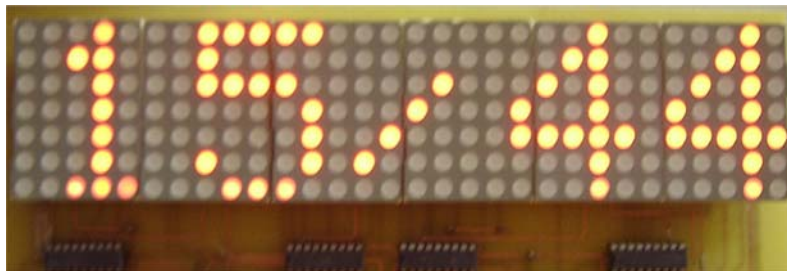


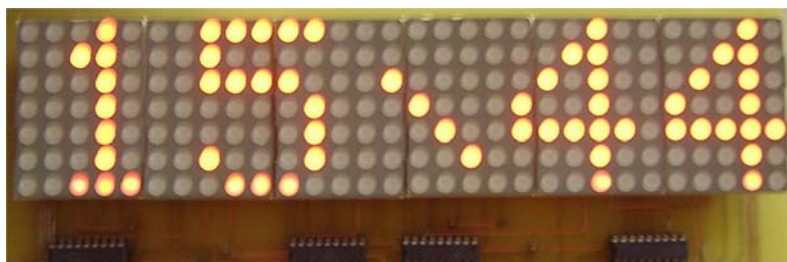
Écriture Lumineuse

Nabil AL-HOSSRI, département GEII IUT Bordeaux1
E-mail: nabil.al-hossri@u-bordeaux1.fr

TIC TAC



/ : il est 15 :44 :00



\ : il est 15 :44 :01 (une seconde de plus)
Et on entend « tic tac » à chaque seconde.
Normal, non ?

Voir vidéo [ici](#)



Pré-requis et liens indispensables

Ce projet sera piloté par un microcontrôleur type MC68HC908GP32 monté en mode moniteur, associé à une interface de visualisation équipée de six matrices géantes de 7x5 LEDs.

Tout écrasement (mettre à blanc) ou reprogrammation de la mémoire Flash du micro sera effectué en mode dit *in-situ*. Ces opérations demandent l'utilisation d'un programmeur Freescale HC08 disponible dans la salle.

La figure 1 illustre les connectiques détaillées qu'on peut adopter entre les éléments : PC, programmeur, micro et LCD.

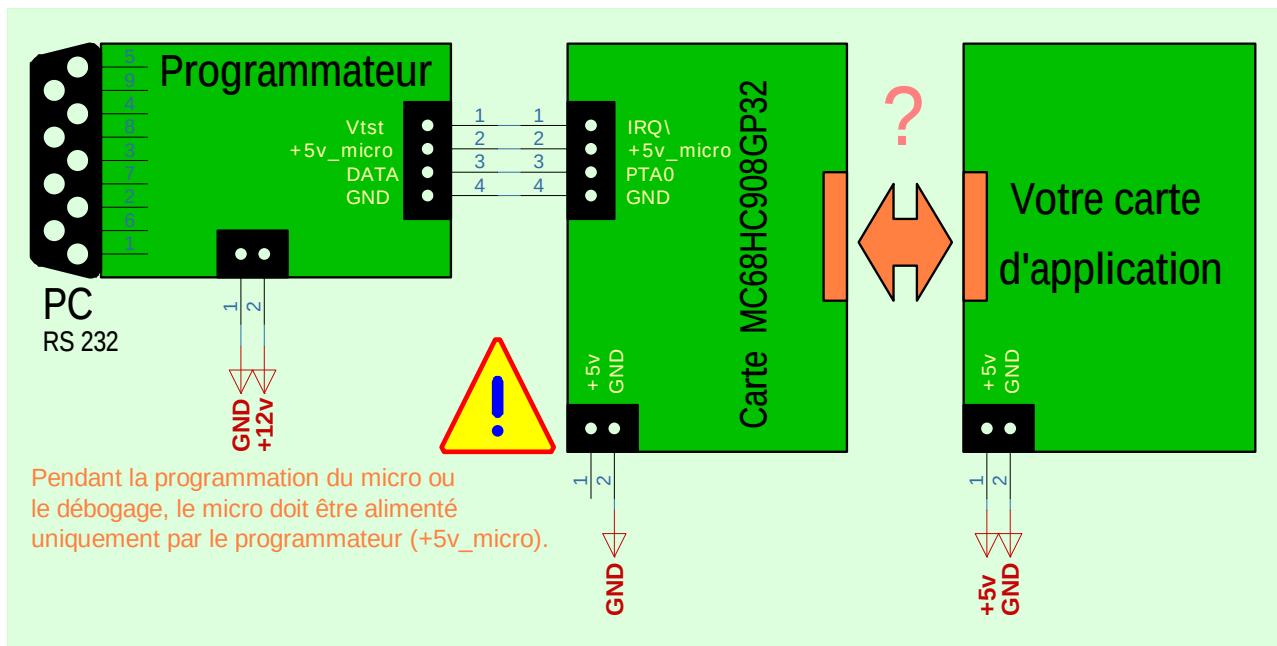


Figure 1 Illustre les connexions qu'on doit adopter entre le programmeur et la carte micro.

Présentation du projet

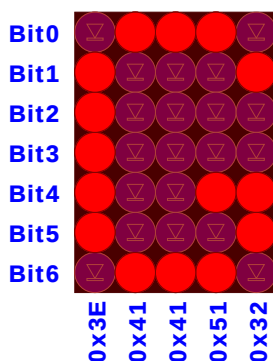
Il existe actuellement une variété innombrable d'écritures lumineuses pour vitrines, ou panneaux indicateurs, à contenu variable, ou tout simplement destinées à attirer l'attention. Les appareils se prêtant à cet emploi sont toutefois relativement coûteux et leur réalisation dépasse le niveau du simple bricolage. Le projet d'écriture lumineuse présenté ici, simple à réaliser qu'à utiliser, présente en outre l'avantage d'être peu coûteux.

Ce projet répond aux critères de réalisation et de fonctionnement évoqués précédemment et a l'avantage d'être réalisable avec des composants disponibles dans notre département.

Les caractères sont représentés par des matrices de 7x5 LEDs à anode commune . Comme il n'est pas possible de commander simultanément toutes les diodes lumineuses, il faut les allumer successivement par multiplexage de façon à ce que l'utilisateur ne s'aperçoive de rien. Le nombre de LEDs qui s'allume simultanément est limité à 7, c'est-à-dire une colonne.

L'inertie de notre vision nous fait voir une image unique de $7 \times 30 = 210$ points (lorsqu'on utilise 6 matrices juxtaposées).

Affichage



Le texte à écrire est stocké dans la mémoire *FLASH* du microcontrôleur. Pour afficher par exemple la lettre G, on fait appel à une [table de conversion](#)  pour représenter la lettre sur une matrice 7x5, comme le montre la figure ci-contre, de sorte que, dans le cas de la lettre G, on obtient les valeurs : 0x3E, 0x41, 0x41, 0x51 et 0x32 qui seront attribuées successivement aux colonnes.

Le microcontrôleur

Le microcontrôleur MC68HC908GP32 de *Freescale* utilisé est parfait pour ce projet :

- mémoire FLASH de 32 Ko,
- 512 octets de RAM,
- deux temporisateurs de 16 bits qui peuvent fonctionner soit en mode capture en entrée, soit en mode comparaison en sortie,
- un nombre de PORTs d'entrées/sorties suffisant,
- une source d'interruption externe : capture de front en entrée.

Son rôle, dans ce projet, est :

- de générer, par son [module PWM](#) , l'horloge du compteur,
- l'envoi des caractères avec le balayage des colonnes,
- la détection, par son [module Input Compare](#) , de la fin de balayage des colonnes.

La figure 2 montre le schéma structurel de niveau 1 ainsi conçu.

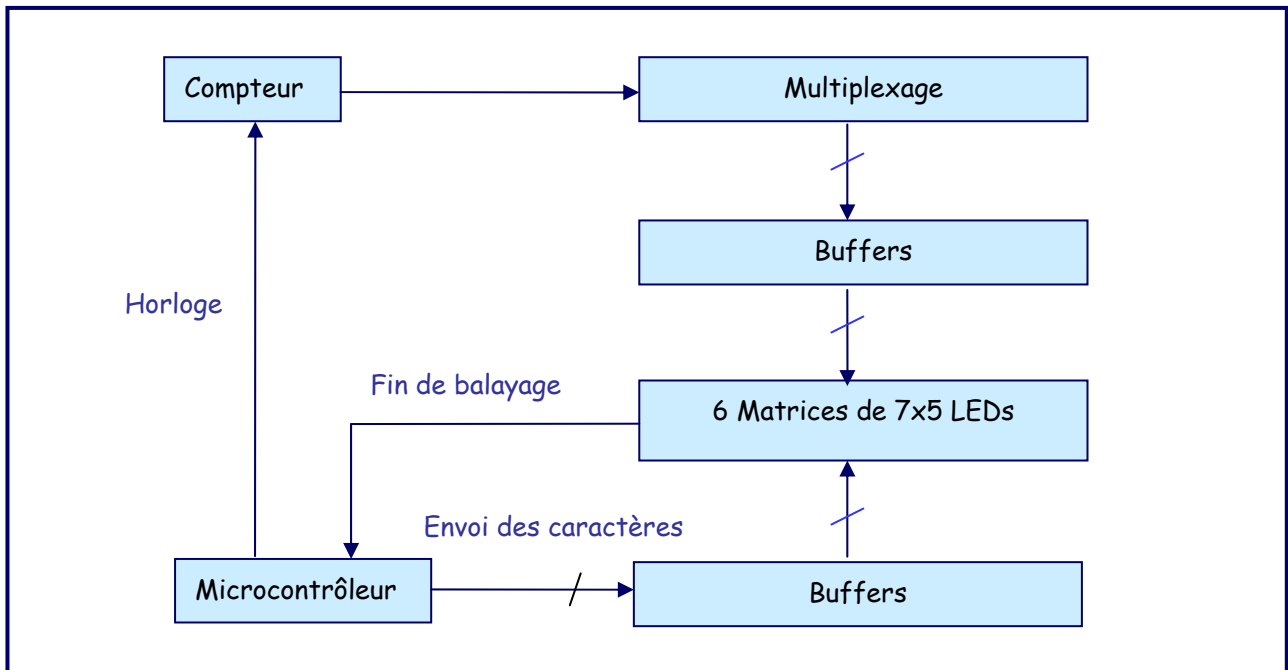


Figure 2 Schéma structurel de niveau 1

Etude théorique

- 1- Expliquer la signification et le mécanisme d'une interruption, suite à une capture en entrée, dans un programme.
- 2- Documenter et résumer le fonctionnement du module capture de front en entrée ([Input Capture](#) ) du microcontrôleur [MC68HC908GP32](#) .
- 3- En examinant l'illustration de la figure (3), expliquer globalement le fonctionnement du système.
- 4- Proposer un schéma électrique fonctionnel détaillé qui peut aboutir à la réalisation des différentes fonctions.
- 5- Calculer la fréquence minimale acceptable pour avoir un balayage produisant une image stable.
- 6- Calculer les valeurs des différents composants.
- 7- Calculer la puissance utile maximale consommée pour avoir une luminosité des LEDs suffisante.

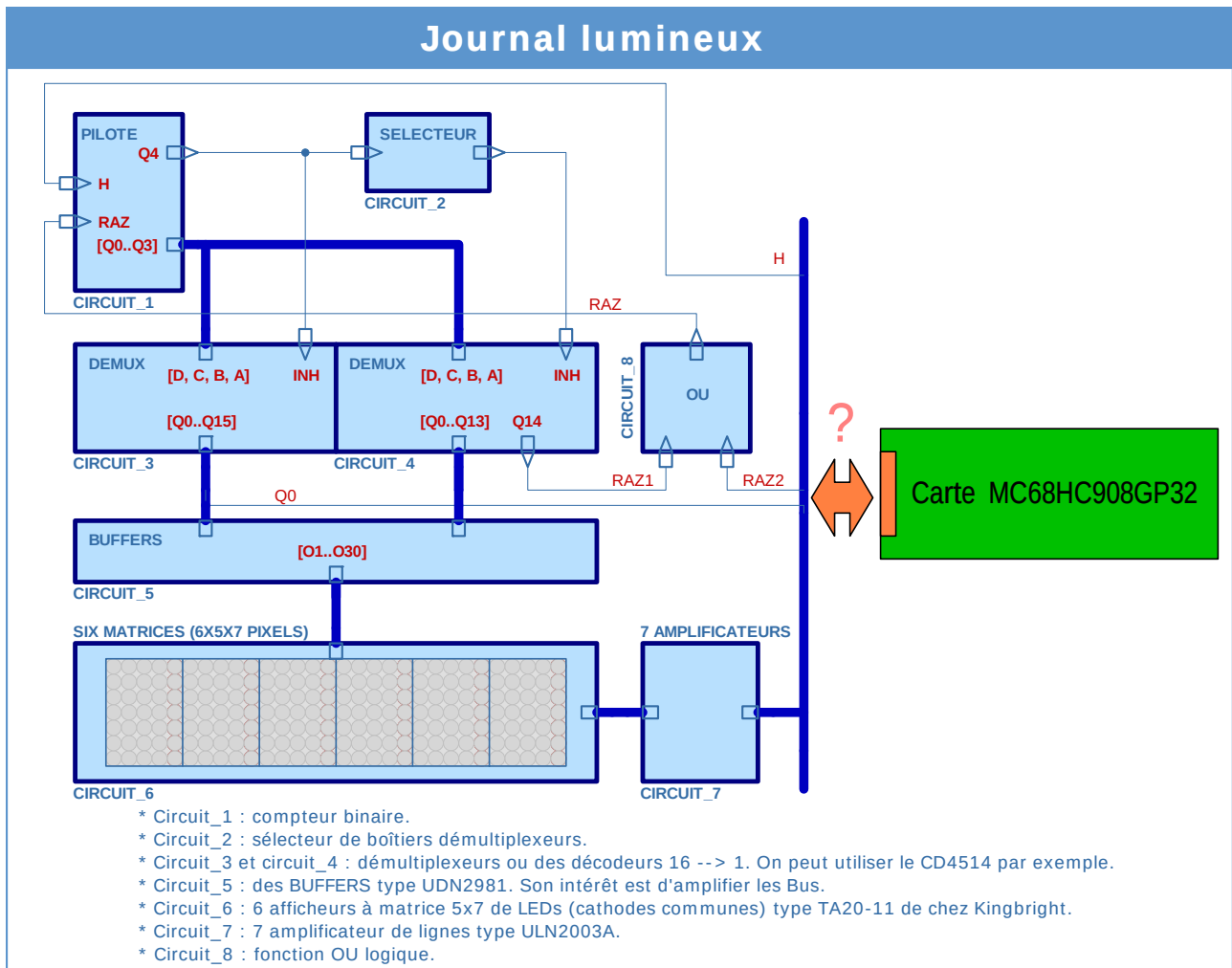


Figure 3 Schéma bloc, de niveau 2, de la carte de visualisation.

Réalisation

- 1- Réaliser la carte micro monté en mode moniteur dont le schéma est donné [ici](#) .
- 2- Faire un schéma sous ISIS permettant une simulation complète du balayage qui gère l'affichage sur les matrices à LEDs. Aidez-vous de l'article exposé [ici](#) .
- 3- Le montage d'affichage sera réalisé sur une carte double-face en prévoyant des connecteurs le reliant à la carte microcontrôleur.
- 4- Souder les composants et les supports des circuits intégrés.
- 5- Faire vérifier le bon fonctionnement du système.

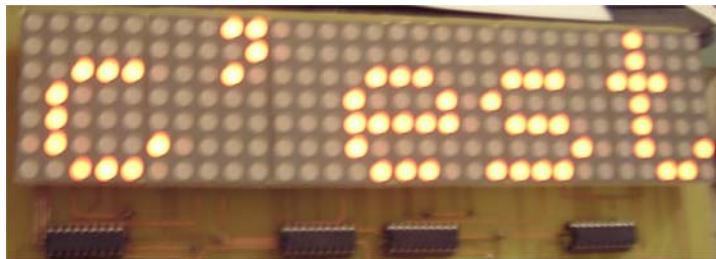
Programme

Le langage de programmation est le C sous le compilateur CODE WARRIOR version 5.1. Les démarches à suivre pour ouvrir un nouveau projet personnalisé avec ce compilateur sont exposées [ici](#) .

Un programme de test de la carte d'application qui affiche le texte défilant suivant :

L'électronique c'est l'avenir. C'est la raison pour laquelle nous sommes au département GEII. N'est-ce pas?

est prévu [ici](#) .



Voir vidéo [ici](#)



Pour mieux comprendre, expliquer et commenter ce programme de test, aidez-vous des bibliothèques de fonctions gérant :

- o la capture de fronts en entrée ([Input compare](#) ) ,
- o le [PWM](#)  (*Pulse Width Modulation*).

Par ailleurs, pour mieux le maîtriser, on vous demande de modifier le programme de test pour qu'il affiche la pendule décrite en tête de ce projet.

Rappel : vidéo [ici](#)



Télécharger les composants additionnels utiles pour la simulation sous ISIS



Important pour mener à bien votre projet et gagner du temps

Après réalisation du circuit imprimé, bien vérifier la continuité des pistes à l'ohmmètre ; il vaut mieux passer une demi-heure à cette opération que trois heures à dépanner l'appareil par la suite... Si possible, étamer le circuit pour réduire le risque de micro coupures.

Enfin, souder les composants et avant de passer au programme final, faire un petit programme pour tester le bon fonctionnement du montage.