

**Lecteur de codes à barres : approche matérielle et logicielle unifiée**

L. Bigué et M. Hassenforder

École Supérieure des Sciences Appliquées pour l'Ingénieur – Mulhouse  
12 rue des frères Lumière  
F-68093 Mulhouse CEDEX, France

# LECTEUR DE CODES A BARRES : APPROCHE MATERIELLE ET LOGICIELLE UNIFIEE

L. Bigué\*, Ecole Supérieure des Sciences Appliquées pour l'Ingénieur – Mulhouse, 12, rue des frères Lumière, 68093 Mulhouse cédex. L.Bigue@uha.fr

M. Hassenforder, Ecole Supérieure des Sciences Appliquées pour l'Ingénieur – Mulhouse, 12, rue des frères Lumière, 68093 Mulhouse cédex. M.Hassenforder@uha.fr

## Résumé :

Nous proposons de décliner une manipulation de lecture de codes à barres en diverses variantes pour aborder plusieurs problématiques. L'approche globale est orientée objets. Chaque classe, correspondant à un capteur, un traitement ou un effecteur, peut être mise en œuvre selon diverses techniques. Les étudiants peuvent ainsi découvrir la programmation sur PC à haut niveau ou le ciblage de microcontrôleurs à partir de code assembleur, C ou C++. De manière marginale, ils peuvent aussi aborder la communication USB et irDA ainsi que l'utilisation de matrices textes LCD. Suivant la complexité du matériel utilisé, on peut corser la difficulté du sujet en proposant des codes à barres numériques ou alphanumériques, de longueur fixe ou variable.

**Mots clés :** génie informatique, approche orientée objets, ciblage, interfaçage.

## 1 INTRODUCTION

L'ESSAIM est une école d'ingénieur universitaire qui forme des ingénieurs en automatique et génie informatique. En deuxième année, les étudiants suivent une dizaine de séances de travaux pratiques de génie informatique.

La première maquette de lecteur de code à barres à l'ESSAIM date de 1997. Il s'agissait alors simplement de décoder des codes numériques UPC-A [1,2] à l'aide d'un crayon optique relié au port parallèle d'un PC. Depuis, la manipulation s'est étoffée et diversifiée au gré des années en utilisant des cibles a priori destinées à d'autres maquettes de TP. Elle impose une approche haut-niveau du système (programmation partiellement ou totalement orientée objets) bien que nécessitant une interaction forte avec le matériel. L'aspect très quotidien bien que mal connu des codes à barres suscite un vif intérêt des étudiants, d'autant plus que le renouvellement annuel des configurations de la maquette et du sujet permet de limiter les tentatives de « copier-coller ».

## 2 APPROCHE GLOBALE DE LA MANIPULATION

Cette manipulation fait intervenir un capteur (« lecteur »), des traitements (« analyseur » et « décodeur ») et un effecteur (« afficheur »). La Figure 1 nous donne un aperçu des déclinaisons possibles de cette manipulation. En pratique, il s'agit suivant les années de mettre l'accent sur l'aspect « traitements » (code alphanumérique EAN 128 [1,3] de longueur variable, par exemple) ou sur l'aspect « lecteur », bas niveau.

## 3 SUJETS D'ETUDE ABORDES DANS CETTE MANIPULATION

### 3.1 Lecteur

Le lecteur de codes à barres le plus simple se compose d'un crayon optique (émetteur et récepteur) Hewlett Packard HBCS-A300 ou similaire. Il délivre simplement un niveau logique haut en présence d'une barre noire, pour peu que l'on déplace le crayon dans une plage de vitesse adéquate. Une utilisation simple consiste à le connecter au port parallèle d'un PC ou à un port d'entrée d'un microcontrôleur.

Pour faciliter la mise au point du TP, nous avons développé un lecteur qui réalise l'acquisition à partir de données provenant d'un fichier texte. Cette approche permet de soumettre facilement et de façon répétitive des vecteurs de tests très différents selon les besoins.

Suivant le type de codes à barres considéré, le lecteur logiciel doit être plus ou moins performant : à l'instar d'EAN 128, la longueur des codes lus peut ne pas être fixée à l'avance (par la norme) et être indiquée dans le code à barres lui-même. Dans ce cas-là, elle ne sera connue qu'après décodage. Le lecteur logiciel devra donc déterminer de lui-même quand il estime avoir terminé la lecture du code à barres.

Cette année a vu la mise en œuvre d'un crayon optique USB. Il s'agissait, à l'aide d'un microcontrôleur PIC 16C745 (voir § 4.3), mis en œuvre au sein d'un kit de démonstration du commerce, de lire les informations issues du crayon et de les envoyer sur un bus USB (un lien série classique ou infrarouge est également possible). Si le protocole choisi pour l'envoi des données, conçu de toute pièce par l'utilisateur ou correspondant à une

norme de fait (du type « USB barcode reader ») impose l'envoi des données représentant les largeurs des barres (les temps de lecture, en fait) en virgule fixe, par exemple sur 8 bits, l'affaire se complique : il faudra au préalable ajuster judicieusement, en fonction de la vitesse de lecture du code, les dits temps, de manière à utiliser préférentiellement les bits les plus significatifs.

### 3.2 Traitements (analyseur et décodeur)

Suivant la complexité de la gestion du matériel proposé au cours de la manipulation, nous demandons aux étudiants de décoder soit des codes numériques de longueur fixe (UPC-A), soit des codes alphanumériques de longueur variable (EAN 128 par exemple).

Nous leur imposons une approche objet basé sur une modélisation UML puis un codage C++, et lorsque le processeur ciblé est très élémentaire un codage en langage C.

Les codes à barres à une dimension considérés dans ces manipulations ne soulèvent pas en première approche de difficulté majeure du point de vue de leur analyse ou de leur décodage, rien qui ne nécessite en tout cas de connaissance particulière hormis celles requises dans l'ensemble des manipulations de TP de génie informatique. Suivant que l'on se propose de découvrir plus les aspects matériels ou logiciels, on pourra choisir lors de la phase de décodage d'aborder ou non le système de contrôle d'erreur utilisant généralement le dernier caractère (hors caractère de fin) du code à barres.

Un autre aspect intéressant, qui oblige les étudiants à bien concevoir leur architecture logicielle, consiste à

ne pas restreindre la lecture des codes à barres de gauche à droite, ainsi qu'ils le feraient naturellement. Ainsi, la lecture « à l'envers » de codes non symétriques du type EAN 128 pourra s'avérer délicate si des structures de données adéquates (vecteur avec itérateur, table, ...) ne sont pas utilisées.

### 3.3 Effecteur

L'affichage peut s'effectuer soit sur une console standard (PC), soit sur un afficheur LCD texte ou graphique, soit sur un assistant personnel (du type Palm™ Zire par exemple) via une liaison USB ou IrDA.

## 4 EXEMPLES CONCRETS DE SUJETS PROPOSES

### 4.1 A l'aide d'un PC sous Windows™

Il s'agit d'une approche intégralement haut niveau avec un codage en C++. Les PC fonctionnant sous Windows™ NT ou 2000, il nous a fallu à l'origine développer un pilote générique d'entrées-sorties pour la plage ISA afin de pouvoir acquérir l'information en provenance du crayon optique. Depuis, un pilote similaire performant et peu onéreux est apparu sur le marché [4]. L'écran du PC peut également servir à l'affichage, une simple console texte standard (mode DOS) évitant que les étudiants se perdent dans la programmation d'une interface graphique Windows. L'utilisation d'une matrice texte LCD est également possible via une carte d'interface ISA, conçue et fabriquée à l'ESSAIM, et le pilote générique évoqué ci-dessus.

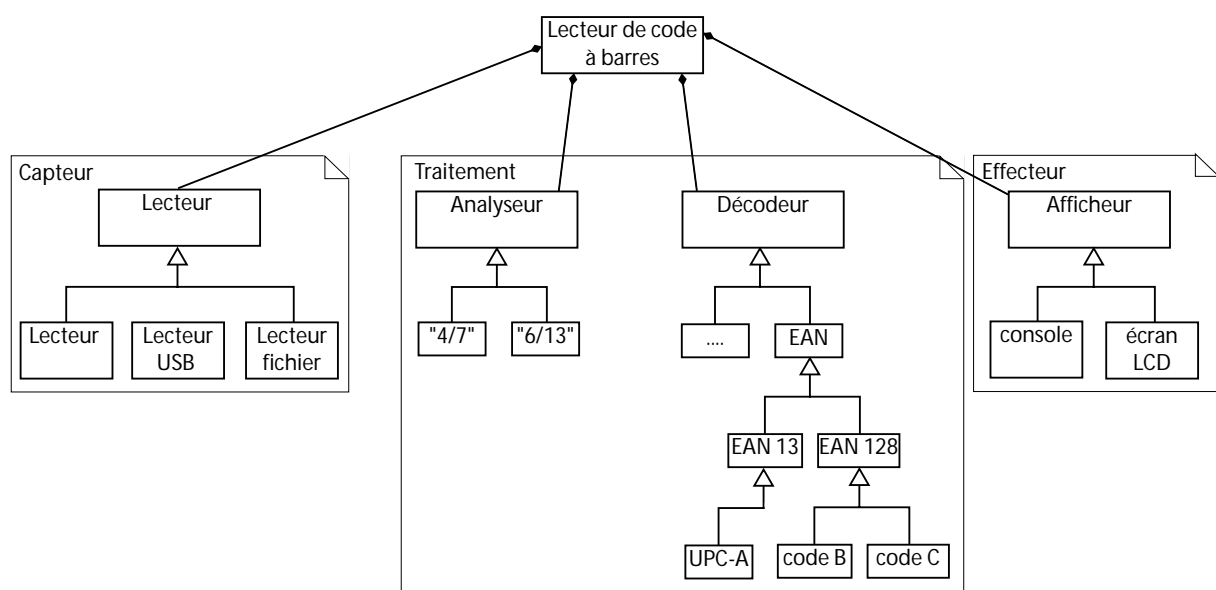


Fig.1 : diagramme UML [5] reflétant l'approche orientée objets et les variantes possibles autour du TP « lecteur de code à barres »

#### 4.2 A l'aide d'un microcontrôleur Infineon™

La mise en œuvre de la manipulation est également intégralement possible sur microcontrôleur Infineon™ 51 (8 bits) ou 16x (16 bits) intégré au sein d'une chaîne logicielle C [6] ou C++ [7]. Des cartes de développement performantes sont disponibles dans le commerce pour un coût modéré [8]. L'affichage peut s'effectuer sur une matrice LCD texte. Un émulateur matériel Hitop teletest 51 peut compléter le dispositif.

#### 4.3 A l'aide d'un microcontrôleur Microchip PIC™

La mise en œuvre de la manipulation est également possible sur microcontrôleur Microchip PIC™ 8 bits du type 16C745 [9]. Ce microcontrôleur, uniquement disponible en version UV-EPROM sur des cartes de développement commerciales [10] ou à intégrer soi-même, se programme en assembleur grâce à l'environnement de développement MPLAB fourni par le constructeur. L'implantation de calculs flottants permettant de réaliser facilement les calculs de normalisation, quoique possible, s'avère lourde et peu pédagogique quand il s'agit de découvrir (ou presque) l'assembleur en une séance de TP. Nous proposons plutôt d'envoyer le résultat de la lecture (fourni par le crayon via le PIC) vers un PC via une liaison USB (à programmer) ou un assistant personnel (du type Palm™ Zire) via une liaison infrarouge irDA. Dans les deux cas, le traitement (analyse et décodage) peut se programmer à haut niveau en C ou C++. Vu la diversité des matériels utilisés dans ce cas, on se contentera de codes à barres simples UPC-A.

### 5 CONCLUSION

La lecture de codes à barres nous fournit l'occasion d'explorer, à travers une approche orientée objets, une variété d'implantations et de traitements. Elle permet de balayer un large spectre de problématiques rencontrées en génie informatique, de l'implantation des traitements jusqu'à

l'interfaçage avec le capteur et l'afficheur. Suivant le sujet et le matériel choisi pour la manipulation, on codera en assembleur, C ou C++. La configuration minimale pour chaque manipulation ne requiert qu'un investissement minimal, dans du matériel aisément recyclable pour d'autres maquettes. Les seuls points communs entre les différentes manipulations demeurent l'approche orientée objets et le crayon optique lui-même.

### Bibliographie

1. Barcode Island, "Barcode Symbolgies", <http://www.barcodeisland.com/symbolgy.phtml> (2000).
2. Norme ISO/IEC 15420, "Technologies de l'information -- Techniques d'identification automatique et de capture des données -- Spécifications pour les symboles des codes à barres -- EAN/UPC" (2000).
3. Norme ISO/IEC 15417, "Technologies de l'information -- Techniques d'identification automatique et de capture des données -- Spécifications pour les symboles des codes à barres -- Code 128" (2000).
4. Ingenieurbuero Paule, Direct I/O device driver, <http://www.direct-io.com/> (2002)
5. P.-A. Muller, "Modélisation objet avec UML", Eyrolles, Paris, 421 p., 1<sup>ère</sup> édition (1997).
6. Keil, CA-51 compiler kit, <http://www.keil.com/c51/ca51kit.htm> (2003)
7. Tasking, C166 Software Development Tools, <http://www.tasking.com/products/C166-ST10/c166-ds.pdf> (2003)
8. Phytec, MiniMODULE-537 et MiniMODULE-167, <http://www.phytecfrance.com/> (2003).
9. Microchip, PIC16C745/765, 8-Bit CMOS Microcontrollers with USB, 164 p, <http://www.microchip.com/download/lit/pline/pimicro/families/16c7xx/41124c.pdf> (2000).
10. Microchip, PICDEM™ USB User's Guide, 80 p, <http://www.microchip.com/download/tools/pimicro/demo/pdemusb/41174a.pdf> (2001).