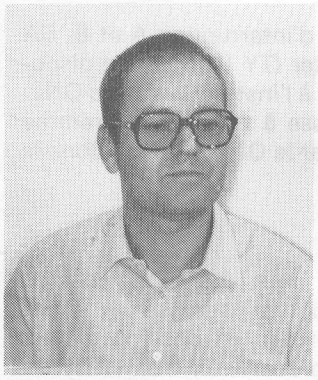
# SYSTEME



## Microordinateur basic à tout faire

# MICRO-VON

**Michel VONLANTHEN HB9AFO**

*Résumé :*

*Le système MICRO-VON est un ensemble microordinateur d'application complet – électronique et programme – destiné à être incorporé dans un ensemble. Une fois le programme mis au point, la partie « système de développement » est débranchée et l'ordinateur fonctionne de façon indépendante. Il est programmable en BASIC, langage de haut niveau facilement assimilable par le débutant, et en langage machine. La description qui suit est destinée à permettre à chacun de composer ses propres applications. Pour en illustrer le processus, un ensemble RTTY performant (radiotélétype) pour radioamateur sera décrit. Les circuits-imprimés et les composants sont distribués par HAMCO (Suisse) et facilement obtenables.*

*La première partie de cette série a décrit les principes généraux du système RTTY qui suivra.*

## LA CARTE « MICROPROCESSEUR » VON257

C'est le cœur du système MICRO-VON. Cette carte contient le microprocesseur 8073, son quartz-horloge de 4MHz, 2k bytes de RAM statique, 2k bytes d'EPROM (2716), le circuit de sélection des RAM (LS138/IC4), le circuit de sélection des EPROM (LS138/IC5) et une quadruple porte NON-ET servant à inverser certains signaux. Un connecteur direct de 2×22 pins sert, en principe, à enficher la carte de programmation lorsqu'on veut programmer l'EPROM IC2. Certaines pins de ce connecteur sont en parallèle avec un connecteur DIN indirect 31 pins qui est normalement équipé de pins à wrapper. Dans le cas où une application le nécessiterait, il est possible de scier la partie connecteur 2×22 et de monter un 31 pôles DIN à la place.

La carte « microprocesseur » n'a pas été conçue comme un système figé mais comme une base pour une infinité d'applications : système RTTY, programmation d'EPROM, commande d'antenne, contrôle de processus, etc... Pour en élargir au maximum les possibilités, tous les signaux de données, d'adresses et de contrôle sont accessibles, soit via le connecteur 2×22, soit via des pins à wrapper (wrap veut dire « enrouler » en anglais). Il s'agit d'un câblage sans soudures où le fil est enroulé autour d'un piton de forme carrée et arêtes tranchantes à l'aide d'un outil spécial. C'est très pratique pour les prototypes car il suffit de dérouler le fil pour modifier le câblage.

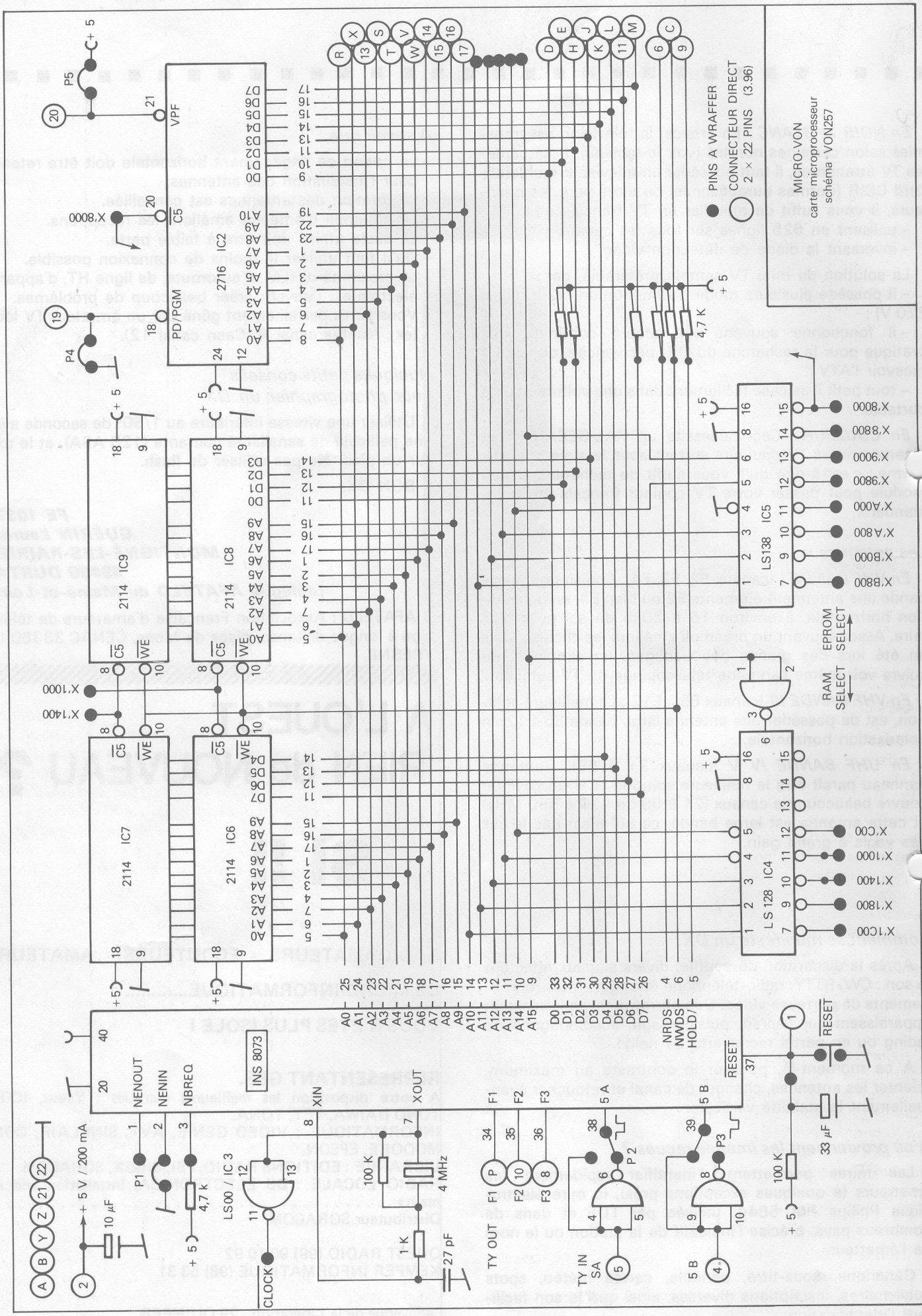
On a donc accès à tous les signaux du microprocesseur. Certains ponts ont aussi été prévus pour tirer parti au maximum des éléments inutilisés sur la carte. Entrons maintenant dans les détails :

*Le schéma :*

Le microprocesseur est un INS8073 de National Semiconductor Company, circuit-intégré à 40 pattes. Il ne nécessite que du 5V et tous les circuits auxiliaires (clock, latch, etc.) y sont intégrés. La particularité de ce microprocesseur est qu'une ROM de 2,5k bytes et une RAM de 64 bytes sont incorporés dans le chip. La ROM (Read Only Memory = mémoire qui ne peut être que lue) est programmée d'usine avec un TINY BASIC, language de programmation évolué qui facilite grandement la programmation, la rendant accessible même au non-initié. La RAM (Random Access Memory = mémoire à accès aléatoire) est une mémoire qui peut être lue, comme la ROM mais aussi écrite. Par contre, elle perd son contenu lorsqu'on coupe le courant. Les lignes d'adresses AO à A15 sont au nombre de 16 ce qui permet d'adresser un maximum de 64k. Il y a 8 lignes de données (datas) D0 à D7 bidirectionnelles. Ces 24 lignes sont du type « trois états ». 14 pins du chip sont réservées aux signaux de contrôle et décrites ci-après :

**Reset**

Un 0 sur cette patte peut être donné soit de l'extérieur de la carte, soit par le bouton-poussoir incorporé. A l'enclenchement également, le condensateur de 33 microfarads garde cette pin à 0 pendant un court instant ce qui suffit à provoquer la fonction de remise a zéro (un trigger de Schmitt est compris dans le circuit-intégré). Ce signal met le processeur en condition d'initialisation : remise à zéro du programm counter, du stack counter, status bits, etc. Ensuite ce dernier fait une recherche non-destructive de la mémoire. Si une EPROM contenant un programme basic valide (il doit débuter par un numéro de ligne suivi d'un CLEAR) est sur le socle d'IC2, le microprocesseur


MICRO-VON
carte microprocesseur
schéma VON257
PINS A WRAFFER
CONNECTEUR DIRECT
2 × 22 PINS (3.96)
INS 8073
2716 IC2
2114 IC6
2114 IC7
2114 IC8
2114 IC9
LS138 IC5
LS 128 IC4
LS00 / IC 3
EPROM SELECT
RAM SELECT
RESET
CLOCK
TY OUT
TY IN
SA
5 B
4 MHz
+ 5 V / 200 mA

commence directement à exécuter ce programme ce qui constitue donc une mise en marche automatique. Dans le cas où il n'y a pas d'EPROM, on passe directement en mode de programmation et le basic attend une commande du clavier via la ligne d'entrée série SA.

### NENOUT et NENIN

Signaux de contrôle de l'accès direct à la mémoire (DMA), non-utilisé pour le moment.

### NBREQ

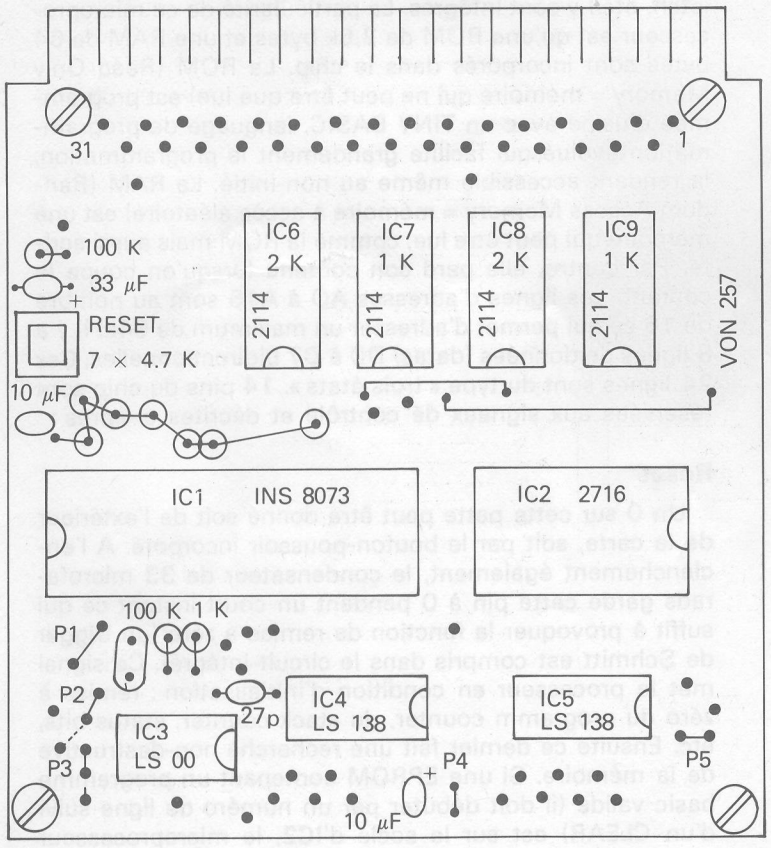
Signal entrée/sortie de contrôle de bus, non-utilisé pour le moment.

### XIN, XOUT

Entrée/sortie de l'horloge interne. Le circuit-imprimé permet de câbler soit une combinaison résistance/condensateur, solution économique, soit un quartz de 4MHz.

### F1, F2 et F3

F = flag (drapeau). Ce sont les bits 1, 2 et 3 du mot d'état (status). Ces signaux peuvent être commandés par software (software = programmation) et sont des sorties série directement utilisables. F1 est la sortie série allant vers le display (TY OUT). F2 est prévue pour commander un lecteur de bandes perforées dans le cas où on utilise un terminal mécanique du genre du TELETYPE ASR33 par exemple à la place du clavier/display. F3 est disponible pour l'utilisateur.




### SA, SB

Sense interrupt A, B = entrée d'interruption A et B. SA est l'entrée série venant du clavier (TY IN). SB est disponible pour le programmeur grâce à l'instruction basic ON2, ... Dès que le niveau de SB passe à zéro, le programme basic saute à la ligne spécifiée par le ON2, ... C'est donc le basic qui gère les interruptions.

### NRDS

Lorsque le microprocesseur veut lire une position mémoire ou périphérique, il en donne l'adresse sur le bus (A0 à A15), met le bus de données en position lecture (données entrant dans le microprocesseur) et met NRDS zéro un court instant ce qui transfère l'information via le bus de données dans le microprocesseur.

### NWDS

Même chose que NRDS mais pour écrire cette fois-ci, donc les données sortent du microprocesseur et vont soit vers les mémoires soit vers un périphérique.

### HOLD

Un zéro sur cette ligne fige le microprocesseur en position d'attente, sans rien effacer. Utilisé par le programmateur d'EPROM.


NOUVEAU?!

SALON RADIOAMATEUR

ORGANISE PAR

l'ONDE MARITIME

le 20 fevrier 1983

Presqu'ile de FOURAS 17

(Renseignements dans le numéro de janvier)

## SÉLECTION DE LA VITESSE DU TERMINAL

Certaines applications, comme par exemple la commande de processus, ne nécessitent pas nécessairement de terminal (terminal = ensemble clavier/display). C'est la raison pour laquelle il n'en a pas été incorporé sur la carte microprocesseur. Pour en connecter un, les lignes SA et F1 sont utilisées en transmission asynchrone ASCII. Le tiny basic, lors d'une remise à zéro, lit le contenu de l'adresse X'FDOO qui donne la vitesse en BAUDS selon le tableau ci-dessous.

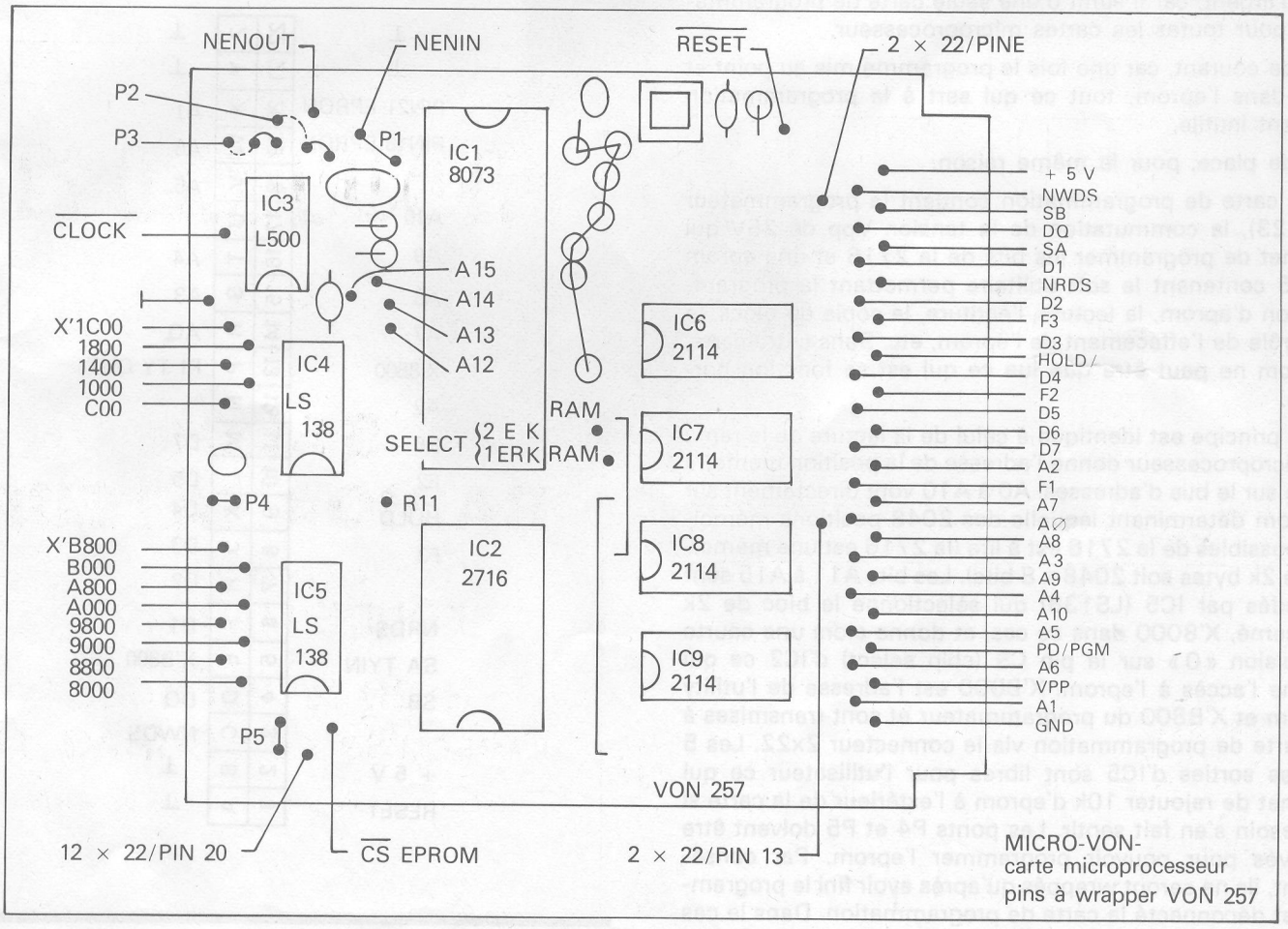
| Vitesse (bauds) | D7 | D2 | D1 |
|---|---|---|---|
| 110 | 1 | 1 | 1 |
| 300 | 1 | 1 | 0 |
| 1200 | 1 | 0 | 1 |
| 4800 | 1 | 0 | 0 |

Dans notre cas, le développement de programmes s'effectue à l'aide d'un terminal mécanique 110 Bauds TELETYPE, sorte de téléimprimeur ASCII et boucle de courant 20mA. Pour simplifier la carte de base (et aussi pour économiser des mA, nécessité toujours présente à notre esprit) la sélection des vitesses a été réduite à sa plus simple expression : 3 résistances sur D1, D2 et D7 ce qui a pour effet, à l'enclenchement, de tirer D1, D2 et D7 au 5V et de sélectionner la vitesse de 110 Bauds. Pour les autres vitesses, on pourra utiliser la carte de sélection des périphériques VON263 qui décode complètement l'adresse X'FDOO entre autres fonctions.

### La RAM

Le circuit-intégré choisi est la mémoire statique la plus répandue : la 2114. Chaque chip contient 1024×4 bits. Il en faut donc deux pour faire 1k bytes (1024×8 bits, 1 byte étant égal à 8 bits). Les IC6 et 8 forment le premier k, de X'1000 à X'13FF tandis que IC7 et 9 vont de X'1400 à X'17FF. Rappelons en passant que le X' devant un nombre signifie que celui-ci est un nombre binaire exprimé en héxadécimal, de 0 à F.

Lorsque le microprocesseur veut lire une donnée en mémoire, il en donne l'adresse sur le bus d'adresses et positionne le bus de données (datas bus) en lecture, c'est-à-dire que les données vont de la mémoire au microprocesseur. Le circuit IC4 (LS138) décode les bits A10 à A15 du bus d'adresses et sélectionne le bloc de 1k ram concerné. Sur la carte elle-même, seules les adresses X'1000 et X'1400 sont utilisées, les autres restant libres pour d'autres utilisations. IC4 donne donc un zéro sur la pin CS (= chip select) du bloc de ram sélectionné. Les adresses A0 à A9 vont en parallèle sur les 2 blocs de 1k et précisent laquelle des 1024 positions possibles sur les 1k on veut lire. Ensuite, le microprocesseur envoie le signal



**********

NRDS (= read device strobe), le N veut dire « barre », donc que ce signal est actif lorsqu'il descend au zéro) ce qui strobe le contenu de la location mémoire désirée dans le registre sélectionné parce que la pin WE/ de la ram est à 1. Voici pour la théorie mais en pratique, le NRDS n'est pas relié aux ram car c'est l'adressage X'1000 ou X'1400 qui strobe via le CS/.

Le processus est exactement le même pour l'écriture sauf que la donnée est positionnée en sortie, sens micro vers mémoire, et que le strobe est cette fois (théoriquement) NWDS (write device strobe) qui met à zéro la pin WE/ de la ram ce qui commute en mode « écriture ».

Les sorties X'1C00 et X'1800 d'IC4 sont disponibles pour adresser 2k de ram supplémentaires à l'extérieur de la carte. X'C00 ne peut pas ête utilisé pour la mémoire car le basic du 8073 n'accepte pas de ram en-dessous de X'1000.

## L'EPROM

(Erasable Programmable Read Only Mémory). C'est la mémoire non-volatile, c'est-à-dire que son contenu ne s'efface pas lorsqu'on coupe le courant et elle contient le programme écrit par l'utilisateur. Elle peut être programmée grâce à la carte de programmation VON267 qui vient s'enficher, le temps de la programmation, sur le connecteur 2×22. Cette solution a été choisie pour des raisons d'économie :

– d'argent, car il suffit d'une seule carte de programmation pour toutes les cartes microprocesseur,

– de courant, car une fois le programme mis au point et figé dans l'eprom, tout ce qui sert à la programmation devient inutile,

– de place, pour la même raison.

La carte de programmation contient le programmateur (LS123), la commutation de la tension Vpp de 25V qui permet de programmer les bits de la 2716 et une eprom 2716 contenant le soft utilitaire permettant la programmation d'eprom, la lecture, l'écriture, la copie de blocs, le contrôle de l'effacement de l'eprom, etc. Sans cette carte, l'eprom ne peut être que lue ce qui est sa fonction normale.

Le principe est identique à celui de la lecture de la ram : le microprocesseur donne l'adresse de la position mémoire à lire sur le bus d'adresses. A0 à A10 vont directement sur l'eprom déterminant laquelle des 2048 positions mémoires possibles de la 2716 est à lire (la 2716 est une mémoire de 2k bytes soit 2048 × 8 bits). Les bits A11 à A15 sont décodés par IC5 (LS138) qui sélectionne le bloc de 2k concerné, X'8000 dans ce cas, et donne alors une courte impulsion « 0 » sur la pin CS (chip select) d'IC2 ce qui donne l'accès à l'eprom. X'B000 est l'adresse de l'utility eprom et X'B800 du programmateur et sont transmises à la carte de programmation via le connecteur 2×22. Les 5 autres sorties d'IC5 sont libres pour l'utilisateur ce qui permet de rajouter 10k d'eprom à l'extérieur de la carte si le besoin s'en fait sentir. Les ponts P4 et P5 doivent être enlevés pour pouvoir programmer l'eprom. Par conséquent, ils ne seront wrappés qu'après avoir fini le programme et déconnecté la carte de programmation. Dans le cas d'un programme évolutif, il vaut mieux câbler un connecteur 2×22 femelle avec la pin 19 à la masse (=P4) et la pin 20 au +5V (=P5) qui sera enfiché chaque fois que la carte de programmation sera retirée.

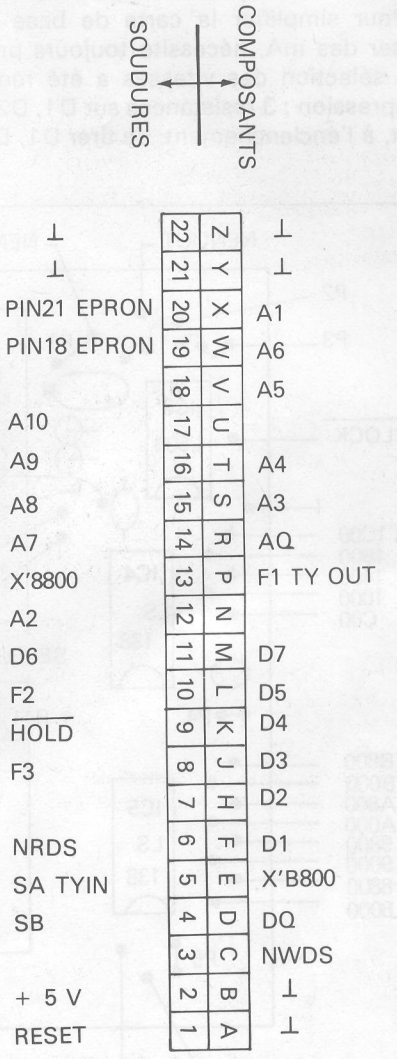
**Circuits annexes :**

a) La fréquence d'horloge de 4MHz est amplifiée par IC3 et se trouve disponible sur une pin à wrapper.

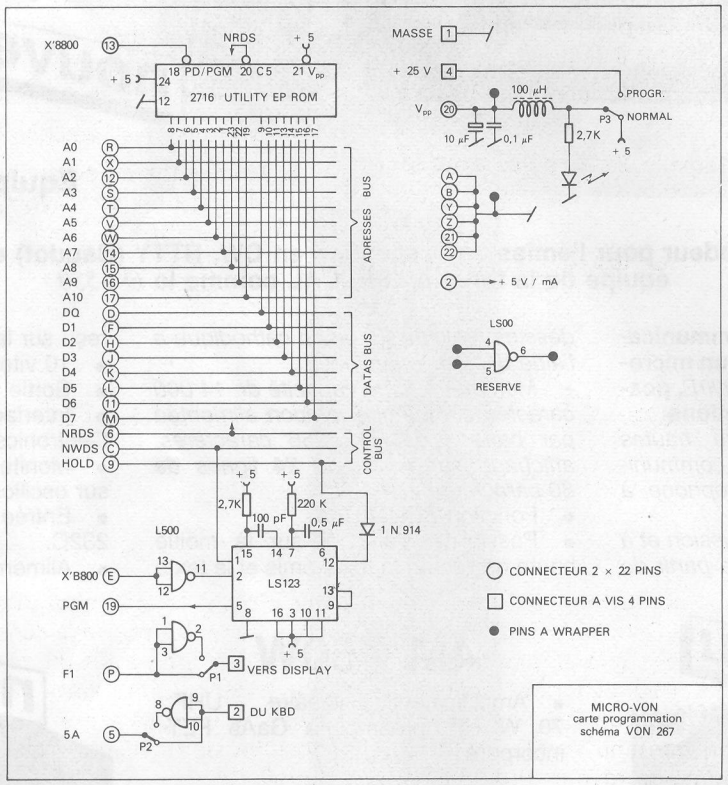
b) Le signal allant du clavier à SA peut être inversé par IC3. P2 est câblé en inverseur au départ.

c) Le signal arrivant à SB, libre pour l'utilisateur, est câblé inversé au départ (P3).

Dans les deux cas, on peut inverser la polarité de ces signaux à volonté en wrappant P2 et P3 correctement.



**********





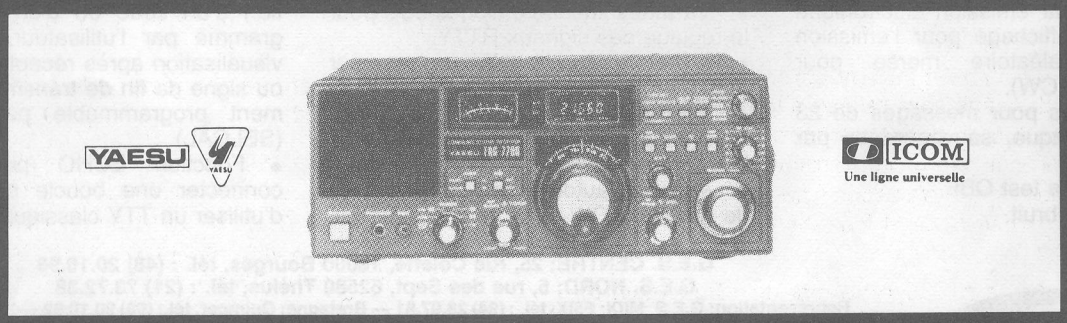
à **BORDEAUX**

SPECIALISTE EN COMMUNICATIONS

Dépannage Mise au point.Conseils.Vente

**RADIO-SHOP**

55 RUE DU TONDU 33000 BORDEAUX




**F6FPQ** a VOTRE SERVICE

**Mise en route**

La carte microprocesseur VON257 peut être utilisée tout d'abord seule avec un terminal pour se familiariser avec le basic résident. Pour ce faire, il faut :

1. Laisser tous les ponts comme sur le schéma et ne pas mettre d'eprom sur le socle.

2. Préparer un connecteur femelle 2x22 pôles avec le câblage suivant :

A masse du 5V et du terminal
2 +5V/200mA max (avec tous les IC, y compris l'eprom)
5 entrée clavier (SA)
P sortie vers display (F1)

3. Le clavier/display doit travailler en ASCII 110 Bauds, sans parité, 7 bits, c'est-à-dire le « standard Teletype ». Le basic n'accepte pas les minuscules.

4. Mettre sous tension.

5. Un> apparaît à gauche de l'écran ce qui indique que le basic est prêt à recevoir une commande. Faites PRINT « SALUT ». SALUT doit s'inscrire sur l'écran !... La programmation en basic peut commencer !...

## LA CARTE DE PROGRAMMATION

### VON 267

La carte « microprocesseur » connectée au terminal permet de programmer en basic. Seules les RAM sont utilisées dans ce cas, ce qui signifie que le programme est perdu dès qu'on coupe le courant. Pour éviter cet inconvénient, il suffit de transférer le programme basic dans l'eprom qui, elle, conserve son contenu indéfiniment, à moins qu'on l'expose à une source de rayons ultra-violets (par exemple une lampe bactéricide car une lampe à quartz pour bronzage n'est pas assez « violente »). Dans ce cas, le contenu de l'eprom est effacé et chacune de ses 2048 locations est remplie par X'FF ce qui permet ensuite d'y entrer d'autres programmes.

La carte de programmation contient :

– le circuit double monostable (LS123) qui donne la cadence nécessaire pour programmer 1 byte de la 2716 ;

– une eprom 2716 programmée avec le programme « UTILITY #5 », qui procure des facilités de programmation en langage machine et permet de remplir l'eprom qui se trouve sur la carte microprocesseur.

Cette carte est équipée d'un connecteur 2x22 qui vient s'enficher, le temps pendant lequel on en a besoin, sur la carte microprocesseur.

*Le schéma :*

On distingue tout d'abord l'utility eprom dont les lignes d'adresses et de données sont reliées à leur bus respectif. L'adresse de sélection de cette mémoire est X'B000 (pin 18). Elle n'est bien-sûr pas programmable et ne peut être que lue. Pour pouvoir programmer l'eprom qui se trouve sur la carte microprocesseur, il faut changer deux de ses niveaux :

1. Il faut alimenter la pin 21 (Vpp) par du + 25V (environ 30-40mA) au lieu de + 5V. Ceci se fait manuellement à l'aide du commutateur P3 qui se trouve sur la carte de programmation. Une diode électro-luminescente signale la présence du 25V en brillant avec plus d'intensité. Lors de la programmation, une fois les adresses de départ et de fin du bloc à transférer entrées, le soft signale « mettre le SW sur prog. » ce qui constitue une sécurité physique empêchant une écriture accidentelle. Cette protection n'aurait pas existé si on avait fait cette commutation par programme, ce qui eut été aussi simple mais moins sûr. Il ne faut évidemment pas oublier de remettre ce commutateur en position normale une fois la programmation de l'eprom terminée !

2. Il faut ensuite alimenter la pin 18 (PD/PGM) selon la procédure suivante :

– mettre l'adresse de la location à programmer sur le bus d'adresses

– mettre la donnée à programmer sur le bus de données

– donner une impulsion de 50ms sur la pin 18 et, pendant ce même temps, stopper le processeur grâce à l'entrée HOLD/. C'est le LS123, dont l'adresse de sélection est X'B800, qui donne cette impulsion de 50ms, retardée de 200 microsecondes pour que les signaux sur le bus soient stabilisés. La diode 1N914 constitue un « OU cablé » sur le hold/ ce qui permet aussi d'utiliser cette entrée depuis l'extérieur.

Deux portes NAND, câblées en inverseurs, autorisent n'importe-quelle polarité des signaux du terminal (SA et F1). Il reste une porte en réserve, équipée de pins à wrapper.

Le terminal vient se connecter à la carte sur un connecteur à vis à 6 pôles :

*La masse, le + 25V, l'entrée clavier, la sortie display* et *du + 5V* pour alimenter éventuellement un circuit extérieur comme, par exemple, le translateur « boucle de courant 20mA/TTL nécessaire lorsqu'on veut travailler avec un TELETYPE mécanique en guise de terminal.

Le circuit-imprimé est du même type que celui de la carte microprocesseur c'est-à-dire à double face et à trous métallisés. Il est recouvert de laque verte anti-soudure qui évite les ponts indésirables. Il mesure 10x10cm. C'est, en principe, la dernière carte de ce type utilisée sur le système MICRO-VON- car elle est difficilement réalisable avec des

moyens amateurs et c'est la raison pour laquelle nous l'avons fait réaliser par une maison spécialisée. Les prochaines cartes seront des simples-faces facilement réalisables par l'amateur. Les dimensions standardisées sont de 5×10cm.

**L'utility eprom**

Elle est livrée déjà programmée avec chaque carte de programmation et utilise aussi bien le basic que le langage machine. Ses fonctions :

1. Lire une zone mémoire ram ou eprom. Chaque ligne sur le display contient 16 bytes consécutifs en hexadécimal et, tout à droite de l'écran, leurs équivalents ASCII s'ils en ont un.

2. Écrire en hexadécimal à partir d'une adresse de départ. On termine en tapant CTRL-C.

3. Copier un bloc de mémoire dont on donne l'adresse de départ, de fin et la nouvelle adresse.

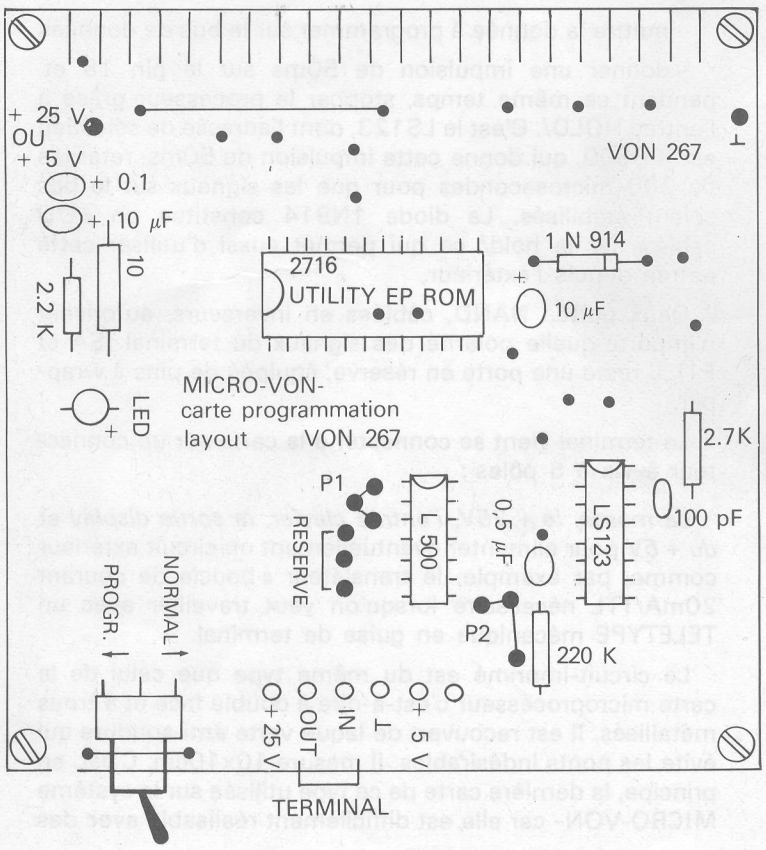
4. Contrôler l'effacement d'une eprom.

5. Remplir une zone mémoire dont on donne les adresses de départ et de fin par une donnée spécifiée (de X'00 à X'FF).

6. Programmer l'eprom. On n'est pas obligé de la remplir entièrement d'un seul coup, on peut même, à la limite, la remplir byte après byte.

7. Transformer un nombre décimal donné au clavier en son équivalent hexadécimal. Le basic répond toujours en décimal d'où l'utilité de cette routine.

L'utilisation de cette eprom est décrite en détail dans le mode d'emploi livré avec chaque carte.




**Mise en route de la carte de programmation :**

D'une manière identique à la carte microprocesseur, la carte de programmation est accompagnée d'une notice de montage détaillée. Il ne devrait donc pas y avoir de problème. Il n'est cependant pas inutile de préciser ce qui suit :

Dans le domaine des microprocesseurs où règnent en maîtres les bus parallèles, la précipitation est mauvaise conseillère. Il est en effet très difficile, à moins de disposer des connaissances et instruments adéquats, de suivre les signaux sur la carte. Tout étant en parallèle, le moindre court-circuit fausse totalement le déroulement du programme. Le dépannage ne pourra s'effectuer qu'au moyen de routines écrites en langage machine, testant chaque fonction l'une après l'autre, ce qui demande une connaissance approfondie du fonctionnement du microprocesseur et de ses circuits. C'est la raison pour laquelle il vaudra mieux perdre une heure de plus à contrôler les soudures à la loupe (ponts de soudure, soudures froides etc.) plutôt que de câbler à la va-vite et de ne rien contrôler. *Dans ce domaine, perdre une heure avant la mise sous tension c'est gagner une semaine de dépannage !...*

Une fois tout bien contrôlé (attention au sens des IC !) et sans enficher la carte, mesurer avec un voltmètre la présence du +5 et du +25V sur la pin marquée « +25 ou +5V » en manœuvrant le commutateur « normal/progr ». Le display et le +25V doivent bien sûr être branchés sur le connecteur à vis auparavant !

Ensuite enficher la carte de programmation sur la carte microprocesseur en prenant évidemment la précaution de couper l'alimentation avant. C'est d'ailleurs une règle générale à ne pas oublier, la seule exception étant l'eprom, qu'il est permis d'enlever ou de mettre sur son socle sans couper l'alimentation.

A l'enclenchement, le prompt > vient sur l'écran. Il suffit alors de taper sur le clavier NEW #B000 pour que le message R, W, C, E, F, P, T ou CTRL-C?? apparaisse ce qui signifie que nous sommes sous contrôle du programme contenu dans l'utility eprom.

Ce message veut dire :

R = lecture (Read)
W = écriture (Write)
C = copie de bloc
E = contrôle de l'effacement de l'eprom
F = remplissage (Fill) d'un bloc mémoire
P = programmation de l'eprom
T = transformation décimal-hexadécimal
CTRL-C = retour au basic

Le programme « utility #5 » est partiellement écrit en basic ce qui ralentit certaines fonctions, telles, par exemple la programmation qui prend un certain temps pour être exécutée. Il faut déjà 50ms pour figer chaque position mémoire et ceci, additionné au temps d'exécution du basic, provoque un temps d'attente de plusieurs dizaines de secondes.

Pendant la phase de développement d'un programme, la carte de programmation restera en permanence connectée à la carte microprocesseur. Les ponts P4 et P5 de cette dernière ne doivent donc pas être immédiatement câblés

COTE COMPOSANTS

| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Z | Y | X | W | V | U | T | S | R | P | N | M | L | K | J | H | F | E | D | C | B | A |
| 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

VU DE L'EXTERIEUR DE LA CARTE (CABLAGE CONNECTEUR)

sous peine de ne pas pouvoir programmer l'eprom. Néanmoins, spécialement dans le cas du hobbyist qui est souvent un perfectionniste invétéré, un programme est rarement définitif ! Dans ce cas, plutôt que de wrapper les ponts à demeure, il est préférable de les laisser libres et enficher un connecteur 2×22 avec le câblage suivant :

pin 20 reliée à la pin 2 (Vpp ou +5V = pont P5)

pin 19 reliée à la pin 21 (PD/PGM à la masse = pont P4)

Si on veut modifier le programme, il suffira alors de retirer ce connecteur et de le remplacer par la carte de programmation. Cela évitera de wrapper et de déwrapper constamment P4 et P5, au risque d'endommager le LS123.

*A suivre :*

La description quelque peu rébarbative des deux cartes de base du système étant faite, nous pourrons maintenant entrer dans le vif du sujet avec la première application pratique du système MICRO-VON- : la RTTY.

***Michel VONLANTHEN HB9AFO***

*Note :*

Les intéressés peuvent obtenir les informations sur le matériel disponible en envoyant une enveloppe self-adressée + coupon-réponse international à : HAMCO, case postale, CH-1024 Ecublens (Suisse).


**POUR VOTRE PUBLICITE**

**16-40-66.55.71**

**RR REGENT RADIO**

**GROSSISTE ● IMPORTATEUR CB ● ACCESSOIRES VAN**

DISTRIBUTEUR : TAGRA HMP TURNER K 40 HY-GAIN
AVANTI ZETAGI CTE ASTON ZODIAC MIRANDA
RAMA DENSEI PORTENSEIGNE Quartz Composants Radio TV - CB

*LIVRAISON SUR PARIS ET EXPEDITION DANS TOUTE LA FRANCE*

**101-103, AV. DE LA RÉPUBLIQUE 93170 BAGNOLET**

**Tél. 364.10.98 - 364.68.39** *Parking derrière Station ELF*