Comment utiliser la fonte post binaire en français pour Commodore 64

C’est pas si compliqué

2025

En cas de questions: Charlotte sur miramar.itch.io
Ou la super commu de Everything C64 sur discord

1- Lance la fonte dans ton émulateur ou ton micro-ordinateur

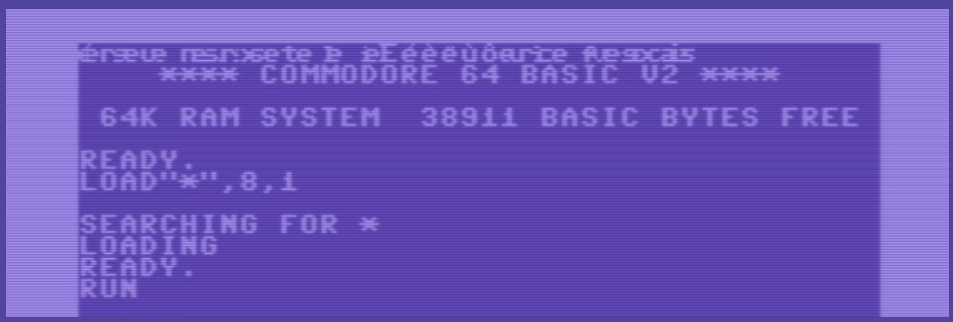
Drag and drop ton .prg, tape la commande pour charger comme d'hab:

LOAD“*”,8,1

Puis ça charge, puis RUN

Et voilà la fonte est intégrée

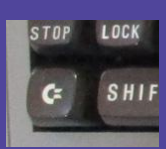
2- Teste



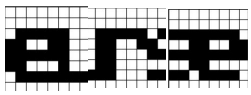
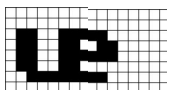
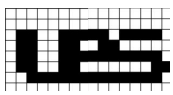
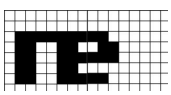
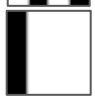
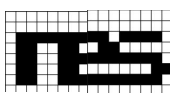
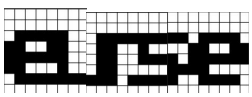
Tous les nouveaux caractères s'affichent en haut de l'écran comme ça

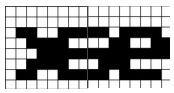
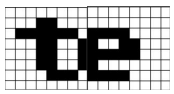
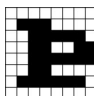
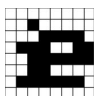
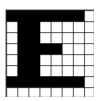
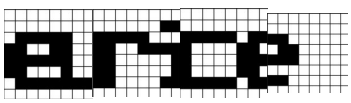
Pour afficher le caractère  tape c= K (chez moi c'est tab+K)

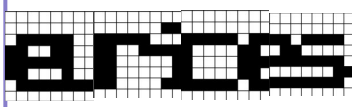
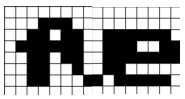
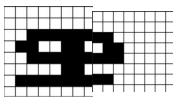
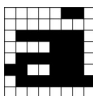
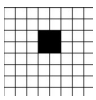
Si t'es en Europe, vérifie les paramètres d'inputs de ton émulateur, par défaut c'est en QWERTY. Tu dois donc mettre ton clavier en QWERTY pour écrire.

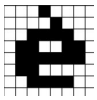
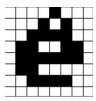
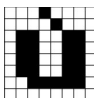
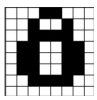
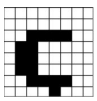
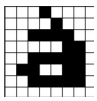


Le bouton c= c'est ce bouton Commodore sur l'ordinateur original

Caractère inclusif	Caractère(s)	Commande(s) clavier	Screen code, Caractère d'origine et code PETSCII
é·e		C=-K	97  161
eur·se		C=-Y C=-I C=-T	119  183 98  162 99  163
u·e		C=-@ (at) C=-G	100  164 101  165
u·e·s		C=-@ (at) C=-M	100  164 103  167
n·e		C=-+ C=-G	102  166 101  165
n·e·s		C=-+ C=-M	102  166 103  167
eur·se		C=-Y C=-£ C=-M	119  183 104  169 106  170

Caractère inclusif	Caractère(s)	Commande(s) clavier	Screen code, Caractère d'origine et code PETSCII
x·se		Shift-£ C=-N	105 169 106 170
t·e		C=-Q C=-D	107 171 108 172
l·e		C=-Z	109 173
e·a		C=-S	110 174
i·e		C=-P	111 175
l·E		C=-A	112 176
eur·ice		C=-Y C=-U C=-O C=-G	119 183 120 184 121 185 101 165

Caractère inclusif	Caractère(s)	Commande(s) clavier	Screen code, Caractère d'origine et code PETSCII
eur·ice·s		C=-Y C=-U C=-O C=-M	119 183 120 184 121 185 103 167
f·ve		Shift-@ C=-F	122 186 123 187
s·e		C=-C	124 188 101 165
x·c		C=-X	125 189
a·i		C=-V	126 190
· (point médian)		↑ (^ sur ordinateur moderne)	30 94

Caractère inclusif	Caractère(s)	Commande(s) clavier	Screen code, Caractère d'origine et code PETSCII
é		C=-E	113  177
è		C=-R	114  178
ê		C=-W	115  179
ù		C=-H	116  180
ô		C=-J	117  181
ç		C=-L	118  182
à		C=-B	127  191

Code entier utilisé pour la fonte poste-binaire en français

```
10 rem copy routine
11 for i=0 to 26: read x: poke 828+i,x: next i
12 data 169,000,160,208,133,095,132,096 : rem lda #0; ldy #$d0; sta 95, sty 96
13 data 169,000,160,224,133,090,132,091 : rem lda #0; ldy #$e0; sta 90; sty 91
14 data 169,000,160,064,133,088,132,089 : rem lda #0; ldy #$40; sta 88; sty 89
15 data 076,191,163 : rem jmp $a3bf
16 rem copy $d000-$dfff -> $3000-$3fff
20 rem char set rom into ram
21 poke 56334,peek(56334) and 254 : rem interrupt off
22 poke 1,peek(1) and 251 : rem char set rom on
23 sys 828 : rem start copy
24 poke 1,peek(1) or 4 : rem char set rom off
25 poke 56334,peek(56334) or 1 : rem interrupt on
26 poke 53272,peek(53272) and 240 or 12 : rem char set ram at $3000
60 rem fonte postbinaire francais
61 sc = 97
62 nc = 31
63 for a=12288+sc*8 to 12288+sc*8+nc*8-1:read ze:poke a,ze:poke a+2048,ze:next a
70 for c=sc to sc+nc-1: poke 1024-sc+c,c:next
3000 data 8,16,126,211,255,208,126,0
3010 data 0,0,125,102,99,96,227,0
3020 data 0,0,254,51,255,48,222,0
3030 data 0,0,103,102,103,102,63,0
3040 data 0,0,192,96,224,0,192,0
3050 data 0,0,127,102,103,102,103,0
3060 data 0,0,222,96,254,3,254,0
3070 data 0,0,127,100,103,96,239,0
3080 data 0,0,103,60,31,60,103,0
3090 data 0,0,222,51,255,112,222,0 : rem 170
3100 data 0,24,126,25,25,25,14,0
3110 data 0,0,240,152,248,128,240,0
3120 data 0,112,62,51,63,48,126,0
3130 data 0,0,126,203,255,211,127,0
3140 data 0,32,30,115,63,48,126,0 : rem 175
3150 data 254,96,96,120,96,96,254,0
3160 data 4,8,60,102,126,96,60,0
3170 data 16,8,60,102,126,96,60,0
3180 data 8,20,60,102,126,96,60,0
3190 data 16,8,102,102,102,102,62,0 : rem 180
3200 data 24,36,60,102,102,102,60,0
3210 data 0,0,60,96,96,96,60,8
3220 data 0,0,124,204,252,196,127,0
3230 data 0,3,248,207,193,193,195,0
3240 data 0,0,251,134,135,134,251,0 : rem 185
3250 data 0,28,54,126,54,50,49,0
3260 data 0,0,126,115,127,96,190,0
3270 data 0,0,63,102,63,6,127,0
3280 data 0,0,207,120,56,120,207,0
3290 data 6,0,126,14,126,206,127,0
3300 data 16,8,60,6,62,102,62,0
3305 sc = 30
3306 nc = 1
3310 for a=12288+sc*8 to 12288+sc*8+nc*8-1:read ze:poke a,ze:poke a+2048,ze:next a
3320 data 0,0,24,24,0,0,0,0
```

Commentaires

Tous les caractères sont consécutifs dans la mémoires, sauf le point médian qui est à un autre endroit de la mémoire. J’ai donc copié quelques lignes spécifiques au point médian.