

# T.P. 7

## Space Invaders (partie 10)

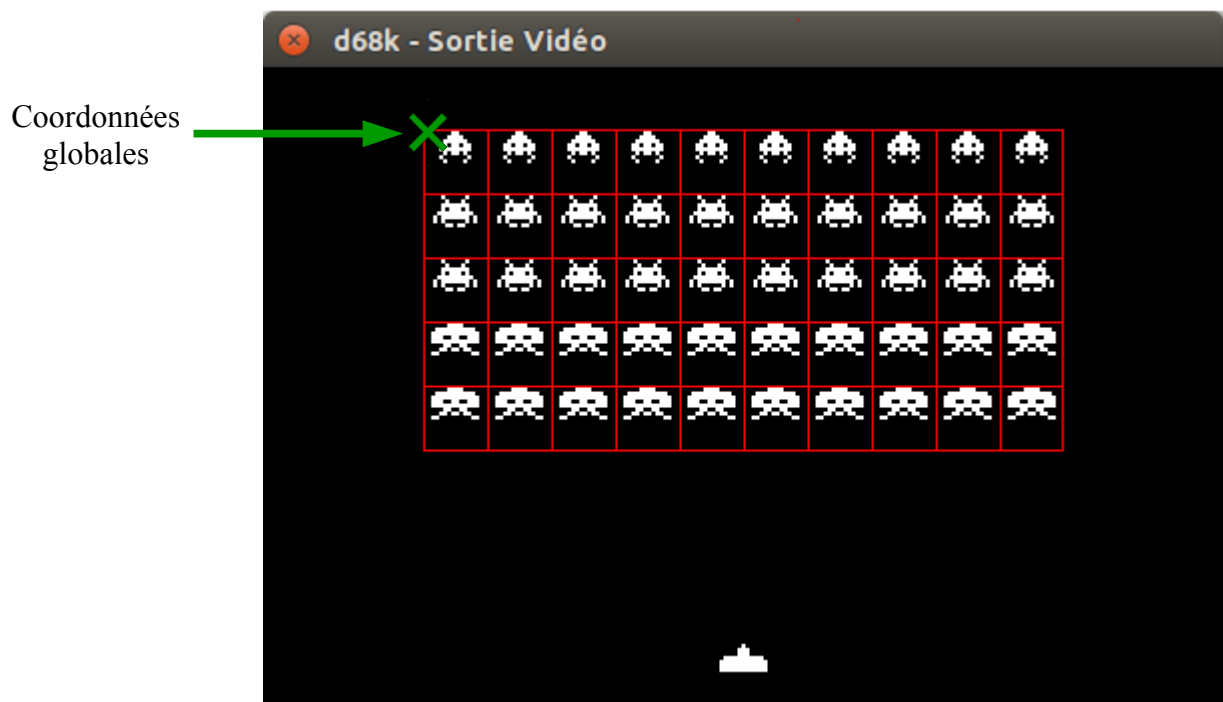
### Étape 1

Nous allons commencer la gestion du déplacement des envahisseurs. Les envahisseurs se déplaceront horizontalement jusqu'à atteindre un bord de l'écran, à ce moment-là, ils descendront d'un cran et repartiront horizontalement dans l'autre sens.

Tout d'abord, il faut définir quelques constantes qui seront utiles pour la mise au point de nos sous-programmes :

Constante	Description
INVADER_STEP_X	Pas en pixels du déplacement horizontal des envahisseurs.
INVADER_STEP_Y	Pas en pixels du déplacement vertical des envahisseurs.
INVADER_X_MIN	Abscisse globale minimale des envahisseurs. Si l'abscisse globale des envahisseurs devient inférieure à cette limite, cela signifie que les envahisseurs ont atteint le bord gauche.
INVADER_X_MAX	Abscisse globale maximale des envahisseurs. Si l'abscisse globale des envahisseurs devient supérieure à cette limite, cela signifie que les envahisseurs ont atteint le bord droit.

Pour rappel, la position globale des envahisseurs correspond aux coordonnées du coin supérieur gauche du tout premier carré contenant les envahisseurs :

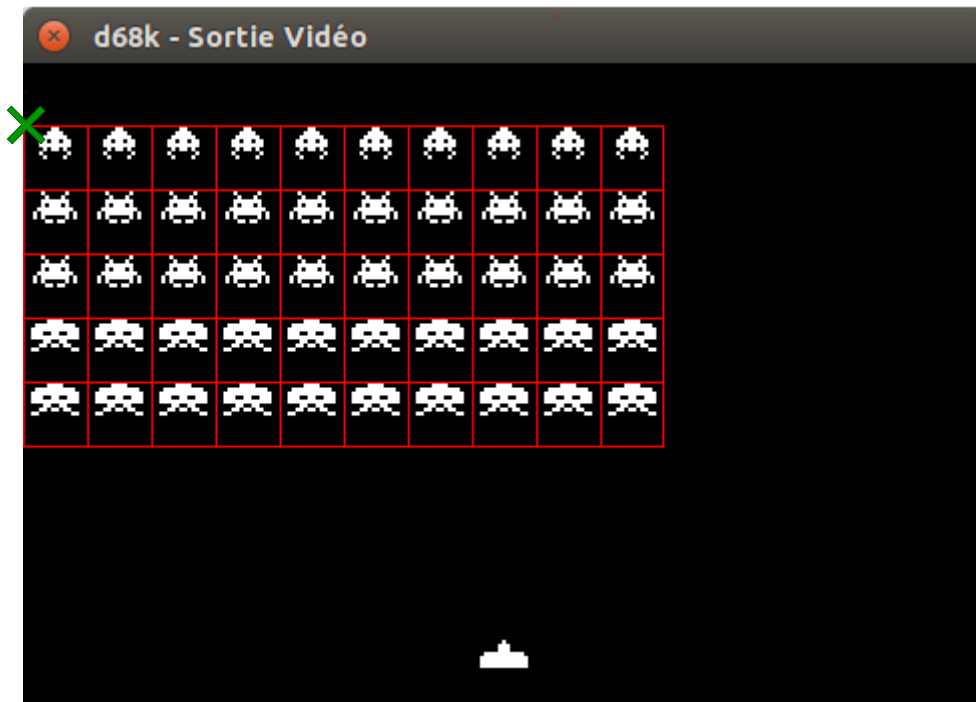


Ajoutez les lignes suivantes dans la partie *Définition des constantes* de votre fichier source :

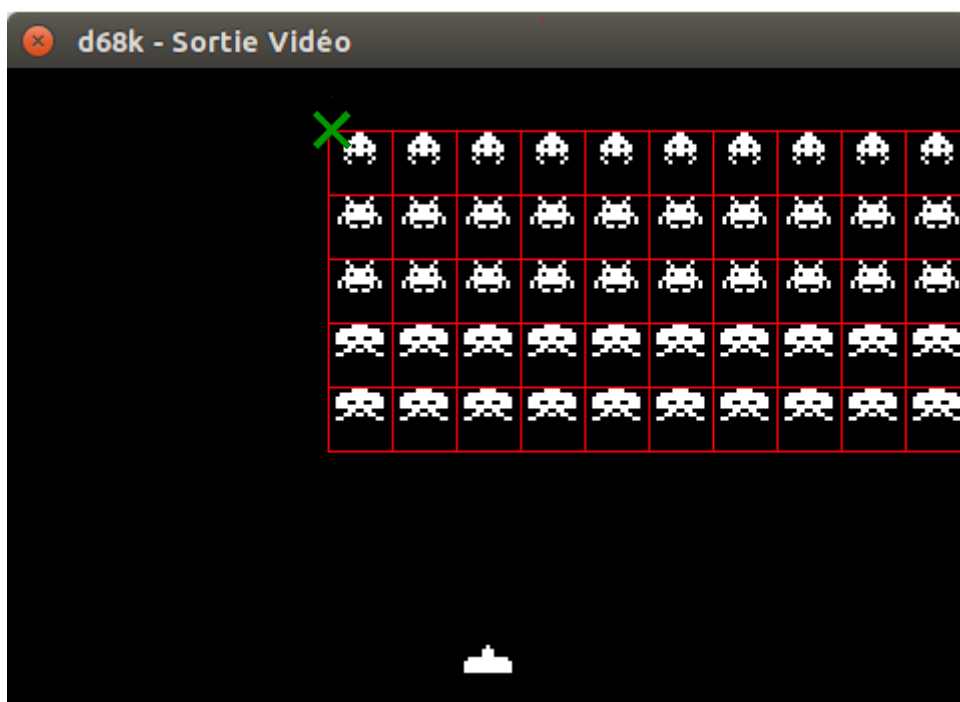
INVADER_STEP_X	equ	4
INVADER_STEP_Y	equ	8
INVADER_X_MIN	equ	0
INVADER_X_MAX	equ	(VIDEO_WIDTH-(INVADER_PER_LINE*32))

À partir des valeurs numériques ci-dessus, nous pouvons en déduire que :

- lorsque les envahisseurs se déplacent vers la gauche (avec un pas de 4 pixels) et que leur abscisse globale atteint 0, alors ils devront descendre de 8 pixels et repartir vers la droite ;



- lorsque les envahisseurs se déplacent vers la droite (avec un pas de 4 pixels) et que leur abscisse globale atteint  $480 - 320 = 160$ , alors ils devront descendre de 8 pixels et repartir vers la gauche.



Comme cela a été vu précédemment, les coordonnées globales des envahisseurs sont contenues dans les emplacements mémoire `InvaderX` et `InvaderY` :

<code>InvaderX</code>	<code>dc.w</code>	<code>(VIDEO_WIDTH-(INVADER_PER_LINE*32))/2</code>	<code>; Abscisse globale</code>
<code>InvaderY</code>	<code>dc.w</code>	<code>32</code>	<code>; Ordonnée globale</code>

Nous allons ajouter un emplacement mémoire qui contiendra le pas de déplacement horizontal des envahisseurs en cours d'utilisation :

<code>InvaderX</code>	<code>dc.w</code>	<code>(VIDEO_WIDTH-(INVADER_PER_LINE*32))/2</code>	<code>; Abscisse globale</code>
<code>InvaderY</code>	<code>dc.w</code>	<code>32</code>	<code>; Ordonnée globale</code>
<code>InvaderCurrentStep</code>	<code>dc.w</code>	<code>INVADER_STEP_X</code>	<code>; Pas en cours</code>

L'emplacement mémoire `InvaderCurrentStep` ne peut prendre que deux valeurs :

- `INVADER_STEP_X` quand les envahisseurs se déplacent vers la droite ;
- `-INVADER_STEP_X` quand les envahisseurs se déplacent vers la gauche.

Nous l'initialiserons à `INVADER_STEP_X`, car les envahisseurs commencent par se déplacer vers la droite.

Réalisez maintenant le sous-programme **GetInvaderStep** qui renvoie les prochains déplacements relatifs des envahisseurs et qui met à jour les coordonnées globales ainsi que le pas de déplacement courant.

Sorties : **D1.W** = Prochain mouvement relatif horizontal en pixels.

**D2.W** = Prochain mouvement relatif vertical en pixels.

`InvaderX`, `InvaderY` et `InvaderCurrentStep` sont mis à jour.

### Indications :

- Le sous-programme doit commencer par calculer la nouvelle abscisse globale en ajoutant le pas en cours d'utilisation à l'abscisse globale des envahisseurs.
- Si la nouvelle abscisse globale est inférieure à `INVADER_X_MIN`, alors il faut changer de direction.
- Si la nouvelle abscisse globale est supérieure à `INVADER_X_MAX`, alors il faut changer de direction.
- S'il n'y a aucun changement de direction :
  - ➔ le pas en cours est mis dans **D1.W** (déplacement horizontal en fonction du sens) ;
  - ➔ un pas de 0 est mis dans **D2.W** (pas de déplacement vertical) ;
  - ➔ `InvaderX` est mis à jour avec la nouvelle abscisse globale.
- S'il y a un changement de direction :
  - ➔ un pas de 0 est mis dans **D1.W** (pas de déplacement horizontal) ;
  - ➔ un pas de `INVADER_STEP_Y` est mis dans **D2.W** (déplacement vertical) ;
  - ➔ `InvaderY` est mis à jour (on y ajoute `INVADER_STEP_Y`) ;
  - ➔ le signe de `InvaderCurrentStep` est inversé.

## Étape 2

Réalisez le sous-programme **MoveAllInvaders** qui déplace tous les envahisseurs.

### Indications :

- Commencez par récupérer les déplacements relatifs à l'aide de **GetInvaderStep**.
- Appliquez ensuite ces déplacements (à l'aide de **MoveSprite**) à tous les envahisseurs dont l'état n'est pas HIDE.

Testez votre sous-programme **MoveAllInvaders** (et par la même occasion **GetInvaderStep**) à l'aide du programme principal suivant :

Main	jsr	InitInvaders
\loop	jsr	PrintShip
	jsr	PrintShipShot
	jsr	PrintInvaders
	jsr	BufferToScreen
	jsr	MoveShip
	jsr	MoveAllInvaders
	jsr	MoveShipShot
	jsr	NewShipShot
	bra	\loop

Vérifiez que les envahisseurs se déplacent correctement de la gauche vers la droite, puis de la droite vers la gauche. Vérifiez également qu'ils descendent bien à chaque changement de direction. Pour le moment, l'arrêt des envahisseurs en bas de l'écran n'est pas pris en compte. Il sera géré dans une prochaine étape.

## Étape 3

Pour l'instant, le vaisseau se déplace à la même vitesse que les envahisseurs. Nous allons ralentir ces derniers afin de rendre le vaisseau plus rapide. Réalisez pour cela le sous-programme **MoveInvaders** qui appelle le sous-programme **MoveAllInvaders** une fois sur huit. C'est-à-dire :

- Le premier appel de **MoveInvaders** appelle **MoveAllInvaders** ;
- Le deuxième appel de **MoveInvaders** ne fait rien ;
- Le troisième appel de **MoveInvaders** ne fait rien ;
- Etc. ;
- Le huitième appel de **MoveInvaders** ne fait rien ;
- Le neuvième appel de **MoveInvaders** appelle **MoveAllInvaders** ;
- Le dixième appel de **MoveInvaders** ne fait rien ;
- Etc.

Remplacez le `jsr MoveAllInvaders` par un `jsr MoveInvaders` dans le programme principal précédent et vérifiez que votre vaisseau se déplace plus rapidement que les envahisseurs.