

# Manuel du débogueur d'Oricutron

par André C.

Cette version française a été établie d'après le fichier ReadMe.txt du 15/05/2020  
(Oricutron version win64-20200515).

*English speaking Oricians can directly use the official Oricutron Manual  
(ReadMe.txt file released with Oricutron).*

Au lancement d'Oricutron, il faut presser sur F1 pour accéder au menu principal, qui permet entre autres choses, d'ajouter un clavier Oric virtuel, ce qui peut être utile, voir indispensable.

## Touches actives dans les menus en général

<b>Flèches</b>	Se déplacer dans le menu
<b>Return</b>	Exécuter l'option choisie
<b>DEL</b>	Revenir en arrière
<b>ESC</b>	Quitter le menu (ou touche Backspace ou BACK avec la souris)

## Touches actives dans le mode émulateur

<b>F1</b>	Aller au menu principal (sortie par ESC ou Backspace ou BACK avec la souris).
<b>F2</b>	Entrer dans le mode Débogueur / Moniteur (sortie par F2).
<b>F3</b>	Reset soft (NMI)
<b>F4</b>	Reset hard (reboot)
<b>Shift+F4</b>	Reset Jasmin
<b>F5</b>	Basculer entre les 3 états de la ligne d'information sous l'écran.
<b>F6</b>	Basculer entre vitesse normale / vitesse x20
<b>F7</b>	Enregistrer toutes les disquettes modifiées
<b>Shift+F7</b>	Enregistrer toutes les disquettes modifiées sous un nouveau nom.
<b>F8</b>	Basculer entre mode fenêtre / mode plein écran
<b>F9</b>	Sauver le fichier tape
<b>F10</b>	Démarrer / arrêter la capture vidéo AVI
<b>F11</b>	Copier l'écran texte dans le presse-papiers.
<b>F12</b>	Coller le presse-papiers dans l'écran texte.
<b>Help</b>	Aide (pour Amiga, MorphOS and AROS)
<b>AltGr</b>	Modificateur supplémentaire

L'intérêt du couple F11/F12 mérite d'être souligné, car il offre une passerelle intéressante entre Oricutron et le système hôte.

## Le débogueur / moniteur d'Oricutron

Pour accéder au mode débogueur / moniteur d'Oricutron, vous devez presser la touche F2 (F2 est en fait une bascule entre le débogueur et l'émulateur qui peut être utilisée à tout moment sans perte, ni dans l'émulateur, ni dans le débogueur / moniteur). Un écran apparaît, structuré comme celui de la Fig. 1, page suivante : Cet écran est composé de 4 fenêtres :

- 1) Ecran de l'Oric
- 2) Status du 6502 ("6502 Status")
- 3) Une zone dans laquelle alternent les 3 fenêtres suivantes : Moniteur ("Monitor") / Débogueur ("Debug console") / Dump mémoire ("Memory watch"). On passe d'une fenêtre à l'autre avec la touche F3.
- 4) Une zone dans laquelle alternent les 2 fenêtres suivantes : Informations VIA ("VIA Status") et Informations AY ("AY Status"). On passe d'une fenêtre à l'autre grâce à la touche F4.

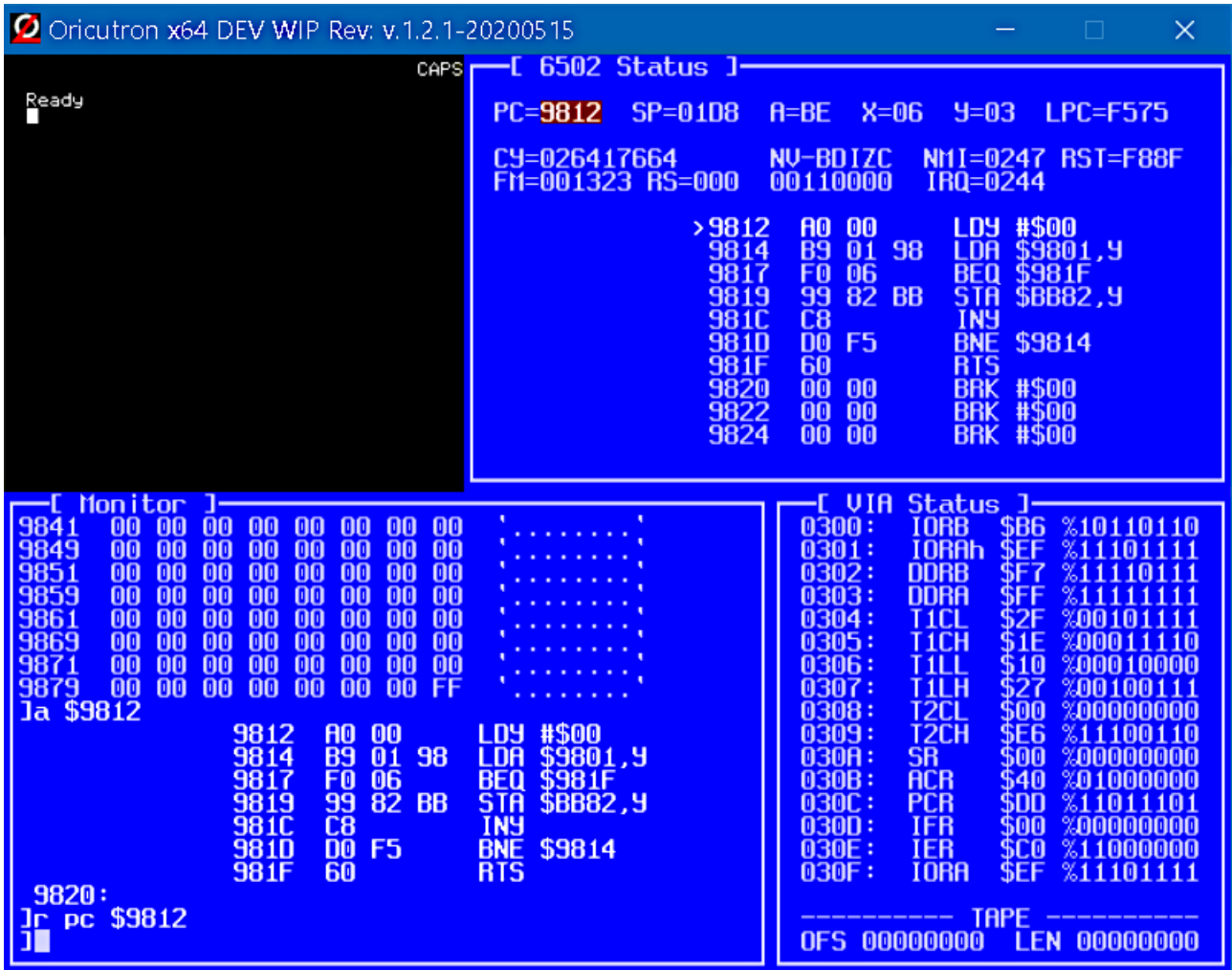


Fig.1 Etat du débogueur après assemblage de code LM (a \$9812) et positionnement du PC (r pc \$9812)

## Touches actives dans le mode Débogueur / Moniteur

Dans ce mode, outre l'écran de l'Oric, les fenêtres suivantes sont visibles :

- "6502 Status" (Fig. 1 en haut à droite).
- "Monitor" (Fig. 1 en bas à gauche) ou "Debug console" (Fig. 2) ou "Memory watch" (Fig. 3).
- "VIA Status" (Fig. 1 en bas à droite) ou "AY Status" (Fig. 5).

Les touches suivantes sont actives de manière générale, sans qu'il soit nécessaire d'indiquer une fenêtre en particulier.

- F1** Aller au menu principal (retour au mode *émulateur* par ESC ou Backspace ou BACK avec la souris, mais F2 permet de retrouver le Débogueur / Moniteur au même point).
- F2** Passer dans le mode émulateur (retour dans le Débogueur / Moniteur par la touche F2).
- F3** Pour basculer entre les fenêtres Moniteur ("Monitor") / Débogueur ("Debug console") / Dump mémoire ("Memory watch").
- F4** Pour basculer entre les fenêtres Informations VIA ("VIA Status") et Informations AY ("AY Status").
- F9** Pour réinitialiser le compteur de cycles (ligne **CY=** dans la fenêtre 6502) (Fig. 4).
- F10** Pour exécuter la ligne de code marquée > dans la fenêtre 6502 (Fig. 1 en haut à droite).
- F11** Comme F10, mais s'il s'agit d'un sous-programme, celui-ci est exécuté en bloc, sans affichage du détail des lignes de ce sous-programme.
- F12** Pour sauter l'instruction marquée > et pointer sur l'instruction suivante.

La paire d'écrans "VIA Status" ou "AY Status" affichent des informations en fonction du contexte et ne disposent pas de commandes spécifiques. Par contre, chacune des fenêtres "Monitor", "Memory watch" et "Debug console", possède des commandes propres, en plus des commandes listées ci-dessus.

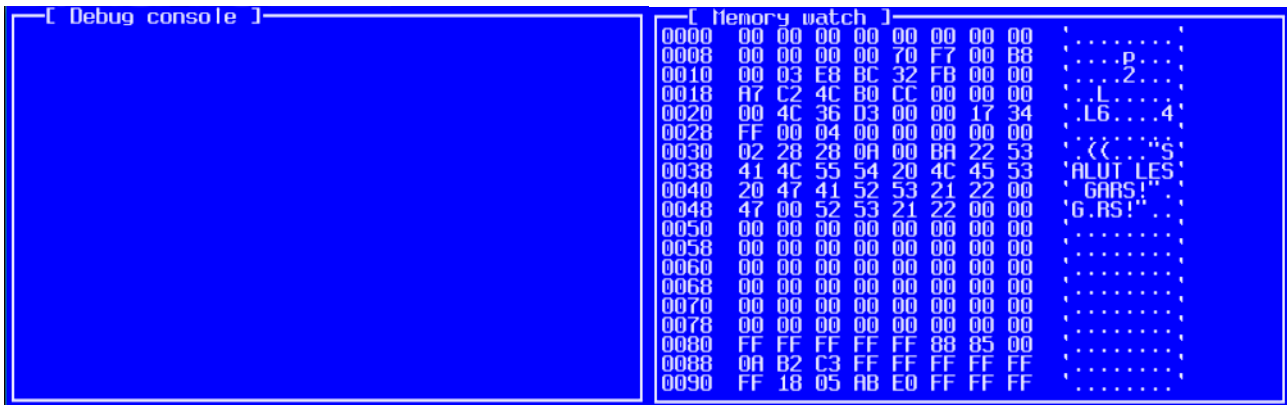


Fig. 2 et 3 Fenêtres “Debug console” (vide pour l’instant) et “Memory watch” (ici on voit la page zéro)

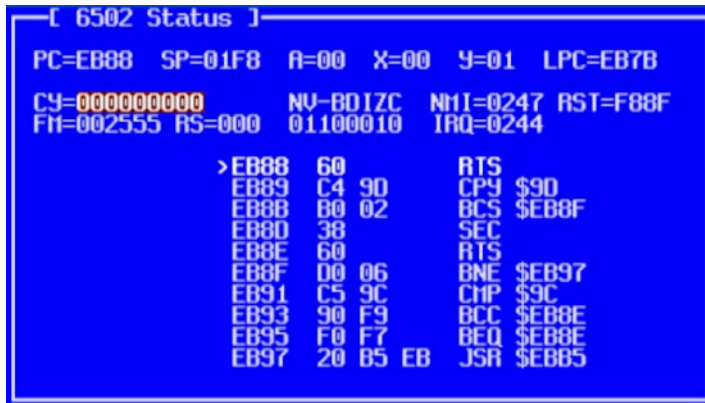


Fig. 4 et 5 : Fenêtres “6502 status” (après réinitialisation de CY par F9) et “AY status”

## Touches actives dans le mode Console (fenêtre "Debug console")

Flèches Haut / Bas Historique des commandes.

## Touches actives dans le mode Dump Mémoire (fenêtre "Memory watch")

Flèches Haut / Bas Scrolling Haut / Bas, ligne par ligne.

Shift + Flèches Haut / Bas Scrolling Haut / Bas, écran par écran.

Page Up/Page Down Idem.

Chiffres hexadécimaux Pour changer de zone mémoire à visualiser. Exemple \$9800 affiche le contenu de la mémoire à partir de \$9800 (Fig. 6 ci-dessous).

**S** Activer / désactiver le mode fractionné (Split). Dans ce mode, deux demi-fenêtres “Memory watch” permettent de visualiser deux zones différentes de la mémoire (Fig. 6).

**Tab** Changer de demi-fenêtre “Memory watch” en mode fractionné. La demi-fenêtre active est indiquée par une coche dans la marge gauche. Les commandes ne sont alors effectives que sur cette demi-fenêtre (Fig. 6).

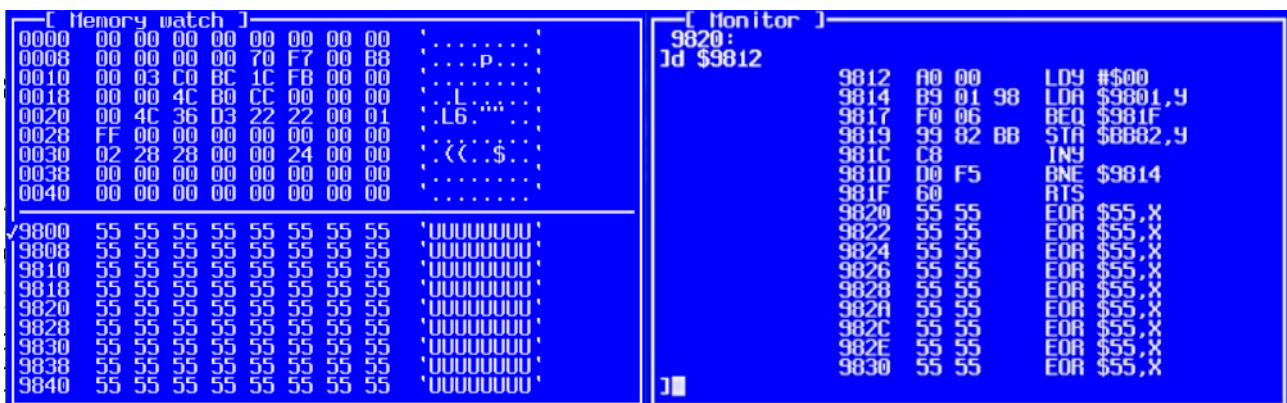


Fig. 6 et 7 : Fenêtres “Memory watch” (Splitée par S) et “Monitor” (désassemblage par d \$9812)

## Instructions en mode moniteur (fenêtre "Monitor")

En entrée les valeurs numériques sont décimales par défaut. Elles peuvent être hexadécimales si elles sont préfixées par \$ ou binaires si elles sont préfixées par %. Presque toutes les sorties sont en mode hexadécimal. Là où vous pouvez entrer un nombre ou une adresse, vous pouvez transmettre un registre CPU ou VIA. (Les registres VIA sont préfixés par V, par exemple VDDRA). Là où vous pouvez entrer une adresse, vous pouvez également utiliser un symbole. En plus des commandes décrites au paragraphe "Touches actives dans le mode Débogueur / Moniteur", voici les commandes spécifiques du mode Moniteur :

<b>?</b>	Affiche toutes les commandes disponibles (RETURN pour avoir l'écran suivant).
<b>Flèches Haut / Bas</b>	Historique des commandes.
<b>a &lt;addr&gt;</b>	Assembler le code à l'adresse <addr> (assemble) (Fig. 1). Au prompt entrer une ligne de code. L'assembleur passe à la ligne suivante, presser RETURN pour finir.
<b>bc &lt;bp id&gt;</b>	Effacer le point d'arrêt indiqué (clear breakpoint).
<b>bcm &lt;bp id&gt;</b>	Effacer le point d'arrêt mémoire indiqué (clear memory breakpoint).
<b>bl</b>	Lister les points d'arrêt (list breakpoints).
<b>blm</b>	Lister les points d'arrêt mémoire (list memory breakpoints).
<b>bs &lt;addr&gt; [zc]</b>	Définir un point d'arrêt classique (set breakpoint) (pour les modificateurs <b>zc</b> , voir plus loin dans la partie consacrée aux points d'arrêts).
<b>bsm &lt;addr&gt; [rwc]</b>	Définir un point d'arrêt mémoire (set memory breakpoint) (pour les modificateurs <b>rwc</b> , voir plus loin dans la partie consacrée aux points d'arrêts).
<b>bz</b>	Supprimer les points d'arrêt (zap breakpoints).
<b>bzm</b>	Supprimer les points d'arrêt mémoire (zap memory breakpoints).
<b>d &lt;addr&gt;</b>	Désassembler à partir de l'adresse <addr> (disassemble) (Fig. 7).
<b>fd &lt;addr&gt; &lt;end&gt; &lt;file&gt;</b>	Désassembler dans un fichier, de <addr> à <end> (disassemble to file). Ce fichier <file> est sauvé dans la racine du répertoire Oricutron. Aucune extension, ni entête, n'est ajoutée. Il ne contient que les data bruts, sans informations complémentaires (adresses, type etc.). NB. L'ancienne commande <b>df &lt;addr&gt; &lt;end&gt; &lt;file&gt;</b> n'est plus reconnue.
<b>fw &lt;addr&gt; &lt;len&gt; &lt;file&gt;</b>	Sauver une zone mémoire sur la disquette (bin file write) (Fig. 10). Ce fichier <file> est sauvé dans la racine du répertoire Oricutron. Aucune extension, ni entête, n'est ajoutée. Il ne contient que les data bruts, sans informations complémentaires (adresses, type etc.). NB. L'ancienne commande <b>wm (write memory to disk)</b> n'est plus reconnue.
<b>fr &lt;addr&gt; &lt;file&gt;</b>	Charge un fichier en mémoire à l'adresse <addr> (bin file read) (Fig. 10). Il s'agit d'un fichier brut préalablement sauvé dans la racine du répertoire Oricutron (ou d'un fichier similaire reconstruit). Il faut donc préciser l'adresse cible et le nom exact.
<b>m &lt;addr&gt;</b>	Dumper les 128 octets présents en mémoire à partir de l'adresse <addr> tout en restant dans la fenêtre Moniteur (memory) (Fig. 9). Presser la touche RETURN pour afficher la demi-page suivante. Cette commande est moins puissante que la commande <b>mw</b> ci-après.
<b>mm &lt;addr&gt; &lt;valeur&gt;</b>	Modifier la mémoire (modify memory) (Fig. 8).
<b>mw &lt;addr&gt;</b>	Afficher la fenêtre Dump Mémoire et dumper à partir de l'adresse <addr> (memory watch at addr) (Fig. 3). Cette commande est plus puissante que la commande <b>m</b> ci-dessus (Fig. 9) (voir les nombreuses touches actives dans ce mode Dump Mémoire).
<b>nl &lt;fichier&gt;</b>	Charger une copie d'écran (load snapshot) (pas d'extension par défaut).
<b>ns &lt;file&gt;</b>	Enregistrer une copie d'écran (save snapshot) (pas d'extension par défaut).
<b>r &lt;reg&gt; &lt;val&gt;</b>	Initialiser le registre <reg> avec la valeur <val> (set register). Exemple "r pc \$9812" positionne le Program Counter en \$9812 (Fig. 1, fenêtre 6502 status en bas à gauche). Les touches F10 et F11 permettent d'exécuter le code qui se trouve à cette adresse.
<b>q, x ou qm</b>	Quitter le moniteur et retourner à l'émulateur (quit, exit, quit moniteur)
<b>qe</b>	Quitter l'émulateur (quit emulator)
<b>sa &lt;nom&gt; &lt;addr&gt;</b>	Ajouter ou déplacer un symbole utilisateur (add symbol)
<b>sk &lt;nom&gt;</b>	Supprimer un symbole utilisateur (kill symbol)
<b>sc</b>	Symboles non sensibles à la casse (symbols not case-sensitive)
<b>sC</b>	Symboles sensibles à la casse (symbols Case-sensitive)
<b>sl &lt;file&gt;</b>	Charger les symboles utilisateur (symbols load) (pas d'extension par défaut)
<b>sx &lt;file&gt;</b>	Exporter les symboles utilisateur (export symbols) (pas d'extension par défaut)
<b>sz</b>	Supprimer les symboles utilisateur (zap symbols)

```

[ Monitor ]
Jmm $9801 $53
Jmm $9802 $61
Jmm $9803 $6c
Jmm $9804 $75
Jmm $9805 $74
Jmm $9806 $20
Jmm $9807 $6c
Jmm $9808 $65
Jmm $9809 $73
Jmm $980a $20
Jmm $980b $67
Jmm $980c $61
Jmm $980d $72
Jmm $980e $73
Jmm $980f $20
Jmm $9810 $21
Jmm $9811 $00
J

[ Monitor ]
Jmm $9811 $00
Jm $9801
9801 53 61 6C 75 74 20 6C 65 'Salut le'
9809 73 20 67 61 72 73 20 21 's gars !'
9811 00 00 00 00 00 00 00 00
9819 00 00 00 00 00 00 00 00
9821 00 00 00 00 00 00 00 00
9829 00 00 00 00 00 00 00 00
9831 00 00 00 00 00 00 00 00
9839 00 00 00 00 00 00 00 00
9841 00 00 00 00 00 00 00 00
9849 00 00 00 00 00 00 00 00
9851 00 00 00 00 00 00 00 00
9859 00 00 00 00 00 00 00 00
9861 00 00 00 00 00 00 00 00
9869 00 00 00 00 00 00 00 00
9871 00 00 00 00 00 00 00 00
9879 00 00 00 00 00 00 00 FF
J

```

Fig. 8 et 9 : Fenêtres "Monitor" (*mm modify memory*) et (*m dump memory*)

```

[ Monitor ]
Jfw $9801 $9820 salut.fm
Written 38944 bytes to 'salut.fm'
Jwm $9801 $9820 salut.wm
???
Jmm $9801 $FF
Jr pc $9801
Jfr $9801 salut.fm
38944 bytes read from 'salut.fm'
J

```

```

Salut les gars !
Ready
█
CAPS

```

Fig. 10 et 11 : Fenêtres "Monitor" (commandes *fw* et *fr*) et "Ecran Oric" (résultat de l'exécution)

## Points d'arrêt

Parmi les commandes listées ci-dessus, 8 sont consacrées aux points d'arrêt. C'est dire l'importance qu'ils occupent dans le débogueur / moniteur d'Oricutron. Voici quelques précisions. Il existe deux types de points d'arrêt. Les points d'arrêt normaux ou "classiques" se déclenchent lorsque le processeur est sur le point d'exécuter l'instruction à l'adresse du point d'arrêt. Les points d'arrêt "mémoire" se déclenchent lors de l'accès ou de la modification de l'adresse du point d'arrêt.

**Les points d'arrêt "classiques"** peuvent utiliser les modificateurs *z* et / ou *c*.

**bs \$0C00** Arrêt lorsque le processeur est sur le point d'exécuter du code à l'adresse \$0C00

**bs \$0C00 z** Idem et met le compteur de cycles à zéro

**bs \$0C00 zc** Met le compteur de cycles à zéro et continue (pas d'arrêt)

**bs \$0C00 c** Continue l'exécution (le point d'arrêt est désactivé)

L'objectif principal des modificateurs *z* et / ou *c* est de faciliter le comptage des cycles.

Si des symboles sont chargés, ils peuvent être utilisés à la place des adresses absolues.

**Les points d'arrêt "mémoire"** peuvent être déclenchés de trois manières: lorsque le processeur est sur le point de lire (*r* pour *read*), ou d'écrire (*w* pour *write*) à l'adresse ou lorsque la valeur à l'adresse change pour une raison quelconque (*c* pour *change*). Le choix du modificateur *c* (*c* pour *change*) n'est pas très heureux, car pour les points d'arrêt normaux, *c* est déjà utilisé pour *continue*. Voici quelques exemples d'utilisation :

**bsm \$0C00 r** Arrêt lorsque le processeur est sur le point de lire à l'adresse \$0C00

**bsm \$0C00 rw** Arrêt lorsque le processeur est sur le point de lire ou d'écrire à l'adresse \$0C00

**bsm \$0C00 c** Déclenche une pause après le changement du contenu de \$0C00

**bsm \$0C00 rwc** Déclenche une pause juste avant que le processeur ne lise ou n'écrive à l'adresse \$0C00 ou juste après le changement, pour une raison quelconque, du contenu de \$0C00.

## Conclusion

Ce manuel est certes très succinct et de nombreuses explications pourraient encore être ajoutées. Néanmoins, j'espère qu'il vous sera utile, comme il l'est pour moi. N'hésitez pas à me faire connaître les manques et les erreurs, afin que je puisse l'améliorer.