

9/9

Programmes divers

9/9.1

Générateur de signaux morse

Survivant des temps héroïques des radiocommunications, le code Morse est aujourd'hui encore très utilisé aux côtés des techniques les plus modernes de transmission numérique.

Il s'agit en effet du moyen le plus simple permettant l'échange de messages relativement fiables dans les plus mauvaises conditions.

L'informatique peut remplacer l'opérateur pour les tâches d'émission et, dans une certaine mesure, de réception. Elle peut aussi l'aider à apprendre le morse, qui figure au programme des principaux examens de radio-amateur.

Le logiciel qui va être publié ici est capable de transformer un mot frappé au clavier, en une suite de tonalités représentant les « points » et les « traits » du code de la figure 1.

Ces signaux audio peuvent évidemment être directement émis par tout émetteur « phonie », mais sont surtout utiles à des fins pédagogiques.

Il ne suffit pas, en effet, d'apprendre par cœur le tableau de l'alphabet pour pouvoir prétendre « connaître » le morse. L'opérateur expérimenté est capable de reconnaître « au vol » non seulement chaque lettre ou signe, mais aussi des mots entiers (et pas seulement le fameux « SOS » !)

C'est seulement de cette façon qu'il est possible de ne pas perdre le fil d'une transmission rapide...

A	•—	N	—•	1	•— — — —
B	—•••	O	— — —	2	••— — —
C	—•—•	P	•— — •	3	•••— —
D	—••	Q	— — •—	4	••••—
E	•	R	•—•	5	•••••
F	••—•	S	•••	6	—••••
G	— — •	T	—	7	— — •••
H	••••	U	••—	8	— — — ••
I	••	V	•••—	9	— — — — •
J	•— — —	W	•— —	0	— — — — —
K	—•—	X	—••—		
L	•—••	Y	—•— —		
M	— —	Z	— — ••		
		Point	• — • — • —	Virgule	— — •• — —

Fig. 1

Le programme de la figure 2 peut être un fort bon professeur, toujours disponible pour « traduire » les mots de votre choix : commencez donc par des mots courants et courts (le fameux CQ-CQ-CQ-CQ des télégraphistes, par exemple), et vous arriverez progressivement à comprendre une partie de ces avalanches de points et de traits que l'on rencontre partout en ondes courtes ! Le cœur du logiciel n'occupe que quelques lignes de BASIC, numérotées de 10 à 510 : le reste du programme est une longue suite d'instructions « IF-THEN » contenant l'équivalent morse de toutes les lettres de l'alphabet. Grâce à l'éditeur de lignes de l'AMSTRAD, chaque ligne peut être rapidement obtenue par simple modification de la précédente.

Lorsque vous serez arrivé à « Z », vous pourrez compléter cette liste par les chiffres, en vous inspirant de la figure 1, et même ajouter les signes de ponctuation si le cœur vous en dit ! Tout est réglé dans ce logiciel pour que le son émis ressemble d'aussi près que possible à une « vraie » transmission. Il est toutefois facile de modifier les instructions SOUND pour obtenir des sons plus aigus ou plus graves, ou pour transformer la cadence de « manipulation ».

Dans cette éventualité, on retiendra qu'un trait doit durer trois fois plus longtemps qu'un point, et qu'un espace entre point et trait doit avoir la durée d'un point, tandis que deux lettres d'un même mot doivent être séparées par la durée d'un trait.

A titre indicatif, précisons aussi que l'espace entre deux mots consécutifs est en principe de sept fois la durée d'un point. Toute modification de la durée du son programmé dans SOUND devrait donc être accompagnée d'une correction des lignes 180 et 500.

Ces normes précises permettent un bon « décodage » des signaux morse non seulement par les opérateurs « humains », mais aussi par les équipements informatiques destinés à la réception automatique.

Il existe en effet des interfaces et des logiciels qui, adaptés à un micro-ordinateur AMSTRAD ou autre, permettent une interprétation automatique des signaux morse reçus.

Même les logiciels les plus élaborés (des milliers d'octets de langage machine) se révèlent assez décevants en présence de signaux morse émis « à la main » : aussi expérimenté soit-il, un opérateur est difficilement capable de respecter une cadence de manipulation parfaitement constante, tandis que même les meilleurs logiciels ont du mal à suivre ces variations.

Les choses se passent beaucoup mieux lorsque c'est un ordinateur qui manipule, en respectant scrupuleusement la cadence.

Seulement, lorsqu'il s'agit de faire communiquer deux ordinateurs entre eux par radio, il existe aujourd'hui des procédés autrement plus performants que le morse (RTTY, radio-paquets, AMTOR, etc.).

Quoi qu'il en soit, votre AMSTRAD est désormais capable de produire d'excellents signaux morse, que vous êtes évidemment libres d'utiliser à votre convenance, amis lecteurs...

```

10 CLS
20 PRINT "mot a transmettre?"
30 INPUT b$
40 CLS
60 FOR f=1 TO LEN(b$)
70 c$=MID$(b$,f,1)
80 GOSUB 1000
90 a$=a$+d$
100 NEXT f
110 CLS:PRINT b$
120 FOR f=1 TO LEN(a$)
130 z$=MID$(a$,f,1)
140 IF z$="\ " THEN GOSUB 500
150 IF z$="\" THEN 190
160 IF z$="." THEN SOUND 1,100,8,12
170 IF z$="-" THEN SOUND 1,100,24,12
175 IF z$="-" THEN GOSUB 500
180 FOR g=1 TO 100 NEXT g
190 NEXT f
200 RUN
500 FOR g=1 TO 300:NEXT g
510 RETURN
1000 IF c$=" " THEN d$=""
1001 IF c$="a" THEN d$=".-"

```

Fig. 2

```
1002 IF c#="b" THEN d#=".. ."
1003 IF c#="c" THEN d#="... ."
1004 IF c#="d" THEN d#="...."
1005 IF c#="e" THEN d#="....."
1006 IF c#="f" THEN d#="....."
1007 IF c#="g" THEN d#="....."
1008 IF c#="h" THEN d#="....."
1009 IF c#="i" THEN d#="....."
1010 IF c#="j" THEN d#="....."
1011 IF c#="k" THEN d#="....."
1012 IF c#="l" THEN d#="....."
1013 IF c#="m" THEN d#="....."
1014 IF c#="n" THEN d#="....."
1015 IF c#="o" THEN d#="....."
1016 IF c#="p" THEN d#="....."
1017 IF c#="q" THEN d#="....."
1018 IF c#="r" THEN d#="....."
1019 IF c#="s" THEN d#="....."
1020 IF c#="t" THEN d#="....."
1021 IF c#="u" THEN d#="....."
1022 IF c#="v" THEN d#="....."
1023 IF c#="w" THEN d#="....."
1024 IF c#="x" THEN d#="....."
1025 IF c#="y" THEN d#="....."
1026 IF c#="z" THEN d#="....."
1990 d#=d#+"<"
2000 RETURN
```

Fig. 2 (suite)