

Un circuit d'horloge programmable : le Timer 6840

Le 6840 est un PTM (Programmable Timer Module) qui contient trois compteurs binaires indépendants, commandés et contrôlés par l'unité centrale grâce à leurs registres de commande, d'état, et de données.

Divers modes de fonctionnement permettent, sous le contrôle total du logiciel, de réaliser facilement des appareils tels que fréquencemètre, chronomètre, générateur d'impulsions ou synthétiseurs de fréquences.

C'est dire l'importance de tels composants que tout « micro-informaticien » doit savoir utiliser aujourd'hui.

Le MC 6840 est commercialisé par la firme Motorola (le SFF 96840 de Sescosem en est la seconde source). Chez les autres constructeurs, il existe des circuits analogues quant au fonctionnement et à la programmation comme le 8253 de la famille du 8080 d'Intel.

Souvent, ces timer font partie intégrante des microordinateurs en un boîtier.

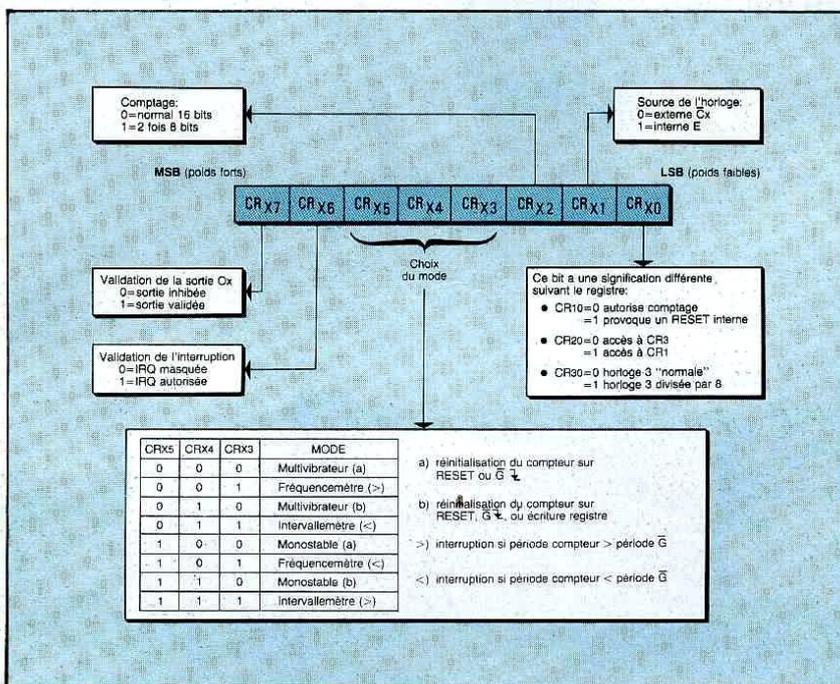


Fig. 1. - Structure du registre de commande CR_X ($X = 1, 2$ ou 3). Son contenu impose le mode de fonctionnement du compteur correspondant. C'est le registre le plus complexe à programmer.

Toutes les durées, périodes ou intervalles de temps engendrés par ce circuit sont définis par le comptage d'un nombre entier de périodes d'horloge, nombre variant de 0 à 2^{16} .

Pour chacun des trois compteurs internes, l'horloge peut être obtenue soit à partir d'un signal externe, soit à partir du signal de validation interne au système (Φ_2).

Une grande souplesse de fonctionnement est apportée par la possibilité de travailler sur 16 bits ou deux fois 8 bits, d'autoriser ou non des interruptions en fin de cycle de chaque compteur, de valider ou non les sorties O_1, O_2, O_3

pouvant engendrer des signaux de formes diverses et programmables.

Le timer 3 possède en plus la possibilité de « pré-diviser » le signal d'horloge par 8, ce qui est utile lorsque l'on veut mesurer des temps très longs. L'encadré 1 donne le synoptique et le brochage du timer et, décrit les différents signaux échangés.

Description interne

Chacun des trois timers se compose d'un compteur 16 bits et d'un registre de chargement où est

stockée la valeur d'initialisation du compteur.

L'arrivée d'un ordre d'initialisation (dépendant du mode de fonctionnement) provoque le positionnement du compteur à la valeur préchargée dans le registre. Le compteur, s'il est validé, se met ensuite à décompter jusqu'à zéro. Le positionnement d'un bit (« drapeau ») dans le registre d'état signale alors la fin du décomptage (« TIME OUT »).

Le registre de commande associé à chacun des timers, CR_X ($X = 1, 2$ ou 3) est accessible en écriture seulement et son contenu impose le mode de fonctionnement du compteur correspondant (fig. 1).

Un registre d'état (Status Register) unique collecte les informations des trois compteurs. Il est accessible, en lecture seulement, par l'unité centrale (fig. 2).

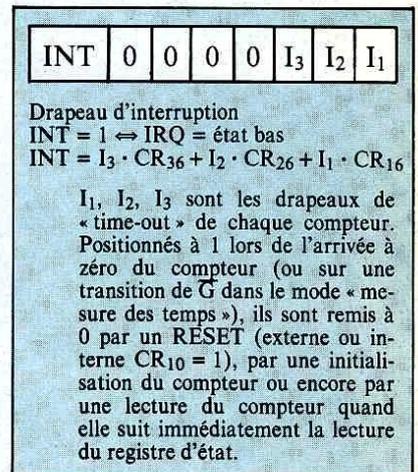


Fig. 2. - Le registre « d'état » (status register). Ce registre ne peut qu'être lu par l'unité centrale. Il est utilisé pour déterminer les fins de comptage (« Time Out ») et les interruptions.

Structure et brochage du 6840

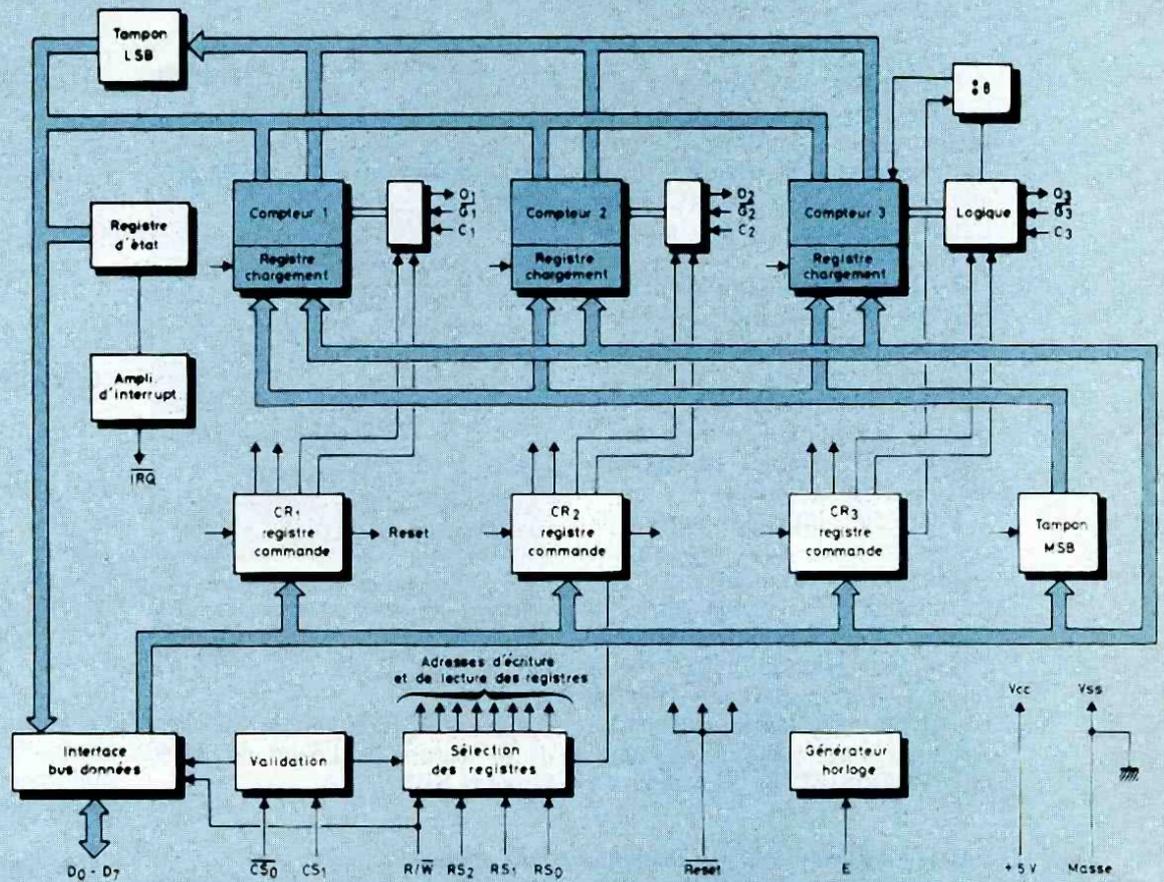


Fig. A : Synoptique du 6840 : un module « timer » programmable contenant trois compteurs commandés par l'unité centrale.

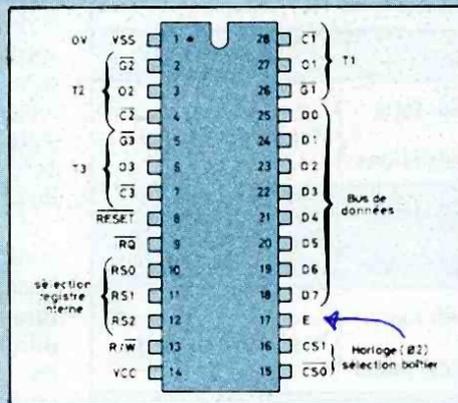


Fig. B : Brochage du 6840 :

- **Alimentation :**
 - VSS : 0 V
 - VCC : + 5 V (consommation moyenne 350 mW)
- **Signaux vers l'unité centrale :**
 - D0-D7 : Bus de données bidirectionnel (trois états).
 - R/W : signal de lecture-écriture (1 = lecture, 0 = écriture).
 - E : signal d'activation (Phase $\Phi 2$ de l'horloge).
 - IRQ : signal de demande d'interruption.
 - CS1, CS0 : sélection du boîtier (PTM adressé pour CS1 CS0 = 10).
 - RS0, RS1, RS2 : sélection des registres internes. Le timer occupe donc 8 cases mémoires.
 - RESET : Un niveau bas sur cette entrée initialise le circuit.
- **Signaux vers la périphérie :**
 - Chaque timer interne X (X = 1, 2 ou 3) est en liaison avec la périphérie grâce à 3 broches :
 - GX : signal de déclenchement (gate) du compteur X ;
 - OX : signal de sortie (output) ;
 - CX : horloge (clock) du compteur X

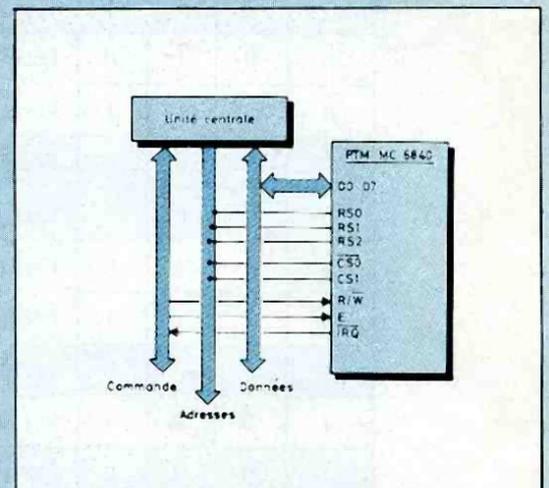


Fig. C : Le circuit s'intègre de façon classique dans un système 6800.

Les échanges de données se font à travers six registres de 16 bits (deux pour chaque timer). L'unité centrale accède en écriture au registre de chargement (Counter Latch) et peut aussi lire la sortie du compteur.

Le bus de donnée n'ayant que huit bits, une opération de lecture ou d'écriture de données doit donc s'effectuer en deux étapes, ce qui pourrait provoquer des erreurs comme le montre l'exemple de la **figure 3** si certaines **précautions** n'avaient pas été prises.

Pour éviter qu'une impulsion de comptage entre deux lectures ne vienne fausser la valeur lue, il suffit de **geler** la valeur de sortie au moment de la première lecture. Un **registre tampon**, placé à cette fin à la sortie du compteur sur l'octet de poids faible, est chargé au moment de la lecture de l'octet de poids fort (MSB). Sa valeur est

Sortie du compteur :

00FF

● 1^{er} temps : lecture de l'octet de poids fort.
Octet lu : 00

0100

● 2^e temps : impulsion de comptage éventuelle, puis lecture de l'octet de poids faible.
Octet lu : 00

Résultat lu :

0000 FAUX

Une bonne lecture aurait dû donner 00FF ou encore 0100.

Fig. 3. - Exemple de lecture erronée d'un compteur, qui aurait pu se produire si le constructeur n'avait pas prévu un registre « tampon » en sortie de l'octet « poids faibles » du compteur.

ensuite transférée sur le bus lors de la lecture de l'octet de poids faible (LSB).

Un **second registre intermédiaire** est de même utilisé en écriture pour éviter qu'une initialisation du compteur, au moment de l'écriture du registre de chargement, ne provoque une erreur.

Accès aux registres internes

La sélection des registres internes s'effectue d'abord par l'utilisation des trois lignes RS (Register Select).

Ceci donne huit possibilités d'adressage (2³). Cependant le nombre de registres accessibles étant supérieur, il est nécessaire d'utiliser le signal R/W (1 si lecture, 0 si écriture) ainsi qu'un bit de sélection (CR₂₀) afin d'obtenir un nombre de combinaisons suffisant pour adresser individuellement chacun des registres internes*.

Le **tableau 1** résume les différentes possibilités d'adressage en fonction des lignes R/W, RS₂, RS₁, RS₀ et du bit « 0 » du registre de contrôle CR₂.

La présence de tels registres intermédiaires impose que l'écriture dans un registre de 16 bits ou la lecture des sorties d'un compteur s'effectue **toujours** dans cet **ordre** : octet de poids fort suivi de l'octet de poids faible.

L'utilisation du signal R/W en complément d'adressage interdit l'usage d'instructions qui opèrent directement en mémoire pour modifier les valeurs contenues dans les registres (DEC, COM, ROL, etc.).

En effet, à une adresse donnée, l'utilisateur n'accède pas au même registre selon qu'il opère une lecture ou une écriture.

Notons que l'accès à CR₁, CR₃ se faisant à la même adresse (selon la valeur du bit CR₂₀), une écriture dans ces registres doit être en général précédée par une écriture dans CR₂. Cependant après l'application d'un signal Reset, CR₂ est mis à 0 ce qui conduit à initialiser le timer dans l'ordre suivant : CR₃, CR₂, CR₁*.

Tableau 1. - Tableau d'adressage des différents registres internes.

R/W	RS ₂	RS ₁	RS ₀	Opération
0	0	0	0	Ecriture de $\left\{ \begin{array}{l} \text{CR}_3 \text{ si le bit « 0 » de CR}_2 = 0 \\ \text{CR}_1 \text{ si le bit « 0 » de CR}_2 = 1 \end{array} \right.$
0	0	0	1	Ecriture du CR ₂
0	0	1	0	Ecriture des poids forts
0	0	1	1	Ecriture des poids faibles
				} Registre « Timer 1 »
0	1	0	0	Ecriture des poids forts
0	1	0	1	Ecriture des poids faibles
				} Registre « Timer 2 »
0	1	1	0	Ecriture des poids forts
0	1	1	1	Ecriture des poids faibles
				} Registre « Timer 3 »
1	0	0	0	Lecture du registre d'état
1	0	0	1	Pas d'opération
1	0	1	0	Lecture des poids forts
1	0	1	1	Lecture des poids faibles
				} Sortie du compteur 1
1	1	0	0	Lecture des poids forts
1	1	0	1	Lecture des poids faibles
				} Sortie du compteur 2
1	1	1	0	Lecture des poids forts
1	1	1	1	Lecture des poids faibles
				} Sortie du compteur 3

Les lecteurs familiers avec le PIA 6821 et l'ACIA 6850 retrouveront là les techniques d'adressage usuelles.

La mise à 1 du bit CR₁₀ provoque un RESET interne qui est actif jusqu'au retour à 0 de ce bit.

Toutes les durées, périodes ou intervalles de temps engendrés par ce circuit, sont définis par comptage d'un nombre entier de périodes d'horloge.

CR _{X5}	Initialisation du compteur	Autorisation de comptage SET A	Arrêt du comptage RAZ A	Positionnement du drapeau I
0	$\overline{G} \downarrow \cdot \overline{T} \cdot \overline{A} + \overline{G} \downarrow \cdot \overline{T} \cdot T + R$	$\overline{G} \downarrow \cdot \overline{W} \cdot \overline{R} \cdot \overline{T}$	W + R + I	$\overline{G} \downarrow$ avant T.O.
1	$\overline{G} \downarrow \cdot \overline{T} + R$	$\overline{G} \downarrow \cdot \overline{W} \cdot \overline{R} \cdot \overline{T}$	W + R + I	T.O. avant $\overline{G} \downarrow$

Tableau 2. - Ce tableau résume le fonctionnement du timer en mode « périodemètre », lorsque CR_{X3} = 1, CR_{X4} = 0.

- A = autorisation de décomptage mis à 1 si SET A = 1 à 0 si RAZ A = 1
- T = mémorisation du T.O. lorsque CR_{X5} = 0
 - mis à 1 si T.O. avant $\overline{G} \downarrow$
 - mis à 0 par initialisation du compteur
- T.O. = condition de « time-out » (fin de comptage), compteur à 0
- $\overline{G} \downarrow$ = front descendant pris en compte sur \overline{G}
- R = condition de reset ($\overline{Reset} = 0$ ou CR₁₀ = 1)
- I = drapeau d'interruption (voir registre d'état)
- W = écriture dans le registre de chargement

comparaison entre le temps séparant deux transitions négatives sur \overline{G} et le temps de décomptage du timer depuis l'initialisation (chargement) jusqu'au zéro (« Time out »).

Lorsque CR_{X5} = 1, l'initialisation se produit soit sur un « Reset » soit sur le premier front descendant de \overline{G} suivant la mise à zéro du « drapeau » d'interruption associé au timer. Une bascule interne est alors positionnée autorisant le décomptage jusqu'au premier des deux événements « time out » ou une nouvelle transition négative de \overline{G} . Dans le premier de ces deux cas, une interruption est demandée (I_X est mis à 1) et le

encadré 2

Mise en œuvre d'un « timer »

L'exemple d'utilisation que nous envisageons ici consiste à produire une série de « bips sonores » constitués par un signal audible de fréquence f_B « modulé » par un signal de fréquence f_M (fig. D).

Pour pouvoir engendrer un tel signal, nous utilisons le circuit MC6840 dans la configuration représentée figure E.

Le timer #1 délivre une fréquence de 440 Hz (f_B) pendant les passages à zéro du signal en sortie du timer #3.

Le timer #2 génère une fréquence de référence f_R à l'entrée \overline{C}_3 du timer #3, sa période T_R est de 10 ms et le signal est carré.

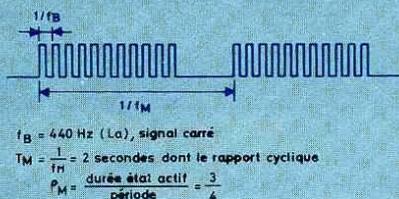


Fig. D. - Le signal que nous désirons produire...

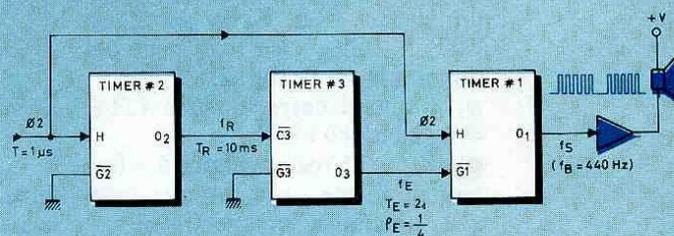


Fig. E. - Configuration de branchement des trois timers internes au boîtier 6840, pour notre application.

Programmation du timer #1

Pour déterminer les paramètres de programmation du timer #1, utilisons l'organigramme de la figure F.

Le signal est symétrique, d'où la valeur de N :

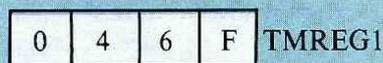
$$N = \frac{1}{2 \times 440 \times \frac{1}{10^6}} - 1$$

$$N \approx 1135 \text{ ou } (046F)_{16}$$

Le registre de commande CR₁ doit avoir la configuration suivante :



De même, pour le registre de chargement :



Programmation du timer #2

D'après l'organigramme de la figure F, nous avons :

$$N = \frac{1}{2 \times \frac{1}{10 \times 10^{-3}} \times 10^{-6}} - 1$$

$$= (4999)_{10}$$

$$N = (1387)_{16}$$

De même que pour le timer #1, nous avons :

compteur est stoppé par une remise à zéro de la bascule interne. Dans le deuxième cas une réinitialisation se produit et le cycle de mesure reprend.

Lorsque $CR_{X5} = 0$ le drapeau d'interruption ne sera positionné que si une transition négative de \bar{G} est détectée avant le retour à zéro du compteur. Si c'est le « time out » qui se produit le premier, le compteur est alors rechargé et recommence le décomptage. Cet événement est mémorisé de façon à ce que le prochain front descendant de \bar{G} provoque une réinitialisation du cycle au lieu d'une demande d'interruption.

Le fonctionnement dans ce

mode « mesure du temps séparant deux transitions descendantes de \bar{G} » est résumé tableau 2.

● Mesure de durée ou de largeur d'impulsion : $CR_{X4} = 1$

Dans ce mode on s'intéresse au temps séparant un front descendant d'un front montant sur \bar{G} . Le fonctionnement est donc assez voisin du mode précédent. Un cycle de décomptage étant initialisé par la transition descendante de \bar{G} , c'est maintenant la transition montante qui est prise en compte pour la comparaison avec le « time out ».

Lorsque $CR_{X5} = 0$ le drapeau d'interruption est positionné à 1 si l'impulsion négative sur \bar{G} est de

durée inférieure à la période de décomptage. Lors du front montant le comptage est stoppé ce qui permet ensuite de connaître la largeur de l'impulsion par une simple lecture du compteur.

Lorsque $CR_{X5} = 1$ le drapeau d'interruption est positionné sur le « time out » si cet événement précède le front montant de \bar{G} .

Dans le but de concrétiser ces notions théoriques toujours un peu délicates à mettre en œuvre, nous développons, encadré 2, un exemple simple d'utilisation simultanée des trois compteurs du timer. ■

J.-M. NOZERAN *
S. PHAN **

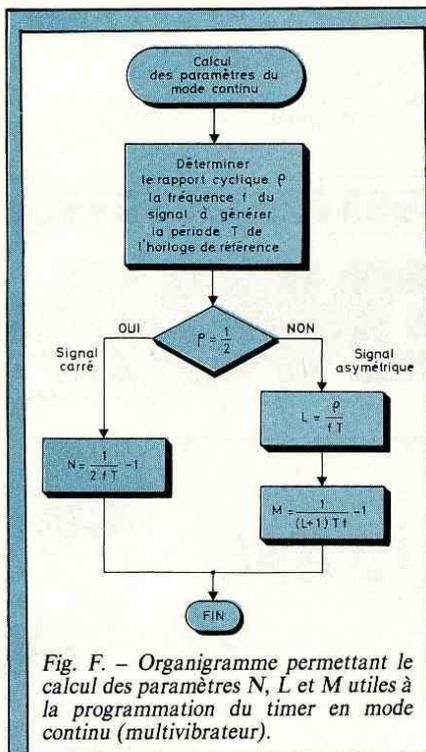


Fig. F. - Organigramme permettant le calcul des paramètres N, L et M utiles à la programmation du timer en mode continu (multivibrateur).

7	6	5	4	3	2	1	0
1	0	0	0	0	0	1	0

CR2

1	3	8	7
---	---	---	---

TMREG2

Programmation du timer # 3

Le signal de fréquence f_E engendré par le timer # 3 est actif au niveau zéro. L'entrée

\bar{G}_1 du timer # 1 doit donc recevoir le complément logique du signal de modulation de fréquence f_M .

C'est un signal asymétrique de même fréquence, mais de rapport cyclique $\rho_E = \frac{1}{4}$.

D'après l'organigramme de la figure F, nous déterminons L et M :

$$L = \frac{\rho_E}{f_E T_R} = \frac{\frac{1}{4}}{\frac{1}{2} \times 10 \times 10^{-3}}$$

$$= (50)_{10}$$

ou $L = (32)_{16}$

et

$$M = \frac{1}{(L+1) f_E T_R} - 1$$

$$= \frac{1}{(50+1) \times \frac{10 \times 10^{-3}}{2}} - 1$$

$$M = \frac{200}{51} - 1 \sim 3$$

d'où la programmation des registres :

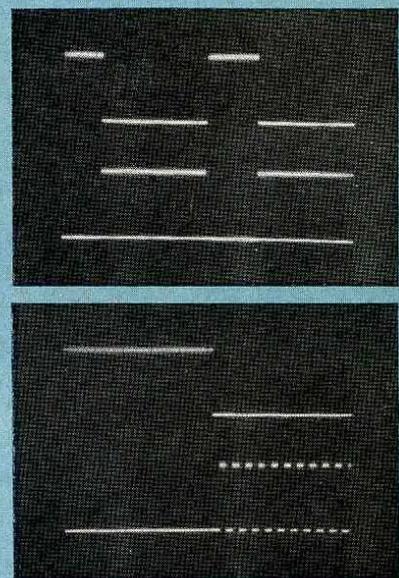
1	0	0	0	0	1	0	0
---	---	---	---	---	---	---	---

CR3

0	3	3	2
---	---	---	---

TMREG3

Le listing 1 détaille les étapes de la programmation de l'application proposée.



Photos 1. et 2. - Détails des signaux en G_1 (enveloppe) et en O_1 (bips). Dans les deux cas, nous remarquons que le signal « enveloppe » (en haut) est bien le complémentaire du signal de modulation (en bas).

Le timer permet de « soulager » le microprocesseur dans la génération des « bips ». En effet, dès l'instruction STAA TIMCR1, le timer fonctionne de façon autonome.

Les photos 1 et 2 montrent les signaux prélevés sur \bar{G}_1 et O_1 . Remarquez que le signal enveloppe est bien le complémentaire du signal de modulation. ■

* Jean-Marc Nozeran est ingénieur « Sup-Aéro » et pratique l'enseignement des microprocesseurs dans plusieurs écoles d'ingénieurs réputées.
** Phan Son est ingénieur « Supélec » et ancien élève de l'I.A.E.

Listing 1. - Un programme destiné à générer une série de bips sonores.

```

PAGE 001 TIMER ,SA:0 TIMER
00001          NAM    TIMER
00002A 0000          ORG    $0000
00003          *****
00004          *
00005          *          EVALUATION DU TIMER PROGRAMMABLE MC 6840
00006          *
00007          *****
00008          *
00009          * ADRESSE DES REGISTRES INTERNES DU TIMER
00010          *
00011          0090 A TIMCR1 EQU $90      REGISTRE DE COMMANDE DU TIMER #1
00012          0091 A TIMCR2 EQU $91      REGISTRE DE COMMANDE DU TIMER #2
00013          0090 A TIMCR3 EQU $90      REGISTRE DE COMMANDE DU TIMER #3
00014          0092 A TMREG1 EQU $92      REGISTRE DE CHARGEMENT DU TIMER #1
00015          0094 A TMREG2 EQU $94      REGISTRE DE CHARGEMENT DU TIMER #2
00016          0096 A TMREG3 EQU $96      REGISTRE DE CHARGEMENT DU TIMER #3
00017          *
00018          * PARAMETRES
00019          *
00020          046F A LA440 EQU 1135      DEMI-PERIODE DU LA 440 HZ = 1135 MICROSECONDES
00021          1387 A H10MS EQU 4999      DEMI-PERIODE DE L'HORLOGE INTERMEDIAIRE DE 10 MILLISEC.
00022          0032 A ENVLP1 EQU 50       DUREE DU NIVEAU HAUT DE L'ENVELOPPE COEF, LSB
00023          0003 A ENVLP1 EQU 3        COEFFICIENT MSE
00024          0082 A CONT16 EQU 1000010B MODE CONTINU 16 BITS , HORLOGE INTERNE
00025          0084 A CONT8 EQU 10000100B MODE CONTINU 2*8 BITS , HORLOGE EXTERNE
00026          *
00027          * PROGRAMMATION DU CIRCUIT
00028          *
00029A 1000          ORG    $1000
00030A 1000 7F 0091 A DEBUT CLR    TIMCR2  ACCES A TIMCR3
00031A 1003 86 84 f) LDAA    #CONT8  PROGRAMMATION
00032A 1005 97 90 A      STAA    TIMCR3  DU REGISTRE DE COMMANDE TIMER #3
00033A 1007 86 82 A      LDAA    #CONT16 PROGRAMMATION DE TIMCR2
00034A 1009 8A 01 A      ORAA    #1     ACCES A TIMCR1
00035A 100B 97 91 A      STAA    TIMCR2
00036A 100D CE 046F A     LDX    #LA440  GENERATION DU LA 440 HZ PAR TIMER #1
00037A 1010 DF 92 A      STX    TMREG1
00038A 1012 CE 1387 A     LDX    #H10MS  GENERATION D'UNE HORLOGE DE 10 MILLISECONDES PAR TIMER #2
00039A 1015 DF 94 A      STX    TMREG2
00040A 1017 86 03 A      LDAA    #ENVLP1  GENERATION DE L'ENVELOPPE PAR TIMER #3
00041A 1019 C6 32 A      LDAB    #ENVLP1
00042A 101B 97 96 A      STAA    TMREG3
00043A 101D D7 97 A      STAB    TMREG3+1
00044A 101F 86 82 A      LDAA    #CONT16
00045A 1021 97 90 A      STAA    TIMCR1  DEMARRAGE DU COMPTAGE DES L'EXECUTION DE CETTE INSTRUCTION
00046A 1023 20 FE 1023 SUITE BRA *     SUITE DU PROGRAMME
00047          END
TOTAL ERRORS 00000---00000

```