

5/11

Tracé de cercles

Les CPC ne possèdent pas d'instruction de tracé de cercles en Basic. Nous allons bien vite combler cette lacune à l'aide de la RSX `CIRCLE`. Tout d'abord, un peu de théorie.

La méthode la plus classique pour tracer un cercle consiste à utiliser ses équations :

$$\begin{aligned} X &= XC + R \times \text{COS}(\text{alpha}) \\ Y &= YC + R \times \text{SIN}(\text{alpha}) \end{aligned}$$

où XC et YC sont les coordonnées du centre du cercle, R est le rayon du cercle, et alpha est un angle variant entre 0 et 360 degrés. Cette méthode de tracé est relativement simple. Son seul inconvénient est sa relative lenteur, même en assembleur.

Heureusement, le mathématicien Bresenham a découvert un algorithme qui offre deux avantages :

- les coordonnées sont calculées à l'aide d'opérations entières ;
- les opérations pour calculer chaque point consistent en de simples additions, et sont donc très rapidement effectuées.

Nous n'allons pas retracer le cheminement logique qui a permis à Bresenham de définir l'algorithme. Nous utiliserons simplement le résultat :

Les coordonnées du premier point du cercle sont $X=0$, $Y=R$.

La valeur $3-2 \times R$ est affectée à la variable CALC. Si cette valeur est positive, la prochaine valeur de CALC est $(X-Y) \times 4 + 10 + \text{CALC}$ et l'ordonnée du prochain point est incrémentée. Dans le cas contraire, la prochaine valeur de CALC est $X \times 4 + 6 + \text{CALC}$. Quelle que soit la valeur de CALC, l'abscisse est incrémentée. Les couples de points suivants appartiennent au cercle : (X, Y), (X, -Y), (-X, Y), (-X, -Y), (Y, X), (Y, -X), (-Y, X), (-Y, -X).

La description du cercle est complète lorsque X-Y devient positif.

Le petit programme Basic suivant applique à la lettre cet algorithme :

```
1000 REM -----
1010 REM Version BASIC du programme de trace de cercles
1020 REM -----
1030 REM
1040 CLS
1050 x=100:y=100:r=80
1060 ORIGIN x,y
1070 x=0:y=r
1080 calc=3-r*2
1090 PLOT x,y:PLOT x,-y:PLOT -x,y:PLOT -x,-y
1100 PLOT y,x:PLOT y,-x:PLOT -y,x:PLOT -y,-x
1110 IF x-y>0 THEN END
1120 IF calc>0 THEN calc=(x-y)*4+10+calc:y=y-1 ELSE calc=x*4+6+calc
1130 x=x+1
1140 GOTO 1090
```

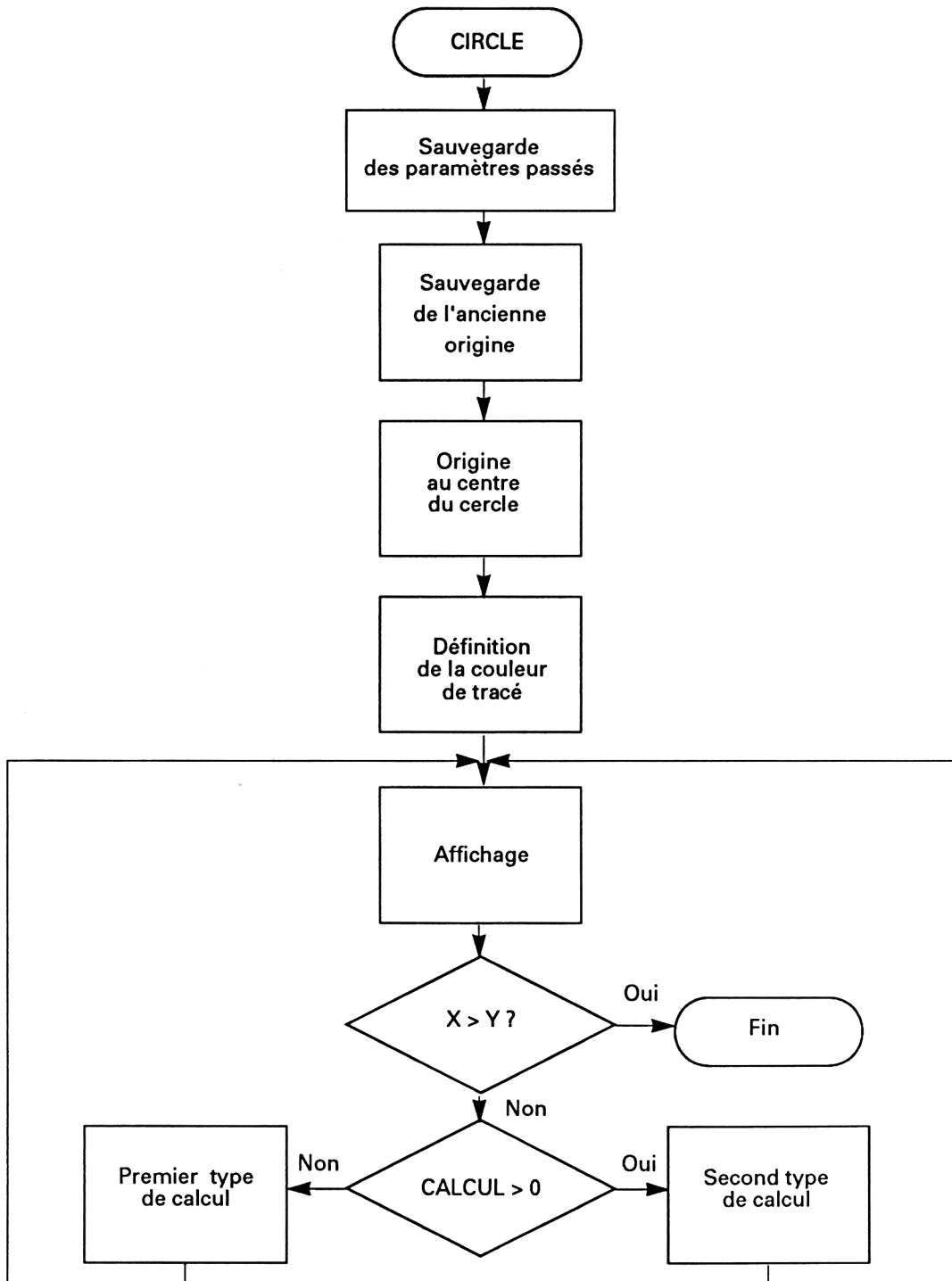
Afin de rendre le tracé de cercles plus accessible à un programme Basic, nous allons définir la RSX **!CIRCLE** dont la syntaxe est la suivante :

!CIRCLE,X,Y,R,C

où X et Y sont les coordonnées du cercle, R est le rayon du cercle, et C la couleur de tracé.

LA RSX EN DÉTAIL

La RSX `!CIRCLE` est bien entendu écrite en Assembleur. Sa logique apparaît dans l'ordinogramme suivant :



La définition de la RSX (lignes 42 à 46) fait désormais partie des opérations classiques. Reportez-vous si nécessaire aux autres RSX de l'ouvrage pour avoir plus de détails à son sujet.

La routine de traitement de la RSX se trouve à l'adresse **CIRCLE**. Les premières actions effectuées par cette routine consistent en la mémorisation des paramètres qui lui sont passés. Ces paramètres sont pointés par le registre **IX** :

- la couleur de tracé est stockée dans la variable **COUL** (lignes 53 et 54) ;
- le rayon du cercle dans la variable **RAYON** (lignes 55 à 57) ;
- les coordonnées du centre du cercle dans les variables **X** et **Y** (lignes 58 à 63).

L'origine graphique de l'écran étant redéfinie, le programme sauvegarde ensuite l'ancienne origine qui est lue à l'aide de la macro **GRA GET ORIGIN** du **FIRMWARE** (lignes 69 à 71).

La nouvelle origine est définie au centre du cercle à l'aide de la macro **GRA SET ORIGIN** du **FIRMWARE** (lignes 77 à 79).

Les coordonnées du point de départ sont initialisées ($X=0$, $Y=R$) lignes 85 à 89.

La couleur de tracé est initialisée à l'aide de la variable **COUL** et de la macro **GRA SET PEN** du **FIRMWARE** (lignes 90 et 91).

La boucle de calcul **BIS** est très proche de celle réalisée en Basic. Notez cependant que les multiplications par 4 ont été effectuées à l'aide de décalages et rotations logiques (lignes 109 et 112 et 130 à 133).

Les points calculés sont affichés à l'aide de la macro **GRA PLOT ABS** du **FIRMWARE** entre les lignes 149 et 209. De nombreuses instructions **PUSH** et **POP** utilisent la pile pour augmenter la vitesse de calcul.

Le programme se termine par la restitution de l'ancienne origine graphique (lignes 215 à 219).

Le listing de la RSX est le suivant :

```

1          ORG  9000H
2          LOAD 9000H
3          ;-----
4          ; RSX CIRCLE
5          ; Format : !CIRCLE,X,Y,R,C
6          ; Entree : X=Abscisse du rayon
7          ;          Y=Ordonnee du rayon
8          ;          R=Rayon
9          ;          C=Couleur
10         ; Sortie : Affichage du cercle
11         ;-----
12         ;
13         ;
14         ;-----
15         ; Declaration des constantes
16         ; et des variables du programme
17         ;-----
18         ;
19         LOGEXT:    EQU  0BCD1H          ;KL LOG EXT
20         GETORI:   EQU  0BBCCH          ;GRA GET ORIGIN
21         SETORI:   EQU  0BBC9H          ;GRA SET ORIGIN
22         PLOTABS:  EQU  0BBEAH          ;GRA PLOT ABS
23         SETPEN:   EQU  0BBDEH          ;GRA SET PEN
24         BUF:      DS    4              ;ZONE RAM POUR LOG EXT
25 9004 0990        PTRTAB:  DW  TABLE          ;Pointeur TABLE
26 9006 C32790      JP    CIRCLE          ;Affichage du cercle
27 9009 43495243   TABLE:  DB  "CIRCL"

```

```

27 900D 4C
28 900E C5          DB  "E"+80H
29 900F 00          DB  0          ;Fin de table
30          X:      DS  2          ;Abscisse centre
31          Y:      DS  2          ;Ordonnee centre
32          XORI:   DS  2          ;Abs ancienne origine
33          YORI:   DS  2          ;Ord ancienne origine
34          RAYON:  DS  2          ;Rayon du cercle
35          CALC:   DS  2          ;Var intermediaire
36          COUL:   DS  1          ;Couleur de trace
37          ;
38          ;-----
39          ; Definition de la RSX
40          ;-----
41          ;
42          DEFRSX: EQU  $          ;Point d'entree
43 901D 010490      LD  BC, PTRTAB  ;Ptr table definition
44 9020 210090      LD  HL, BUF    ;Buffer pour LOG EXT
45 9023 CDD1BC      CALL LOGEXT  ;Definition de la RSX
46 9026 C9          RET
47          ;
48          ;-----
49          ; Traitement de CIRCLE
50          ;-----
51          ;
52          CIRCLE: EQU  $          ;Point d'entree
53 9027 DD7E00      LD  A, (IX+0)

```

```

54 902A 321C90          LD   (COUL),A
55 902D DD6603          LD   H,(IX+3)
56 9030 DD6E02          LD   L,(IX+2)
57 9033 221890          LD   (RAYON),HL      ;Sauv Rayon
58 9036 DD6605          LD   H,(IX+5)
59 9039 DD6E04          LD   L,(IX+4)
60 903C 221290          LD   (Y),HL          ;Sauv Ordonnee
61 903F DD6607          LD   H,(IX+7)
62 9042 DD6E06          LD   L,(IX+6)
63 9045 221090          LD   (X),HL          ;Sauv Abscisse
64                      ;
65                      ;- - - - -
66                      ; Sauvegarde ancienne origine
67                      ;- - - - -
68                      ;
69 9048 CDC0BB          CALL GETORI           ;GRA GET ORIGIN
70 904B ED531490        LD   (XORI),DE
71 904F 221690          LD   (YORI),HL
72                      ;
73                      ;- - - - -
74                      ; Definition nouvelle origine
75                      ;- - - - -
76                      ;
77 9052 ED5B1090        LD   DE,(X)
78 9056 2A1290          LD   HL,(Y)
79 9059 CDC9BB          CALL SETORI           ;GRA SET ORIGIN
80                      ;

```

```

81          ;-----
82          ; Initialisations diverses
83          ;-----
84          ;
85 905C 210000      LD  HL,0
86 905F 221090      LD  (X),HL
87 9062 221A90      LD  (CALC),HL
88 9065 2A1890      LD  HL,(RAYON)
89 9068 221290      LD  (Y),HL
90 906B 3A1C90      LD  A,(COUL)
91 906E CDDEBB      CALL SETPEN          ;Couleur de trace
92          ;
93          BIS:     EQU  $          ;Boucle principale
94 9071 CDD490      CALL AFFICHE          ;Aff de 8 points
95 9074 2A1090      LD  HL,(X)
96 9077 ED5B1290    LD  DE,(Y)
97 907B AF          XOR  A
98 907C ED52        SBC  HL,DE
99 907E F23A91      JP   P,FIN          ;Fin du programme
100         ;
101 9081 2A1A90      LD  HL,(CALC)
102 9084 110000      LD  DE,0
103 9087 AF          XOR  A
104 9088 ED52        SBC  HL,DE
105 908A F2AD90      JP   P,SECOND          ;2eme type de calcul
106         ;
107          PREMIER: EQU  $          ;1er type de calcul
108 908D 2A1090      LD  HL,(X)

```



```

109 9090 CB25          SLA  L
110 9092 CB14          RL   H
111 9094 CB25          SLA  L
112 9096 CB14          RL   H                ; X*4
113 9098 110600        LD   DE,6
114 909B 19            ADD  HL,DE                ; X*4+6
115 909C ED5B1A90      LD   DE,(CALC)
116 90A0 19            ADD  HL,DE                ; X*4+6+CALC
117 90A1 221A90        LD   (CALC),HL
118                    ;
119                    SUITE: EQU  $
120 90A4 2A1090        LD   HL,(X)
121 90A7 23            INC  HL
122 90AB 221090        LD   (X),HL                ; X+1
123 90AB 18C4          JR   BIS                ; Suite du traitement
124                    ;
125                    SECOND: EQU $                ; 2eme type de calcul
126 90AD 2A1090        LD   HL,(X)
127 90B0 ED5B1290      LD   DE,(Y)
128 90B4 AF            XOR  A
129 90B5 ED52          SBC  HL,DE
130 90B7 CB25          SLA  L
131 90B9 CB14          RL   H
132 90BB CB25          SLA  L
133 90BD CB14          RL   H                ; (X-Y)*4
134 90BF 110A00        LD   DE,10
135 90C2 19            ADD  HL,DE                ; (X-Y)*4+10

```

```

136 90C3 ED5B1A90      LD   DE, (CALC)
137 90C7 19           ADD  HL, DE           ; (X-Y)*4+10+CALC
138 90C8 221A90      LD   (CALC), HL
139 90CB 2A1290      LD   HL, (Y)
140 90CE 2B          DEC  HL
141 90CF 221290      LD   (Y), HL
142 90D2 18D0        JR   SUITE
143                   ;
144                   ; - - - - -
145                   ; Affichage des points calculés
146                   ; - - - - -
147                   ;
148                   AFFICHE: EQU $
149 90D4 2A1290      LD   HL, (Y)
150 90D7 ED5B1090    LD   DE, (X)
151 90DB D5          PUSH DE
152 90DC E5          PUSH HL
153 90DD E5          PUSH HL
154 90DE D5          PUSH DE
155                   ;
156 90DF 210000      LD   HL, 0
157 90E2 ED5B1290    LD   DE, (Y)
158 90E6 AF          XOR  A
159 90E7 ED52        SBC  HL, DE
160 90E9 ED5B1090    LD   DE, (X)
161 90ED D5          PUSH DE
162 90EE E5          PUSH HL

```

```

163 90EF E5          PUSH HL
164 90F0 D5          PUSH DE          ; -Y, X
165                  ;
166 90F1 210000      LD   HL, 0
167 90F4 ED5B1090    LD   DE, (X)
168 90F8 AF          XOR  A
169 90F9 ED52        SBC  HL, DE
170 90FB ED5B1290    LD   DE, (Y)
171 90FF E5          PUSH HL
172 9100 D5          PUSH DE          ; -X, Y
173 9101 D5          PUSH DE
174 9102 E5          PUSH HL          ; Y, -X
175                  ;
176 9103 E5          PUSH HL          ; -X
177 9104 210000      LD   HL, 0
178 9107 ED5B1290    LD   DE, (Y)
179 910B AF          XOR  A
180 910C ED52        SBC  HL, DE
181 910E E5          PUSH HL
182 910F D1          POP  DE          ; -Y
183 9110 E1          POP  HL          ; -X
184 9111 E5          PUSH HL
185 9112 D5          PUSH DE          ; -X, -Y
186                  ;
187 9113 CDEABB      CALL PLOTABS     ; -Y, -X
188 9116 E1          POP  HL
189 9117 D1          POP  DE

```

```

190 9118 CDEABB          CALL PLOTABS          ; -X, -Y
191 911B E1              POP HL
192 911C D1              POP DE
193 911D CDEABB          CALL PLOTABS          ; Y, -X
194 9120 E1              POP HL
195 9121 D1              POP DE
196 9122 CDEABB          CALL PLOTABS          ; -X, Y
197 9125 E1              POP HL
198 9126 D1              POP DE
199 9127 CDEABB          CALL PLOTABS          ; -Y, X
200 912A E1              POP HL
201 912B D1              POP DE
202 912C CDEABB          CALL PLOTABS          ; X, -Y
203 912F E1              POP HL
204 9130 D1              POP DE
205 9131 CDEABB          CALL PLOTABS          ; Y, X
206 9134 E1              POP HL
207 9135 D1              POP DE
208 9136 CDEABB          CALL PLOTABS          ; X, Y
209 9139 C9              RET
210                      ;
211                      ; - - - - -
212                      ; Restitution de l'origine
213                      ; - - - - -
214                      ;
215                      FIN:      EQU  $
216 913A EDSB1490        LD      DE, (XORI)

```

```
217 913E 2A1690      LD   HL,(YORI)
218 9141 CDC9BB      CALL SETORI          ;GRA SET ORIGIN
219 9144 C9          RET
220                  END
```

```
AFFICHE      90D4 BUF      9000 BIS      9071 CALC      901A
COUL         901C CIRCLE  9027 DEFRSX   901D FIN       913A
GETORI       BBCC LOGEXT  BCD1 PLOTABS  BBEA PTRTAB    9004
PREMIER      908D RAYON   9018 SETORI   BBC9 SETPEN    BBDE
SUITE        90A4 SECOND  90AD TABLE   9009 X         9010
XORI         9014 Y       9012 YORI     9016
```

Comme toujours, voici la version chargeur Basic, bien plus pratique à utiliser :

```

1000 REM -----
1010 REM Chargeur BASIC de la RSX de trace de cercles
1020 REM -----
1030 REM
1040 FOR i=&9000 TO &9144
1050   READ a$
1060   a$="&" + a$
1070   POKE i,VAL(a$)
1080 NEXT i
1090 CALL &901D
1100 END
1110 REM - - - - -
1120 REM Donnees du programme de trace
1130 REM - - - - -
1140 REM
1150 DATA FC,A6,4,90,9,90,C3,27,90,43,49,52,43,4C,C5,0
1160 DATA C,0,B,0,0,0,0,11,0,C,0,2,1,4,90
1170 DATA 21,0,90,CD,D1,BC,C9,DD,7E,0,32,1C,90,DD,66,3
1180 DATA DD,6E,2,22,18,90,DD,66,5,DD,6E,4,22,12,90,DD
1190 DATA 66,7,DD,6E,6,22,10,90,CD,CC,BB,ED,53,14,90,22
1200 DATA 16,90,ED,5B,10,90,2A,12,90,CD,C9,BB,21,0,0,22
1210 DATA 10,90,22,1A,90,2A,18,90,22,12,90,3A,1C,90,CD,DE
1220 DATA BB,CD,D4,90,2A,10,90,ED,5B,12,90,AF,ED,52,F2,3A
1230 DATA 91,2A,1A,90,11,0,0,AF,ED,52,F2,AD,90,2A,10,90
1240 DATA CB,25,CB,14,CB,25,CB,14,11,6,0,19,ED,5B,1A,90
1250 DATA 19,22,1A,90,2A,10,90,23,22,10,90,18,C4,2A,10,90
1260 DATA ED,5B,12,90,AF,ED,52,CB,25,CB,14,CB,25,CB,14,11
1270 DATA A,0,19,ED,5B,1A,90,19,22,1A,90,2A,12,90,2B,22
1280 DATA 12,90,18,D0,2A,12,90,ED,5B,10,90,D5,E5,E5,D5,21
1290 DATA 0,0,ED,5B,12,90,AF,ED,52,ED,5B,10,90,D5,E5,E5
1300 DATA D5,21,0,0,ED,5B,10,90,AF,ED,52,ED,5B,12,90,E5
1310 DATA D5,D5,E5,E5,21,0,0,ED,5B,12,90,AF,ED,52,E5,D1
1320 DATA E1,E5,D5,CD,EA,BB,E1,D1,CD,EA,BB,E1,D1,CD,EA,BB
1330 DATA E1,D1,CD,EA,BB,E1,D1,CD,EA,BB,E1,D1,CD,EA,BB,E1
1340 DATA D1,CD,EA,BB,E1,D1,CD,EA,BB,C9,ED,5B,14,90,2A,16
1350 DATA 90,CD,C9,BB,C9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

```

et les données de checksum correspondantes :

```
81 CB 5A 55 E0 F3 98 C2 63 C5 3E 8E 17 DA 67 A2 2C 62 5A 66 AD
```

Le petit programme de démonstration suivant montre qu'il est très simple d'utiliser la RSX `!CIRCLE` :

```
100 REM -----
110 REM Programme de demonstration
120 REM -----
130 REM
140 MODE 1
150 FOR I=1 TO 50
160   X=INT(RND(1)*600)
170   Y=INT(RND(1)*400)
180   R=INT(RND(1)*100)
190   C=INT(RND(1)*3)+1
200   !CIRCLE,X,Y,R,C
210 NEXT I
```

Avant de pouvoir utiliser `!CIRCLE`, il faut bien entendu l'avoir installée, par exemple avec le chargeur Basic précédent.

