

LE 21^e VIRUS INFO

**Les Ipods
dangereux !**

**Père-Noël,
la fin ?**

**Windows
interdit en
Grèce ?**

**Fabriquez
votre
console !**

**Nouvelles
consoles**

**Les tests exclusifs
de Shinco DVD 868,
Nuon 504, GP32,
Piece, WS Crystal...**

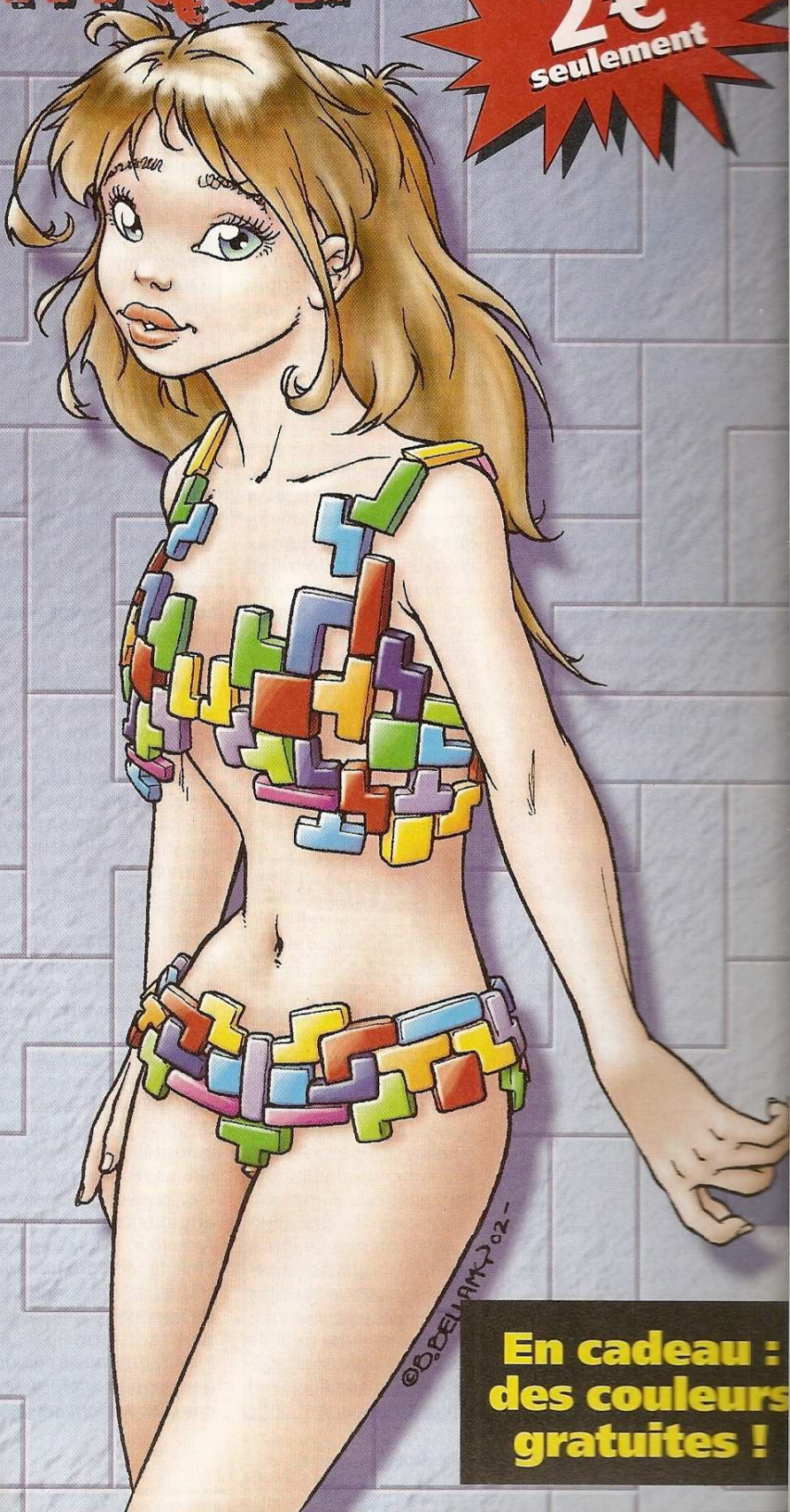
Page 21, les briques tombent ! •
Novembre 2002 • Trimestriel •
Imprimé en / Printed in France •
13,10 FF • 3,60 FS • 80 FB •
80 FL • 1400 CFA • Pol. Fr.
320 XPF • Nelle Cal. 300 XPF
• 2,95 \$Can • 23 DH • 4500 LB

Les VRAIS tests de toutes les consoles !

PlayStation 2 - Xbox - GameCube - Game Boy Advance...

FORMATIQUE

**2€
seulement**



**En cadeau :
des couleurs
gratuites !**

M 05847-21-F: 2,00 € - RD



Fabriquez une (vraie) console de jeux !

De nombreux magazines vous ont déjà proposé de taper des listings de jeux vidéo. Tetris étant un exemple classique, qu'on ne présente plus. Cette fois, nous irons plus loin car nous vous proposons de réaliser votre propre console pour une poignée d'euros. Carrément !

Raft (gros merci à Rickard Gunée !)

Le secret réside dans le microcontrôleur utilisé : il assure toutes les fonctionnalités du système, y compris l'affichage, qu'il est chargé de générer à la volée. C'est pourquoi avant de se lancer dans cette aventure, un petit rappel sur le fonctionnement de la vidéo s'impose (pour ceux que cela intéresse). Un faisceau d'électrons est envoyé dans un tube à vide sur la surface d'un

Dans l'autre procédé, dit Vector, le faisceau d'électrons est envoyé uniquement sur les zones de l'écran que l'on souhaite éclairer. Ce procédé est très peu utilisé dans des applications grand public puisqu'il est incompatible avec les écrans classiques (le fabricant du jeu étant alors obligé de fournir un écran dédié, comme pour la console Vectrex, par exemple). Mais revenons à nos télévisions. Dans ce cas, la

qu'en Europe une image Pal est composée de 625 lignes, le signal Sh aura effectué 625 oscillations pendant que Sv aura effectué celle relative à toute la hauteur de l'écran. Bien sûr, lors des redescendentes des deux signaux correspondant au retour du spot à la ligne (Sh) et en haut de l'écran (Sv), le faisceau d'électrons est coupé pour ne pas laisser de trace de son passage !

L'entrelacement et la luminance

La fréquence de rafraîchissement de l'image est conçue pour que le spectateur ne perçoive pas le balayage, elle doit donc être supérieure à la persistance rétinienne. En France, la fréquence de trame retenue est de 50 Hz qui correspond à celle du courant EDF. En réalité, pour éviter le scintillement, l'écran complet est affiché en deux passages. Une première trame dessine les 313 lignes impaires de l'image, et la seconde les 312 lignes paires. Une image construite de cette manière, une ligne sur deux, est dite « entrelacée ». Le rafraîchissement de l'image complète se déroule donc tous les $1/25^e$ de seconde. Il en découle la fréquence de ligne, soit $625 \times 25 = 15,625$ kHz. Maintenant que nous savons comment le spot se déplace, voyons comment l'intensité lumineuse est modulée pour afficher l'image tant attendue. La figure 3 représente un

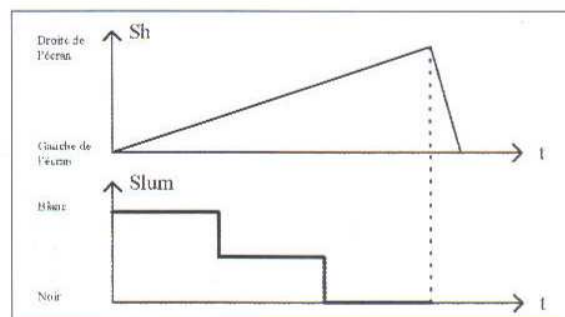


Figure 3 : Exemple de luminance

exemple de balayage d'une ligne. L'intensité du point lumineux affiché à un instant t est proportionnelle au signal de la luminance Slum. La ligne affichée dans cet exemple sera donc composée de trois segments successifs, un blanc, un gris, puis un noir. Le faisceau parcourra ensuite la ligne du dessous et, ainsi de suite, pour afficher l'image complète !

le PIC utilisé dans notre montage ne permet pas de générer un signal couleur en temps-réel.

Fonctionnement de la console

De son côté, un jeu vidéo, quel qu'il soit, doit se conformer aux principes énoncés ci-dessus pour envoyer au récepteur l'image à afficher. Pour définir la position du

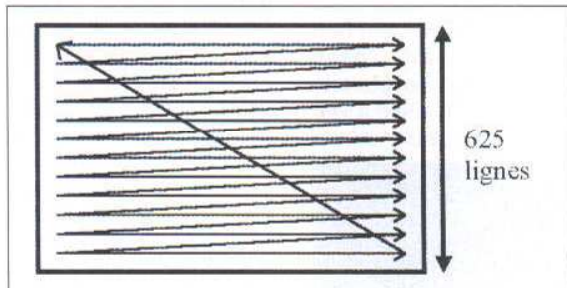


Figure 1 : Déplacement du spot vidéo

écran phosphorescent. Deux principes différents peuvent être employés pour construire l'image. La grande majorité des systèmes utilise la technique Raster qui consiste à balayer toute la surface de l'écran, de gauche à droite, ligne par ligne, jusqu'en bas (fig. 1).

déviations du faisceau est commandée par deux champs électriques, l'un vertical, l'autre horizontal. Pour simplifier, en considérant que tout ce dispositif est linéaire, les signaux électriques générant ces champs vont prendre la forme de rampes (fig. 2). Et sachant

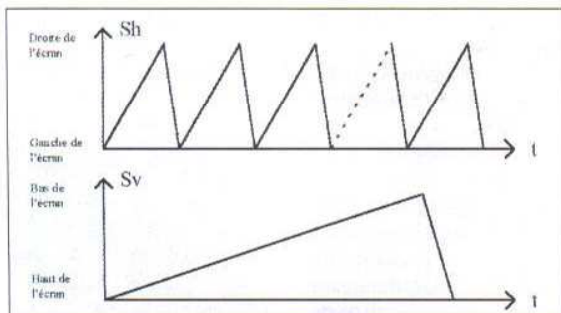


Figure 2 : Rampes de commande du raster

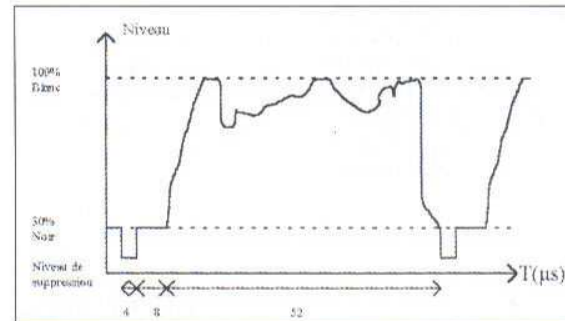


Figure 4 : Vidéo composite

En ce qui concerne les écrans en couleurs, les mêmes principes sont appliqués à trois canons à électrons, un rouge, un vert et un bleu. Ils reproduisent toutes les couleurs à l'écran par trichromie additive. Arrêtons-nous à l'étude du noir et blanc, puisque

spot, il envoie à la télévision des signaux de synchronisation sur lesquels les oscillateurs des rampes du récepteur vont se caler. Il s'agit donc bien de mettre en phase les deux systèmes du point de vue temporel. Ceci fait, le jeu n'a plus qu'à moduler les intensités lumineuses au

LES JEUX VIDÉO SONT-ILS POUR LES ENFANTS?



moment opportun pour allumer un point précis à l'écran. Le signal de synchro horizontale indique le début d'une ligne. Lorsqu'il est actif, il détermine le moment où la rampe H doit revenir à zéro. Le signal de synchro

Le niveau du noir se situe à 0,3 V, le blanc à 1 V, les valeurs intermédiaires se traduisant en niveaux de gris à l'écran. Les synchros correspondent, quant à elles, à 0 V.

Le hardware

Dans le montage présenté aujourd'hui, pratiquement tout est réalisé par le programme d'un microcontrôleur PIC de chez Microchip, le 16F84-10. Le hardware se réduit donc à sa plus simple expression ! Contrairement à la technique communément utilisée consistant à stocker l'image à reproduire dans une mémoire, puis à la faire afficher par un circuit spécialisé, notre montage génère la vidéo en temps réel ! Pour cela, nous avons été obligés d'overclocker un peu le PIC initialement conçu pour tourner

à 10 MHz, afin d'obtenir la puissance de calcul nécessaire. Rassurez-vous, ce composant s'accommodera très bien d'une fréquence de 12 MHz ! Pour une description détaillée de ce circuit, reportez-vous au site du fabricant www.microchip.com/1010/pline/picmicro/category/~digictrl/8kbytes/devices/~16f84/index.htm. Sur le schéma, le signal vidéo se retrouve en sortie des deux résistances connectées aux bits 0 des ports A et B. Voyons comment deux composants aussi simples suffisent à réaliser un convertisseur numérique/analogique délivrant des niveaux adaptés. Sachant que l'impédance d'entrée d'un téléviseur est de 75 Ω , le schéma équivalent du convertisseur est donné figure 6. Le PIC pilote ces deux « interrupteurs » de sortie pour générer *grossomodo* les amplitudes d'un signal composite. En effet, les niveaux de blanc, gris, noir et synchro peuvent être fournis par ce diviseur de tension.

Voici résumées les tensions obtenues en fonction des deux bits de sortie :

RB0	RA0	Calcul (*)	Signal de sortie
0	0	$VTV = 0V$	Synchro
0	1	$VTV = 5 / (1000 * (1/1000 + 1/450 + 1/75)) = 0,3V$	Noir
1	0	$VTV = 5 / (470 * (1/1000 + 1/450 + 1/75)) = 0,64V$	Gris
1	1	$VTV = 0,3 + 0,64 = 0,94 = \text{quasiment } 1V$	Blanc

(*) $VTV = VB * (RB * (1/RA + 1/RB + 1/RTV)) + VA * (RA * (1/RA + 1/RB + 1/RTV))$

Comme vous pouvez le remarquer, le bit RB0 sert à générer la luminance, et RA0 la synchro. À noter que le gris n'est pas utilisé dans cette application. Comme le PIC doit tout faire lui-même, il s'occupe de fournir un signal audio avec le convertis-

seur 1 bit mis en œuvre par les résistances en sortie du port RA4. Eh oui, un signal carré qui oscille, ça fait du bruit !

Les deux sorties, audio et vidéo, sont disponibles sur les deux connecteurs RCA (Cinch) à raccorder au téléviseur avec des cordons adaptés. Si votre écran ne possède pas d'entrée composite, vous pouvez câbler ces sorties à une prise Péritel comme suit :

- Broche 20 : Entrée vidéo composite
- Broche 17 : Masse vidéo composite
- Broche 6 : Entrée audio mono
- Broche 4 : Masse audio

À fond les manettes !

Pour diriger les pièces du jeu, un joystick « classique » doit être branché sur l'une des prises mâles 9 broches. Il vous reste sûrement des manettes de ce type dans le carton de votre vieil Amstrad, C64, Atari, Amiga... Pourquoi deux connecteurs ? Parce

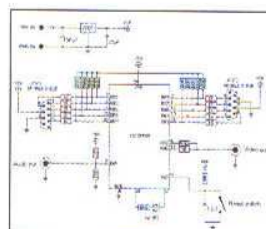


Figure 5 : Schéma fonctionnel du Tetris

afin de positionner ces entrées au niveau haut au repos. Lorsqu'un contact est réalisé avec le joystick, l'entrée associée, ramenée à la masse, passe à l'état bas. Les résistances de 1 k Ω ne sont là que pour éviter un court-circuit éventuel. En effet, le port B est utilisé comme registre à décalage de sortie pendant le balayage du spot, afin de produire un signal vidéo sur RB0. Si cette méthode permet d'obtenir un signal de plus haute qualité (8 caractères par ligne !), elle positionne l'ensemble du port en sortie pendant un moment. L'appui sur un bouton de la manette pendant cet instant provoquerait irrémédiablement un court-circuit si les résistances de 1 k Ω n'étaient pas présentes

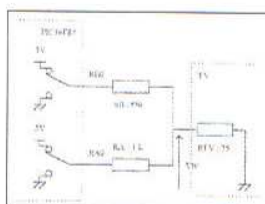


Figure 6 : Sortie Vidéo du Tetris

verticale indique, quant à lui, le début d'une image, soit le moment où la rampe V doit revenir à zéro. Comme nous l'avons vu, une image complète s'affiche en deux fois, lignes impaires puis paires. Le jeu va donc signaler le type de trame qu'il est en train d'envoyer grâce à une séquence particulière dans sa synchronisation verticale.

Il en résulte que, pour un système noir et blanc, seuls trois signaux sont à générer : la synchro horizontale, la synchro verticale et la luminance. Parfois, les deux synchros sont transmises ensemble avec le signal de luminance, on parle alors de vidéo composite. Le schéma 4 montre la forme d'un signal composite positif lors de la génération d'une ligne.

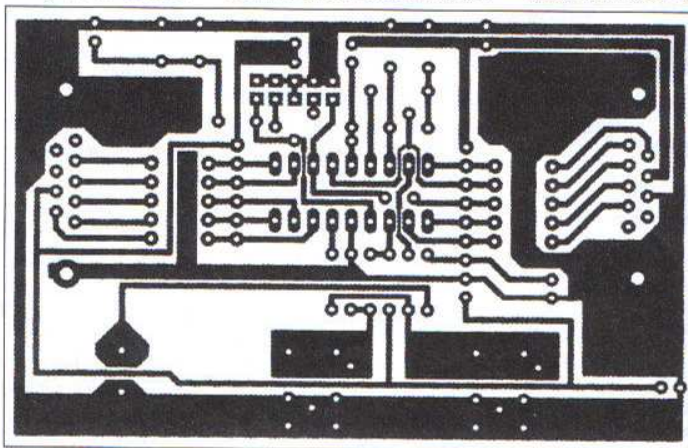


Figure 7 : Typon

pour limiter le courant. Quant au reste du montage, rien de particulier ! L'alimentation du jeu peut se faire avec un bloc de 8 à 18 V. Cette tension est ensuite régulée par le circuit intégré 7805 pour produire les 5 V nécessaires au microcontrôleur. Le reset est réalisé comme d'habitude à la mise sous tension du montage grâce au condensateur et à la résistance. Enfin, un interrupteur vous permet de redémarrer le jeu. Au niveau du coût de réalisation, le circuit imprimé de petite taille ne vous ruinera pas, et les composants électroniques non plus. Comptez environ 8 € pour le PIC 16F84-10. Commencez par implanter les composants passifs en respectant les polarités, puis continuez



Figure 8 : Schéma d'implantation

par le support CI, le régulateur, les connecteurs et le bouton de reset. Voilà, c'est tout !

Le logiciel

Le PIC reste à mettre en place, me direz-vous. Avant cela, il vous faudra y transférer le programme sous forme d'un fichier compilé de codes hexadécimaux.

Pour les courageux que ça intéresse, le code source en assembleur tetris.asm

gramme. On se calera, bien sûr, sur la vidéo, en fonction de la durée des instructions CPU. Cela rappellera des souvenirs aux mordus de cycle counting sur console VCS2600...

L'algorithme principal est exécuté pendant les premières scanlines, alors que le spot n'apparaît pas encore sur l'écran. C'est lors de ces routines que sont testées les positions du joystick et que les pièces sont déplacées. En début de ligne, avant l'affichage du mur, le processeur n'a pas grand-chose à faire. Il en profite pour nous jouer une petite musique qui, malheureusement, par manque de cycles machine, n'est pas de grande qualité. Cette musique est enregistrée

petite mélodie augmente en même temps que celle du jeu !

64 µs chrono !

À part ça, le PIC passe bien le plus clair de son temps à générer de la vidéo et, vous allez le constater, ses 12 MHz ne sont pas de trop ! Comme nous l'avons vu, en Pal, chaque ligne dure 64 µs (1/15626) dont 52 µs sont réellement visibles à l'écran. Sachant que ce microcontrôleur est capable d'exécuter environ trois instructions par µs, on obtient le nombre de cycles utilisables par ligne, soit : $3 \times 52 = 156$. Si l'on considère qu'un pixel peut être envoyé à chaque cycle (ce qui suppose l'utilisation d'une instruction ne durant qu'un cycle), la résolution horizontale maximale est donc de 156 pixels. Pas si mal ! En plus du graphique, le programme doit envoyer à chaque ligne la synchro horizontale pendant 4 µs, puis du noir pendant 8 µs.

Le PIC doit aussi signaler au récepteur le début d'une nouvelle image. Il envoie donc deux types de synchronisation verticale correspondant à une nouvelle trame impaire ou paire. Les séquences appropriées sont présentées figure 11. Dans Tetris, les images des deux trames sont identiques, divisant par deux la résolution verticale. Cette contrainte est loin de porter préjudice à la qualité du jeu puisque la résolution verticale reste cependant supérieure à la résolution horizontale !

Algorithme principal

L'ensemble du graphisme est formé à partir d'une matrice de 16 carrés en largeur sur 16 en hauteur. Cet espace comprend les murs extérieurs, les briques qui tombent, sans oublier la pièce présente à gauche qui doit vous aider à affiner votre stratégie. Ce tableau représente donc 16 x 16 bits en mémoire. Pendant le jeu, le déplacement des blocs est réalisé de la manière

Changez de jeu !

Avec le même hardware, vous pourrez retrouver le plaisir d'un Pong entre amis ! Pour cela, rien de plus simple,

programmez votre PIC avec le fichier pong.hex présent à l'URL suivante www.efdlth.se/~e96rg/mc/video/pong.zip.

suivante : effacement de la pièce qui tombe, tests pour savoir si cette pièce peut descendre, puis affichage à la nouvelle position. Une brique ne peut plus bouger si au moins un des pixels de la future position est occupé. Dans ce cas, une nouvelle pièce est choisie au hasard, et c'est reparti !

Pour les scores, l'affaire est un peu plus complexe. Les points composant la ligne du caractère à afficher sont chargés sur le port B, puis décalés vers la droite à chaque cycle vers le bit RB0. Les bitmaps de chaque caractère sont stockés à l'envers pour apparaître dans le bon ordre à l'écran (fig. 10).

La dernière étape consiste à transférer le fichier tetris.hex dans votre

dérouler sans encombre. Néanmoins, si votre programmeur ne tient pas compte des bits de configurations présents dans le fichier, positionnez les manuellement dans le logiciel de transfert (PicStart Plus, par exemple) comme suit : High Speed sur on, Watch Dog sur off, Power sur off et, enfin, Code Protection sur off.

Comment on joue ?

Ça y est, c'est le moment tant attendu ! Après avoir branché votre joystick dans un des ports, appuyez sur Feu pour débiter la partie. Les briques vont commencer à tomber. À vous de les diriger au mieux, à droite et à gauche, à l'aide de la

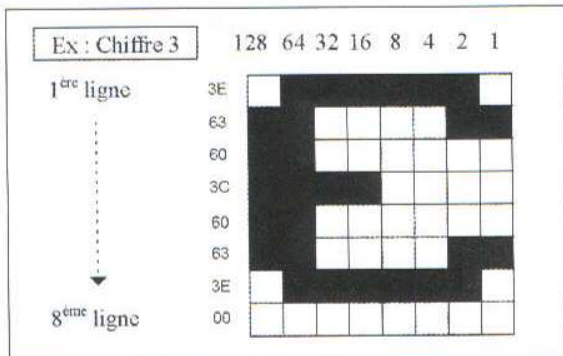


Figure 10 : Définition d'un caractère

est aussi disponible sur le site de son concepteur, en www.efdlth.se/~e96rg/mc/video/tetris.zip. Voyons maintenant comment ce programme fonctionne !

Étant donné que tout est géré en temps réel, il faudra estimer au mieux le temps pour conserver un timing constant quels que soient les chemins empruntés par le pro-

gramme dans l'Eeprom de données sous forme d'une suite de notes compressées. En effet, un seul octet suffit à stocker la hauteur et la durée de chaque note. Ces deux informations sont ensuite transformées avec deux tables de correspondances pour générer le son désiré. Et, histoire de vous stresser un peu plus, la vitesse de cette

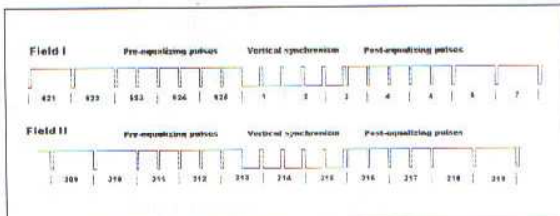


Figure 11 : Pulses de synchronisation verticale

16F84 grâce à un programmeur de PIC. Ce fichier contient à la fois le jeu, qui va se loger dans l'Eeprom de programme (1K x 14 bits), et la musique, qui prendra place dans les 64 octets de l'Eeprom de données.

Si vous n'avez pas accès à ce type de programmeur, il vous reste la possibilité d'en fabriquer un. De nombreux montages sont présentés sur le net, par exemple, sur le site de Microchip (www.microchip.com/download/appnote/devspec/16cxx/00589a.pdf). Sinon, votre revendeur préféré se fera un plaisir de vous en vendre un en kit. Normalement le transfert devrait se

manette. Le but étant de remplir au maximum les espaces pour que le mur n'atteigne jamais le haut de l'écran ! Pour faire coïncider les formes, vous avez la possibilité de tourner la pièce en appuyant sur le bouton. Enfin, si vous êtes pressé, vous pouvez toujours positionner le joystick vers le bas pour accélérer le mouvement. Attention, l'abus de Tetris nuit gravement à la santé !

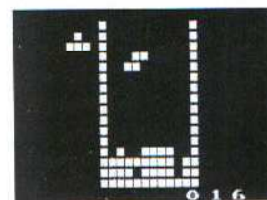


Figure 12 : Tetris en action !

Nomenclature (Figure 9)

- | | |
|--|---|
| 1 PIC 16F84-10 | 2 diodes 1N4148 |
| 1 quartz de 12 MHz | 3 condensateurs de 3,3 µF |
| 1 support CI | 2 condensateurs de 22 pF |
| 1 connecteur de programmation 10 plots (optionnel) | 1 condensateur de 100 µF |
| 1 CI 7805 | connecteurs DSUB |
| 2 réseaux de résistances (5 x 100 kohm) | Mâle 9 broches |
| 11 résistances 1 k 1 | 2 connecteurs CINCH |
| 2 résistances 220 1 | 1 bouton poussoir normalement ouvert |
| 1 résistance de 180 1 | 1 connecteur d'alimentation (en fonction de votre bloc d'alim') |
| 1 résistance de 470 1 | |
| 1 résistance de 10 k 1 | |