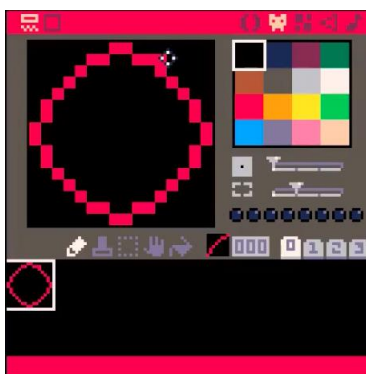
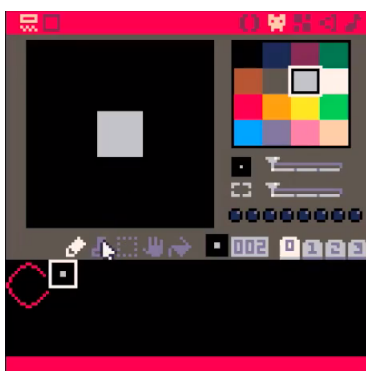


## #002



L'objectif est de déplacer une mire au-dessus d'une carte. La mire est un sprite de 16x16 pixels (soit 4 emplacements 8x8 pixels). Elle sera placée au centre de l'écran.



La carte est constituée de marques de petite taille (carrés gris clair 2x2 dans un emplacement 8x8) et de plus grande taille (carrés gris foncé 8x8).



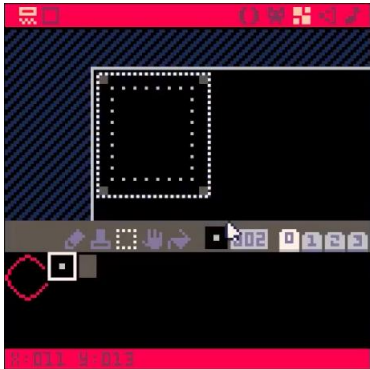
Les outils **crayon** et **remplissage** sont utilisés pour la création des différents visuels.



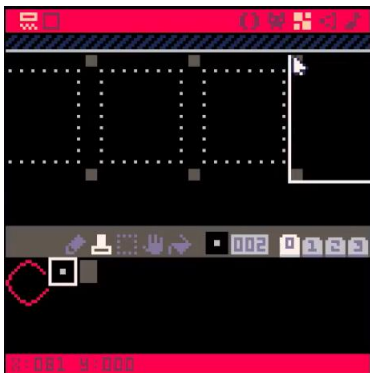
La carte est constituée de blocs répétés de 10x11 sprites. Les coins sont signalés par les marques de grande taille. Un rectangle de délimitation est dessiné à l'aide des petites marques à l'intérieur de l'espace délimité.

La taille idéale pour répéter un motif sur une carte est 16x16 sprites  $\Leftrightarrow$  taille de l'écran (128x128 pixels) exprimée en sprites. Les multiples et sous-multiples (de 16) donnent également de bons résultats.

Le choix 10x11 est arbitraire.

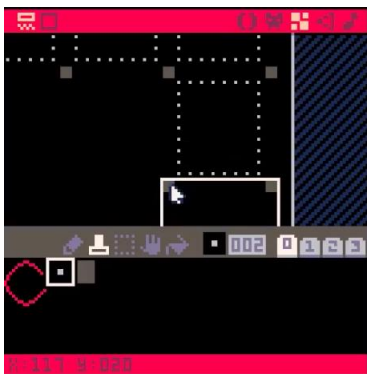


L'outil **copier** permet de mémoriser le modèle du bloc à dupliquer. La sélection s'effectue à la souris en encadrant le bloc (clic sur le coin supérieur haut gauche du bloc maintenu jusqu'au coin inférieur bas droit). Elle est mémorisée suite à une pression sur **[Ctrl]-[C]** (sous Windows et Linux, [Pomme]-[C] sous OSX).



L'outil **coller** reproduit le bloc mémorisé sur l'emplacement signalé par le cadre blanc. Les coins des différents blocs sont superposés pour dessiner le contour de la carte.

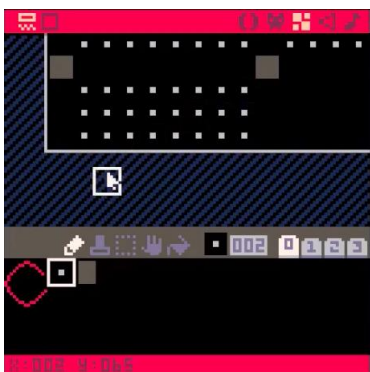
La largeur et la hauteur de la carte ne correspondent pas à un nombre de bloc entier.



Des espaces résiduels restent disponibles en haut et bas de la carte. La largeur de la carte est de :

- 14 (blocs)
- X 9 (largeur du bloc - 1 colonne de superposition)
- + 1 colonne non superposée
- +1 (largeur de l'espace résiduel)

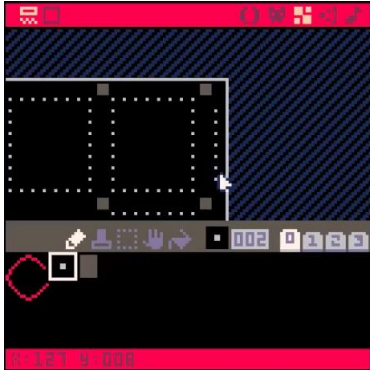
Soit 128 sprites de large, l'équivalent de 8 écrans.



La petite marque est employée en bas à gauche et en haut à droite pour mettre en évidence l'espace résiduel. La hauteur de la carte est de :

- 6 (blocs)
- x 10 (hauteur du bloc- 1 ligne de superposition)
- + 1 ligne non superposée
- +3 (hauteur de l'espace résiduel)

Soit 64 sprites de haut, l'équivalent de 4 écrans.



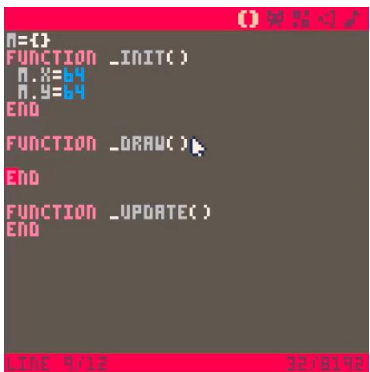
La carte peut également être utilisée pour stocker 32 (8x4) écrans (ex. niveaux d'un jeu de plate-forme).

ATTENTION :

- Les écrans stockés dans la carte sont composés de sprites. Ces écrans ne sont pas éditables au niveau du pixel.
- La carte et les sprites partagent une portion de mémoire. Lorsque l'ensemble de la carte est utilisée, seules les banques 0 et 1 des sprites sont exploitables.



Une fois les éléments graphiques en place, la programmation peut débuter. La structure du programme reprend celle précédemment décrite (voir #001).



La variable (de type table) M comporte les coordonnées du centre de la mire à l'écran. Au lancement, celle-ci est placée dans le coin supérieur haut gauche de la carte (64,64).



Le déplacement de la carte est obtenu par une translation de l'origine (0,0) vers la position courante (M.X-64, M.Y-64) à l'aide de l'instruction **CAMERA**. La position courante correspond au coin, supérieur gauche de l'écran (d'où -64 en X et Y correspondant à la position du centre de la mire à l'écran).

La recopie de la carte est effectuée en intégralité à l'aide de la commande **MAP**.

```

END
FUNCTION _DRAW()
--CAMERA(0,0)
RECTFILL(0,0,127,127,15)
--CAMERA(n.X-64,n.Y-64)
MAP(
0,
0,
-n.X+64,
-n.Y+64,
128,64)
--CAMERA(0,0)
SPR(0,56,56,2,2)
COLOR(1)
CURSOR(56,0)
PRINT(n.X)
PRINT(n.Y)
END
L'UNE 25/94 175/8192

```

Une autre solution (sans l'emploi de l'instruction **CAMERA**) consiste à effectuer la translation lors de la copie de la carte à l'écran.

#### MAP(

0, colonne (x) du sprite de départ sur la carte

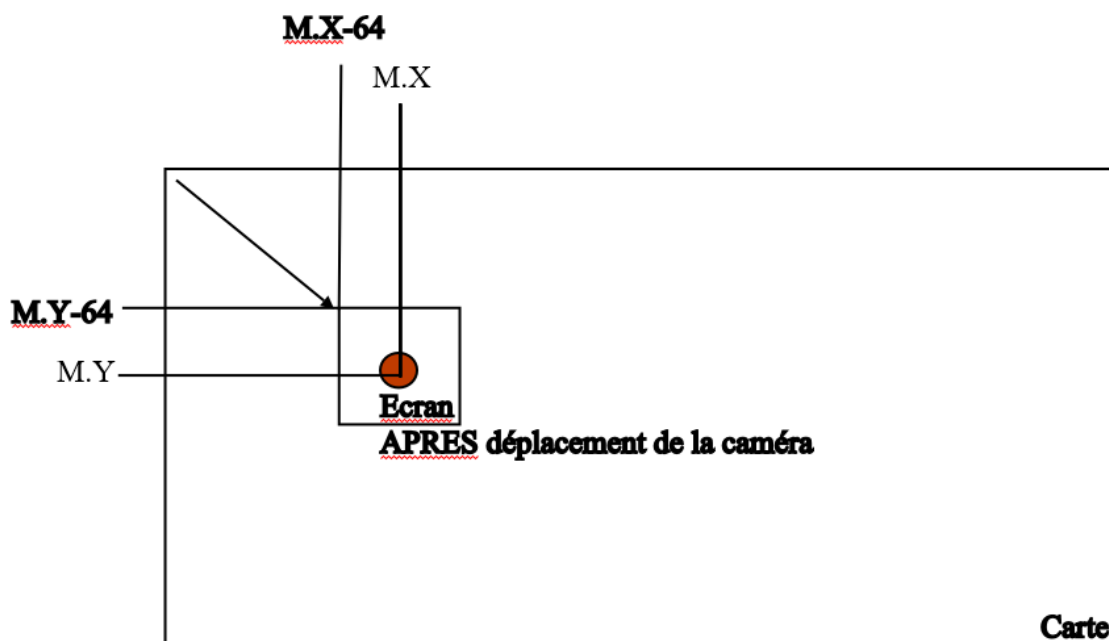
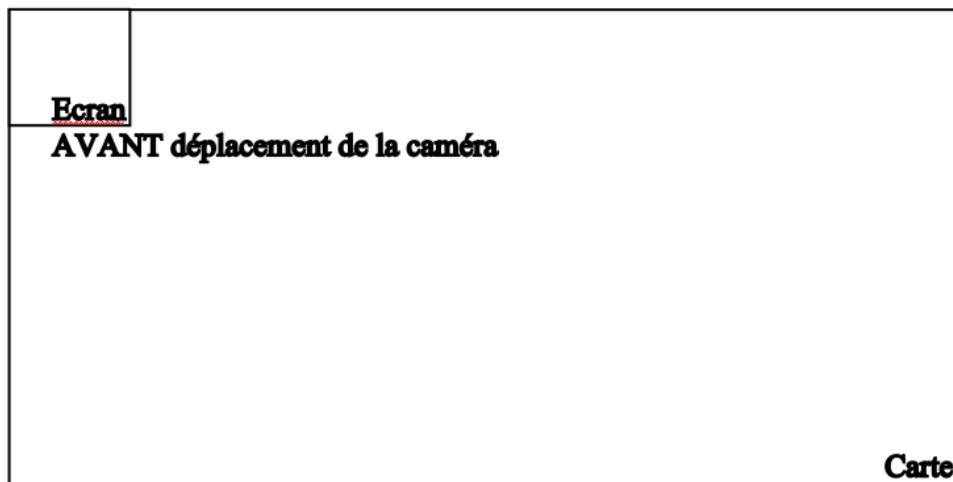
0, ligne (y) du sprite de départ sur la carte

-M.X+64, position x d'arrivée (en pixels) à l'écran

-M.Y+64, position y d'arrivée (en pixels) à l'écran

130, nombre de sprites copiés (en largeur)

124) nombre de sprites copiés (en hauteur)

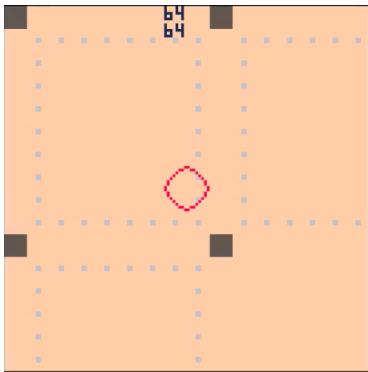


```

M={1}
FUNCTION _INIT()
  M.X=64
  M.Y=64
END
FUNCTION _DRAW()
  CAMERA(0,0)
  RECTFILL(0,0,127,127,15)
  CAMERA(M.X-64,M.Y-64)
  MAP(0,0,0,0,128,64)
  CAMERA(0,0)
  SPR(0,56,56,2,2)
  COLOR(1)
  CURSOR(56,0)
  PRINT(M.X)
  PRINT(M.Y)
END
LINE 20/39 191/8192

```

La séquence **COLOR**, **CURSOR**, **PRINT** présente les coordonnées de la mire à l'écran. Les prints successifs s'affichent les uns sous les autres. La position de départ est fixée par l'instruction **COLOR**.



[Echap.], **RUN** donne un aperçu du résultat. Les coordonnées de la mire sont affichées à l'écran (64, 64). La portion de la carte souhaitée s'affiche (30 fois par seconde). La mire se superpose à la carte.

```

FUNCTION _UPDATE()
  IF BTN(0) AND
  M.X>64 THEN
  M.X-=1
  END
  IF BTN(1) AND
  M.X<960 THEN
  M.X+=1
  END
  IF BTN(2) AND
  M.Y>64 THEN
  M.Y-=1
  END
  IF BTN(3) AND
  M.Y<448 THEN
  M.Y+=1
  END
END
LINE 34/37 202/8192

```

La fonction **\_UPDATE** accueille la gestion de l'interaction avec l'utilisateur. Les boutons 0 à 3 (associés aux touches fléchées) sont exploités pour déplacement la carte à l'écran. Les déplacements sur les axes X et Y sont limités pour que l'ensemble de l'écran soit occupé par la carte.

Il est possible de modifier les limites afin d'atteindre les rebords de la carte en modifiant les conditions :

- M.X>64 → M.X>0
- M.X<960 → M.X<1024
- M.Y>64 → M.Y>0
- M.Y<448 → M.Y<512