,2
PAUSE 1
BSF PORTA,2
INCF LIGNE,1
OUT PORTB, AFFI[LIGNE]
BCF PORTA,1
PAUSE 1
BSF PORTA,1
INCF LIGNE,1
OUT PORTB, AFFI[LIGNE]
BCF PORTA,0
PAUSE 1
BSF PORTA,0
INCF LIGNE,1
OUT PORTB, AFFI[LIGNE]
BCF PORTB,7
PAUSE 1
BSF PORTB,7
NEXT FOIS
NEXT IMAGE
GOTO DEBUT
```

;(4)les données de l'affichage

```
AFFI 191,191,179,179,191,191
191,161,173,173,161,191
128,158,158,158,158,128
191,161,173,173,161,191
191,191,179,179,191,191
```

retrouveront de profondes similitudes avec des programmes de réception infrarouge parus dans des numéros précédents d'ELECTRONIQUE PRATIQUE. Le programme écrit en BASIC F84 se comprend facilement avec les quelques commentaires qui suivent sur le rôle de chaque partie du programme.

1: Déclaration des variables. Pour utiliser une variable dans le programme, il faut la déclarer en tête de programme. Pour rendre plus facile la compréhension du programme, il est recommandé de baptiser les variables par un nom en rapport



8 Construction et installation d'un lecteur à code-barres

boucle **ATT...GOTO ATT** tant que RBO reste à l'état haut, donc tant qu'aucune barre ne passe devant le capteur.

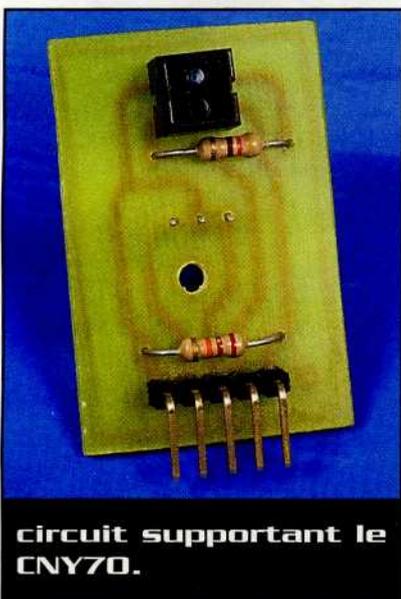
4 : décodage des 8 premiers bits. Le capteur ayant détecté le passage d'une barre, le PIC va mesurer la durée de son passage (DUREE_B) et la durée de l'espace qui suit (DUREE_E). Pour mesurer ces temps, l'état de la broche RBO est échantillonné toutes les 200 µs (GOSUB DELAI_200). La fin du premier bit commence lors de la détection de la barre du second bit. A cet instant, la comparaison de DUREE_B et DUREE_E du premier bit permet d'en déduire sa valeur 0 ou 1. La même opération est effectuée sur les 7 autres bits et au final, CODE_1 contient la valeur des 8 bits lus.

5 et 6 : décodage des 16 autres bits. CODE_2 et CODE_3 sont obtenus de la même manière que CODE_1.

7 : est-ce le bon code-barres ? Si la clé qui a défilé devant le capteur est bien la clé que doit détecter le programme, alors **CODE_1 = NOMBRE_1, CODE_2 = NOMBRE_2, CODE_3 = NOMBRE_3**. Dans ce cas, la LED bico-

avec leur fonction: les variables NOMBRE_1, NOMBRE_2, NOMBRE_3 représentent les trois nombres définissant la clé et utilisés par CODE_BARRES.exe pour la dessiner, DUREE_B et DUREE_E représentent les durées respectives d'une barre et d'un espace successif lors du défilement de la carte devant le capteur, CODE_1, CODE_2, CODE_3 sont les trois nombres lus sur la carte par le capteur, etc.

2 : Initialisation . Avec les deux instructions CONFIG, toutes les broches des ports A et B sont programmées en sortie sauf RBO. La boucle FOR...NEXT produit cinq clignotements de la LED L2 en deux



circuit supportant le CNY70.



mise en place de la LED bicolore côté cuivre.

couleurs : c'est une manière de constater que le programme et le montage fonctionnent à la mise sous tension. Après cette série de clignotements, seule la partie rouge de cette LED reste allumée (BSF PORTB,1). L'initialisation se termine par l'attribution de valeurs numériques aux variables **NOMBRE_1, NOMBRE_2, NOMBRE_3** : il est impératif que ces trois variables aient les mêmes valeurs que dans CODE_BARRES.exe.

3 : le programme principal. Après avoir initialisé les 3 nombres CODE_1, CODE_2 et CODE_3 à 0, le programme parcourt la

lore passe en vert (**BSF PORTB,1 et BSF PORTB,2**) et le relais est activé (**BSF PORTA,1**) pendant 10 secondes (**PAUSE 100**).

8 : pause de 200 µs. Pour comparer les deux nombres représentant les durées des défilements des barres et espaces, il est important que ces nombres soient inférieurs à 255 puisque les nombres sur un PIC ne sont codés que sur 1 octet. DUREE_B et DUREE_E sont donc incrémentés d'une unité toutes les 0,2 ms. La lecture du capteur est interrompue si DUREE_B dépasse 100, soit si la lecture

d'une barre dure plus de 20 ms. Une barre "large" mesurant environ 2 mm, la vitesse de défilement minimale est donc de 10 cm/s. La carte mesurant environ 10 cm, son passage devant le capteur ne doit pas durer plus de 1 s.

Le logiciel CODE_BARRES.exe

Utilitaire sans autre prétention que celle de créer rapidement une clé à code-barres, le logiciel CODE_BARRE.exe se prend en main rapidement. Entrez dans les trois champs prévus à cet effet les **trois nombres NOMBRE_1, NOMBRE_2 et NOMBRE_3** définissant la clé puis appuyez sur le bouton **Dessiner code-barres. Sauvegardez** alors ce dessin. Votre clé pourra alors être gravée en même temps que les deux autres circuits imprimés.

Réalisation du montage

Pour des raisons évidentes de sûreté, le montage a été scindé en deux parties : une première comportant le capteur CNY70 et la LED bicolore de signalisation, la seconde comportant le PIC et le relais commandant la serrure. Les deux circuits imprimés sont présentés **figures 4 et 5**. Les composants sont implantés en respectant les dessins des figures **6 et 7** et en contrôlant la bonne orientation des supports et des composants polarisés, circuits intégrés CI_1 , CI_2 , CI_3 , des LED, régulateur et condensateur C_1 . Il ne faudra pas oublier le condensateur C_3 , placé dans le support du PIC.

Ces deux parties du montage sont reliées par un câble cinq conducteurs muni de deux connecteurs femelles. Les broches des connecteurs K_3 et K_4 sont dans le même ordre identifiable par les signes "-" et "s".

Le schéma de la **figure 8** décrit un exemple de construction d'un lecteur de code-barres. La clé à code-barres glisse dans une fente d'une profondeur de 25 mm. Ainsi, les barres et espaces sont détectés par le capteur CNY70 placé à 15 mm du bord de la fente. Le capteur CNY70 étant orienté vers le bas, la LED L_2 est soudée coté cuivre pour être vu sur la face supérieure. Ce lecteur avec le capteur étant placé sur un mur extérieur, la carte du PIC est placée à l'intérieur, le câble cinq conducteurs enfilé dans une gaine, assurant la liaison.

Réglage et utilisation

Le programme chargé dans le PIC et le lecteur réalisé, il ne reste que le réglage de l'ajustable à effectuer pour rendre notre montage opérationnel. Ce réglage n'a pas besoin d'être extrêmement pointu : la tension sur la broche 3 devant être supérieure à 0 et inférieure à 3 V, le plus simple sera de mettre le curseur de

l'ajustable en position centrale et de reprendre éventuellement ce réglage par la suite. Vous devez constater que si vous placez votre clé contre le capteur, la LED L_1 reste allumée quand il n'y a pas de réflexions lumineuses sur une barre et s'éteint dans le cas contraire. Si c'est le cas (et il n'y a aucune raison que cela ne le soit pas) et si la LED L_2 a clignoté 5 fois en teinte orangée à la mise sous tension, c'est que le programme fonctionne et il ne reste plus qu'à essayer la clé. Faites-la défiler rapidement **dans le sens de la flèche** devant le capteur, la LED L_2 doit passer au vert et le relais s'enclencher pendant 10 secondes. Il ne faut pas hésiter à effectuer un défilement très rapide, le code ne pouvant être reconnu si le défilement est trop lent (voir paragraphe "le programme du PIC" : le défilement doit durer 1 s au maximum). Si cet article manque par endroit de clarté, n'hésitez pas à contacter l'auteur pour quelques renseignements complémentaires: alain.reboux@wanadoo.fr

A. REBOUX

Nomenclature

CI_1 : PIC16F84 + support 18 broches

CI_2 : UA741 + support 8 broches

CI_3 : CNY70

R_1, R_2 : 1 k Ω

marron, noir, rouge

R_3, R_4, R_5 : 430 Ω

jaune, orange, marron

R_6 : 22 k Ω

rouge, rouge, orange

R_7 : 4,7 k Ω

jaune, violet, rouge

R_8 : 2,2 k Ω

rouge, rouge, rouge

C_1 : 10 μ F

C_2 : 100 nF

C_3 : 22 pF

D_1 : 1N4001

D_2 : 1N4148

L_1 : LED ROUGE

L_2 : LED BICOLORE

REG : Régulateur 7805

T : 2N2222A

REL : relais 5 V

K_1 : bornier 2

K_2 : bornier 3

K_3, K_4 : connecteurs mâles 5 broches

Câble 5 conducteurs + 2 prises femelles 5 broches.

Construisez et lancez des fusées à eau

Écologique
Spectaculaire
À la portée de tous

CONSTRUISEZ ET LANCEZ DES FUSÉES À EAU

Ivan Lanoë

Écologique
Spectaculaire
À la portée de tous

ETSF

Prenez un bouchon de liège, une valve de chambre à air, une bouteille en plastique ainsi qu'une pompe à vélo et lancez-vous dans la fabrication de votre première fusée à eau !

Pour cela, il n'y a rien de plus simple, il vous suffit de vous laisser guider par l'auteur, qui vous accompagnera étape par étape dans cette réalisation.

Bien sûr, le charme de cette activité pour le moins ludique opérant, vous ne vous arrêterez pas là et progresserez rapidement dans la maîtrise des fusées à eau : vous apprendrez à déclencher le départ au moment voulu, à agir sur la trajectoire de vol, à améliorer les performances de l'engin, etc.

Pour toute personne désireuse de se lancer dans la réalisation et l'exploitation des fusées à eau, cet ouvrage aux potentiels pédagogiques et ludiques indiscutables est l'outil idéal.

Ivan Lanoë
ETSF-Planete Sciences
136 pages - 22,50 €