

T.P. 9

Space Invaders (partie 12)

Étape 1

Notre objectif est maintenant de gérer les tirs d'envahisseurs. Commençons par définir leurs bitmaps :

```

InvaderShot1_Bitmap dc.w    4,6
                   dc.b    %11000000
                   dc.b    %11000000
                   dc.b    %00110000
                   dc.b    %00110000
                   dc.b    %11000000
                   dc.b    %11000000

InvaderShot2_Bitmap dc.w    4,6
                   dc.b    %00110000
                   dc.b    %00110000
                   dc.b    %11000000
                   dc.b    %11000000
                   dc.b    %00110000
                   dc.b    %00110000

```

Le nombre de tirs maximal sera défini dans une constante que nous appellerons `INVADER_SHOT_MAX`. Vous initialiserez cette valeur à cinq. Cela signifie que nous pourrons avoir au maximum cinq tirs d'envahisseurs présents sur l'écran (jamais plus). Pour la suite, vous utiliserez cette constante plutôt que directement la valeur 5.

```
INVADER_SHOT_MAX    equ    5
```

À l'aide de la directive `DS`, réservez maintenant un espace mémoire qui servira à stocker les cinq sprites de tirs. Vous nommerez cet espace mémoire `InvaderShots`.

```
InvaderShots        ds.b    SIZE_OF_SPRITE*INVADER_SHOT_MAX
```

Réalisez le sous-programme `InitInvaderShots` qui initialisera ces cinq sprites de la façon suivante :

| Champ | Valeur |
|---------|---------------------|
| STATE | HIDE |
| X | Peu importe |
| Y | Peu importe |
| BITMAP1 | InvaderShot1_Bitmap |
| BITMAP2 | InvaderShot2_Bitmap |

Suivez la procédure ci-dessous afin de tester votre sous-programme :

- Mettez en commentaire toutes les instructions de votre programme principal.
- Tapez le programme principal ci-dessous :

```

Main      lea    InvaderShots,a0
          lea    InvaderShot1_Bitmap,a1
          lea    InvaderShot2_Bitmap,a2
\loop    move.w  #INVADER_SHOT_MAX*SIZE_OF_SPRITE-1,d7
          move.b #a0,0(a0,d7.w)
          dbra  d7,\loop

          jsr    InitInvaderShots

```

- Assemblez et lancez le débogueur.
- Agrandissez la fenêtre du débogueur (en plein écran si possible).
- Cliquez sur l'onglet **[Mixte (3)]** afin d'afficher le contenu de la mémoire. Si tout est correct, le contenu devrait s'afficher à partir de l'adresse \$500.
- Effectuez un clic droit dans la vue sur la mémoire, puis dans le menu contextuel, sélectionnez l'option *Nombre d'octets par ligne / Automatique*.
- Vous allez maintenant régler le nombre d'octets par ligne à 14. Pour cela, placez le pointeur de la souris sur la barre verticale séparant les deux vues (vue de désassemblage et vue sur la mémoire), puis tout en maintenant le bouton gauche appuyé, déplacer la barre verticale jusqu'à ce que l'adresse de la deuxième ligne soit \$50E.
- Dans la vue de désassemblage, placez un point d'arrêt à l'adresse \$520, appuyez sur **[F9]**, puis supprimez le point d'arrêt. Effectuez de nouveau un clic droit dans la vue sur la mémoire, puis dans le menu contextuel, sélectionnez l'option *Adresse de départ de la vue / A0*.
- Appuyez ensuite sur **[F10]** pour exécuter votre sous-programme.
- Le résultat de l'exécution doit ressembler à la capture d'écran ci-dessous :

| Valeur de A0 | Valeur de A1 | Valeur de A2 | |
|--------------|-------------------|-------------------|-----------------------------|
| 000DDC | 00 00 AA AA AA AA | 00 00 0F 72 | 00 00 0F 7C ..aaaa...r... |
| 000DEA | 00 00 AA AA AA AA | 00 00 0F 72 | 00 00 0F 7C ..aaaa...r... |
| 000DF8 | 00 00 AA AA AA AA | 00 00 0F 72 | 00 00 0F 7C ..aaaa...r... |
| 000E06 | 00 00 AA AA AA AA | 00 00 0F 72 | 00 00 0F 7C ..aaaa...r... |
| 000E14 | 00 00 AA AA AA AA | 00 00 0F 72 | 00 00 0F 7C ..aaaa...r... |
| 000E22 | 00 18 00 10 00 FF | 00 00 FF 00 | 3F FF FC 3Fÿ..ÿ.?ÿü? |
| 000E30 | FF FC FF FF FF FF | FF FF FC 3C | 3F FC 3C 3F ÿÿÿÿÿÿÿÿü<?ü<? |
| 000E3E | FF FF FF FF FF FF | 03 C3 C0 03 C3 C0 | 0F 3C ÿÿÿÿÿÿ.ÄÄ.ÄÄ.< |
| 000E4C | F0 0F 3C F0 F0 00 | 0F F0 00 0F 00 | 18 00 10 ð.<ðð..ð..... |
| 000E5A | 00 FF 00 00 FF 00 | 3F FF FC 3F FF | FC FF FF .ÿ..ÿ.?ÿü?ÿÿÿÿÿ |
| 000E68 | FF FF FF FF FC 3C | 3F FC 3C 3F FF | FF FF FF FF ÿÿÿÿü<?ü<?ÿÿÿÿÿ |

- Seules les valeurs dans les encadrés rouges et bleus sont à prendre en compte. Vous devez obtenir les mêmes valeurs numériques que celles présentes dans les encadrés bleus. Par contre, les valeurs numériques dans les encadrés rouges peuvent être différentes selon votre programme, mais vérifiez bien que les valeurs présentes dans les encadrés rouges sont bien celles de vos registres **A0**, **A1** et **A2**.

Une fois que votre sous-programme fonctionne correctement, vous pouvez supprimer le programme principal de test. Enlevez maintenant les commentaires de votre programme principal et ajoutez-y un appel à `InitInvaderShots` :

```

Main          jsr    InitInvaders
              jsr    InitInvaderShots

\loop        ; ...
             ; ...

```

Étape 2

Nous devons maintenant écrire un sous-programme qui renvoie l'adresse d'un tir disponible. Le moment venu, cela nous permettra d'associer un tir disponible à un envahisseur. Nous considérerons qu'un tir est disponible si son état est `HIDE` et indisponible si son état est `SHOW`. En effet, étant donné que nous avons cinq tirs au maximum (valeur contenue dans `INVADER_SHOT_MAX`), si les cinq tirs sont déjà affichés, nous ne pourrions pas associer un nouveau tir à un envahisseur.

Réalisez le sous-programme **GetHiddenShot** qui renvoie l'adresse d'un tir disponible.

Sortie : **Z** renvoie *false* (0) si aucun tir disponible n'a été trouvé.

Z renvoie *true* (1) si un tir disponible a été trouvé.

Si **Z** renvoie *false*, **A0.L** est perdu.

Si **Z** renvoie *true*, **A0.L** renvoie l'adresse du tir disponible.

Étape 3

Il faut pouvoir connecter un tir disponible à un envahisseur. La connexion entre un tir et un envahisseur se caractérisera de la manière suivante :

- Le tir sera centré horizontalement juste sous le sprite de l'envahisseur.
- L'état du tir sera positionné à `SHOW`.

Réalisez le sous-programme **ConnectInvaderShot** qui connecte un tir disponible à un envahisseur.

Entrée : **A1.L** pointe sur l'envahisseur que l'on veut connecter à un tir disponible.

Sortie : Si aucun tir n'est disponible, aucune connexion possible.

Si un tir est disponible, mais que l'envahisseur n'est pas visible, aucune connexion possible.

Si un tir est disponible et que l'envahisseur est visible :

- Positionne le tir au niveau de l'envahisseur.
- Positionne l'état du tir à `SHOW`.

Indications :

- Si l'envahisseur n'est pas visible, sortir du sous-programme.
- Récupère l'adresse d'un tir disponible (si aucun tir disponible, sortir du sous-programme).
- L'abscisse du tir est centrée horizontalement par rapport à l'envahisseur.
- L'ordonnée du tir est placée juste au-dessous de l'envahisseur.
- Le tir est rendu visible.