

# RETROFORGE 64

Manuel complet du logiciel

VERSION TEST PUBLIQUE

Version 0.1.0 - build 41

IDE autonome pour assembleur 6502/6510, BASIC natif, projets multi-fichiers, graphismes et supports Commodore 64 / VIC-20

Réalisé par ChatGPT  
Piloté et dirigé par Loic Lété  
Août 2026

# Sommaire

1. Statut de la version test publique
2. Installation et premier lancement
3. Interface et thèmes
4. Éditeur de texte
5. Gestion des projets
6. Assembleur 6502/6510
7. Directives et syntaxe de type TMP
8. BASIC natif compilé
9. Profils mémoire C64 et VIC-20
10. Atelier graphique
11. Disquette 1541 et D64
12. Cassette, T64 et cartouches CRT
13. Éditeur hexadécimal
14. Émulateur et débogueur
15. Exportation et headers
16. Diagnostics et dépannage
17. Raccourcis et aide-mémoire
18. Limites connues

# 1. Statut de la version test publique

RetroForge 64 v0.1.0 build 41 est une version test publique. Elle regroupe dans un seul fichier HTML autonome un éditeur, un assembleur, un compilateur BASIC natif, des outils de projet et plusieurs ateliers Commodore.

Le logiciel fonctionne hors ligne et n'envoie pas les sources vers un serveur. Les projets sont conservés dans le stockage du navigateur et peuvent être exportés sur le disque.

- Conserver régulièrement une archive complète du projet.
- Tester les fichiers PRG, D64, T64 ou CRT dans un émulateur reconnu avant utilisation sur matériel réel.
- Les fonctions d'émulation et certains formats complexes restent expérimentaux.
- Lors d'un rapport de bug, joindre le numéro build 41 et le message de la zone Sortie.

## 2. Installation et premier lancement

### Utilisation hors ligne

- Ouvrir le fichier HTML dans un navigateur moderne.
- Dans Fichier, choisir Télécharger l'IDE HTML ou Enregistrer l'IDE sous pour conserver une copie autonome.
- Le titre de l'onglet doit afficher : RetroForge 64 - Test public - v0.1.0 build 41.
- Autoriser l'accès aux fichiers ou aux dossiers uniquement lorsque le navigateur le demande.

### Premier programme assembleur

```
* = $0801
start:
    lda #$06
    sta $d020
    rts
```

- Sélectionner C64.
- Créer ou ouvrir un fichier .asm.
- Appuyer sur F5.
- Consulter les erreurs, symboles et informations mémoire.
- Exporter en PRG pour conserver l'adresse de chargement.

### Premier programme BASIC natif

```
CLS
CENTRE "RETROFORGE"
LOCATE 5,5
PRINT "TEST PUBLIC"
END
```

Choisir le mode BASIC dans l'éditeur, sélectionner la machine, puis utiliser F5. Le compilateur traduit le BASIC vers l'assembleur 6502 avant de produire le code machine.

## 3. Interface et thèmes

| Zone             | Rôle                                                                          |
|------------------|-------------------------------------------------------------------------------|
| Barre supérieure | Fichier, ASM, BASIC, exportation, graphismes, hexadécimal, émulateur et aide. |

| Zone                | Rôle                                                       |
|---------------------|------------------------------------------------------------|
| Projet à gauche     | Dossiers, fichiers, filtre, création et déplacement.       |
| Centre              | Onglets, barre d'édition, source et sortie de compilation. |
| Inspecteur à droite | Symboles, mémoire et rapport du programme.                 |
| Barre d'état        | Position du curseur, sélection, version et build.          |

Les thèmes modifient le fond, les panneaux, les textes et les couleurs syntaxiques. Après un changement, vérifier la lisibilité des commentaires, chaînes, nombres, labels et directives.

## 4. Éditeur de texte

| Fonction             | Utilisation                                                                     |
|----------------------|---------------------------------------------------------------------------------|
| Coloration ASM/BASIC | Le langage suit l'extension .asm/.s/.inc ou .bas/.nbas et le sélecteur de mode. |
| Tab / Maj+Tab        | Indente ou désindente toutes les lignes sélectionnées.                          |
| Complétion           | Ctrl+Espace affiche instructions, directives, symboles et commandes BASIC.      |
| Recherche locale     | Ctrl+F cherche et remplace dans le fichier actif.                               |
| Recherche projet     | Parcourt tous les fichiers texte du projet.                                     |
| Aller à la ligne     | Ctrl+G place le curseur à la ligne demandée.                                    |
| Symboles             | Accès aux labels, macros, procédures et fonctions.                              |
| Navigation           | Trouver les références et renommer un symbole dans le projet.                   |
| État modifié         | Un point orange indique les changements non sauvegardés sur le disque.          |
| Curseur persistant   | Position, sélection et défilement sont mémorisés par onglet.                    |
| Snippets             | Bibliothèque séparée pour ASM et BASIC.                                         |

## Édition productive

- Ctrl+/ commente ou décommente la sélection.
- Alt+Flèche haut/bas déplace une ligne.
- Dupliquer et supprimer la ligne sont disponibles dans le menu Édition.
- Ctrl+Maj+P ouvre la palette de commandes.
- La sortie conserve une hauteur limitée et devient défilable lorsqu'elle contient plusieurs diagnostics.

## 5. Gestion des projets

Un projet contient des sources ASM/BASIC, des inclusions, des ressources binaires et les données des ateliers. Les chemins internes utilisent /, même sous Windows.

### Arborescence

- Clic sur un dossier : le sélectionner ; clic sur la flèche : le replier.
- Clic droit dossier : fichier ASM, sous-dossier, renommer ou supprimer.
- Clic droit fichier : dupliquer, déplacer, renommer ou supprimer.
- Glisser un fichier sur un dossier pour le déplacer.
- Un doublon de nom dans le dossier cible bloque le déplacement.

## Sauvegardes

| Commande                 | Résultat                                                                 |
|--------------------------|--------------------------------------------------------------------------|
| Ctrl+S / Sauver actif    | Met à jour le fichier lié au disque ou propose un téléchargement.        |
| Ctrl+Maj+S / Sauver tout | Écrit tous les fichiers dans le dossier autorisé ou produit une archive. |
| Projet complet RFP JSON  | Conserve sources, dossiers, réglages ASM, graphismes, disquette et ROM.  |
| ZIP du projet            | Conserve l'arborescence sous forme d'archive transportable.              |
| Fermer le projet         | Ferme la session après contrôle des modifications.                       |

Le stockage du navigateur est une sauvegarde de session, pas une sauvegarde de travail définitive.

## 6. Assembleur 6502/6510

### Modes d'adressage

| Mode            | Exemple      | Octets |
|-----------------|--------------|--------|
| Implicite       | CLC          | 1      |
| Accumulateur    | ASL A        | 1      |
| Immédiat        | LDA #\$20    | 2      |
| Page zéro       | LDA \$FB     | 2      |
| Page zéro,X/Y   | LDA \$FB,X   | 2      |
| Absolu          | LDA \$C000   | 3      |
| Absolu,X/Y      | STA \$0400,X | 3      |
| Indirect        | JMP (\$FFFC) | 3      |
| Indexé indirect | LDA (\$FB,X) | 2      |
| Indirect indexé | LDA (\$FB),Y | 2      |
| Relatif         | BNE boucle   | 2      |

## Instructions officielles

| Famille                | Instructions                           |
|------------------------|----------------------------------------|
| Chargement et stockage | LDA, LDX, LDY, STA, STX, STY           |
| Transferts             | TAX, TAY, TXA, TYA, TSX, TXS           |
| Pile                   | PHA, PHP, PLA, PLP                     |
| Arithmétique           | ADC, SBC, INC, INX, INY, DEC, DEX, DEY |
| Logique                | AND, ORA, EOR, BIT                     |
| Décalages              | ASL, LSR, ROL, ROR                     |
| Comparaisons           | CMP, CPX, CPY                          |
| Branches               | BCC, BCS, BEQ, BMI, BNE, BPL, BVC, BVS |
| Sauts                  | JMP, JSR, RTS, RTI                     |
| Drapeaux               | CLC, CLD, CLI, CLV, SEC, SED, SEI      |
| Système                | BRK, NOP                               |

## Expressions

- Hexadécimal : \$D020 ; binaire : %11000000 ; décimal : 49152.
- Octet faible : <adresse ; octet fort : >adresse.
- Opérateurs : +, -, \*, /, %, <<, >>, &, |, ^ et parenthèses.
- Constante : mon\_etiquette = \$C000.
- Les labels locaux commençant par un point sont rattachés au dernier label global.

## 7. Directives et syntaxe de type TMP

| Directive            | Description                               | Exemple                        |
|----------------------|-------------------------------------------|--------------------------------|
| * = / .org           | Définit l'origine.                        | * = \$C000                     |
| .byte / .db / !byte  | Octets, chaînes ou expressions.           | .byte 1,\$ff,<label            |
| .word / .dw / !word  | Mots little-endian.                       | .word start                    |
| .text / .pet         | Texte brut ou PETSCII simplifié.          | .text "OK"                     |
| .hex                 | Suite d'octets hexadécimaux.              | .hex A9 00 60                  |
| .binary              | Groupes de huit bits.                     | .binary 11000000               |
| .fill / .res         | Réserve et initialise.                    | .fill 256,0                    |
| .align               | Aligne le compteur programme.             | .align 256                     |
| .include             | Inclut un source, chemin relatif accepté. | .include "lib/video.asm"       |
| .incbin              | Inclut un binaire avec offset/taille.     | .incbin "gfx.bin", \$20, \$100 |
| .macro / .endmacro   | Macro paramétrée.                         | .macro SETCOLOR value          |
| .repeat / .endrepeat | Répète un bloc.                           | .repeat 8                      |
| .assert              | Échoue si l'expression vaut zéro.         | .assert end-start<256          |
| .proc / .scope       | Organisation symbolique.                  | .proc update                   |
| .banksz / .bank      | Placement de banques ROM.                 | .bank 1,\$8000                 |

## Macro paramétrée

```
.macro SET_BORDER value
    lda #\value
    sta $d020
.endmacro

SET_BORDER $06
```

## Includes

Un chemin relatif est d'abord recherché depuis le dossier du fichier appelant. Les cycles d'inclusion sont signalés avec la chaîne complète. INCBIN accepte une ressource du projet et la convertit en octets lors du prétraitement.

# 8. BASIC natif compilé

Le BASIC natif ne dépend pas de l'interpréteur BASIC du Commodore. Le source est transformé en assembleur, puis assemblé en programme machine exportable en PRG ou BIN.

## Fondamentaux

| Commande         | Description                         | Exemple             |
|------------------|-------------------------------------|---------------------|
| LET / =          | Affectation.                        | A=10                |
| CONST / CONSTANT | Constante de compilation.           | CONST SCREEN=\$0400 |
| DIM              | Variable typée ou tableau statique. | DIM SCORE AS UINT16 |
| END / STOP       | Fin du programme.                   | END                 |
| REM / ' / //     | Commentaires.                       | REM INIT            |

## Texte

| Commande                             | Description                   | Exemple                 |
|--------------------------------------|-------------------------------|-------------------------|
| PRINT                                | Texte ou valeur.              | PRINT "SCORE ";SCORE    |
| CLS / TEXT CLEAR                     | Efface l'écran.               | CLS                     |
| HOME                                 | Origine du curseur.           | HOME                    |
| LOCATE                               | Place le curseur.             | LOCATE 5,10             |
| CENTRE / CENTER                      | Centre un texte.              | CENTRE "TITRE"          |
| TEXT AT / CHAR / HLINE / VLINE / BOX | Primitives d'interface texte. | TEXT BOX 1,1,38,20,81,1 |
| GET / WAIT INPUT                     | Lecture clavier.              | WAIT INPUT K            |
| INPUT                                | Saisie numérique ou chaîne.   | INPUT "NOM";NOM\$       |

## Flux

| Commande                    | Description           | Exemple           |
|-----------------------------|-----------------------|-------------------|
| IF / ELSEIF / ELSE / END IF | Condition structurée. | IF A=1 THEN       |
| SELECT CASE                 | Sélection multiple.   | SELECT CASE A     |
| FOR / NEXT / STEP           | Boucle numérique.     | FOR I=0 TO 10     |
| EXIT FOR                    | Sortie anticipée.     | EXIT FOR          |
| WHILE / WEND                | Test avant boucle.    | WHILE A<10        |
| DO / LOOP                   | Boucle générale.      | LOOP UNTIL K<>0   |
| REPEAT / UNTIL              | Test après boucle.    | UNTIL A=10        |
| GOTO / GOSUB / RETURN       | Branchements.         | GOSUB draw        |
| ON ... GOTO                 | Branchement indexé.   | ON A GOTO un,deux |

## Procédures

| Commande      | Description               | Exemple                |
|---------------|---------------------------|------------------------|
| PROCEDURE     | Sous-programme paramétré. | PROCEDURE couleur(C)   |
| FUNCTION      | Fonction avec résultat.   | FUNCTION double(V)     |
| CALL          | Appel de procédure.       | CALL couleur(6)        |
| CALL ... INTO | Récupère le résultat.     | CALL double(12) INTO R |

## Mathématiques

| Commande                    | Description               | Exemple                |
|-----------------------------|---------------------------|------------------------|
| RND / RANDOM                | Aléatoire 8 bits.         | MATH RANDOM INTO R     |
| ABS / MIN / MAX / CLAMP     | Valeur et bornes.         | MATH CLAMP X,0,39      |
| ADD / SUB / MUL / DIV / MOD | Calcul entier.            | MATH DIV A,10 INTO R   |
| AND / OR / XOR              | Logique bit à bit.        | MATH AND A,\$0F INTO R |
| INC / DEC                   | Incrément ou décrétement. | INC X                  |

## Mémoire et données

| Commande              | Description                              | Exemple                         |
|-----------------------|------------------------------------------|---------------------------------|
| DATA / READ / RESTORE | Tables intégrées.                        | DATA valeurs: 1,2,3             |
| DATA FILL / RANGE     | Génération de données.                   | DATA RANGE palette,0,15         |
| PEEK / POKE           | Accès mémoire 8 bits.                    | A=PEEK(\$D012)                  |
| PEEK16/32 / POKE16/32 | Accès multi-octets.                      | POKE32 \$C100,123456            |
| MEM FILL / COPY       | Traitement de zones.                     | MEM FILL \$C000,256,0           |
| INCLUDE / INCBIN      | Sources et ressources.                   | INCBIN "gfx/data.bin",gfx_data  |
| RLE                   | Compression par couples quantité,valeur. | RLE COMPRESS image TO image_rle |

## Jeu et graphismes

| Commande                   | Description                    | Exemple                    |
|----------------------------|--------------------------------|----------------------------|
| BORDER / BACKGROUND        | Couleurs principales.          | BACKGROUND 0               |
| GRAPHICS                   | Modes texte et bitmap.         | GRAPHICS MULTICOLOR BITMAP |
| SET/GET_SCREEN             | Accès écran par coordonnées.   | SET_SCREEN 2,3,81          |
| SET/GET_COLOR_RAM          | Accès couleur par coordonnées. | SET_COLOR_RAM 2,3,7        |
| GAME TILE / SCREEN / COLOR | Gestion de tuiles.             | GAME TILE 10,10,65,1       |
| GAME JOY / KEY / RASTER    | Entrées et synchronisation.    | GAME JOY FIRE 2 INTO F     |
| GAME SCORE / LIVES         | Variables de jeu.              | GAME SCORE ADD 10 TO SCORE |

## Sprites C64

| Commande                          | Description                 | Exemple                   |
|-----------------------------------|-----------------------------|---------------------------|
| SPRITE ENABLE / DISABLE           | Visibilité.                 | SPRITE ENABLE 0           |
| SPRITE POSITION / COLOR / POINTER | Configuration.              | SPRITE POSITION 0,160,100 |
| SPRITE MULTICOLOR / MULTICOLOR1/2 | Palette multicolore.        | SPRITE MULTICOLOR 0,ON    |
| SPRITE X/Y INTO                   | Lecture de position.        | SPRITE X 0 INTO X         |
| SPRITE COLLISION                  | Collisions sprite et décor. | SPRITE COLLISION INTO HIT |

## Son et disque

| Commande          | Description                         | Exemple                        |
|-------------------|-------------------------------------|--------------------------------|
| SOUND / SOUND OFF | Voix SID simplifiée.                | SOUND 1,4400,33,15             |
| SID INIT / PLAY   | Appel d'un pilote SID chargé.       | SID PLAY \$C003                |
| DISK DEVICE       | Périphérique 4 à 30.                | DISK DEVICE 8                  |
| DISK LOAD / SAVE  | PRG, SEQ, USR et REL selon support. | DISK LOAD "LEVEL",\$C000,8,PRG |
| DISK STATUS INTO  | Statut KERNAL.                      | DISK STATUS INTO ST            |

# 9. Profils mémoire C64 et VIC-20

| Machine / profil      | Départ BASIC conseillé | Écran typique       | Remarques                                    |
|-----------------------|------------------------|---------------------|----------------------------------------------|
| C64                   | \$0801                 | \$0400              | Color RAM \$D800, VIC-II \$D000, SID \$D400. |
| VIC-20 sans extension | \$1001                 | \$1E00              | Mémoire réduite, configuration VIC \$9000.   |
| VIC-20 +3 Kio         | \$0401                 | Selon configuration | Le lanceur doit correspondre à l'extension.  |
| VIC-20 +8 Kio ou plus | \$1201                 | Selon configuration | Réserver les zones vidéo et variables.       |

Le module Headers BASIC peut adapter automatiquement le profil à la machine et à l'extension VIC-20 sélectionnées. L'aperçu PRG/BIN affiche les deux octets du header, le début, la fin et la taille.

## 10. Atelier graphique

| Atelier           | Fonctions principales                                                                      |
|-------------------|--------------------------------------------------------------------------------------------|
| Charset / tileset | Planche de 256 caractères, sélection directe, édition 8x8, hires et multicolore.           |
| Screen / map      | Placement de tiles, dimensions de carte, Color RAM par case et navigation dans le tileset. |
| Sprites           | Sprites 24x21, modes hires/multicolore, couleurs partagées et planche.                     |
| Bitmap            | 320x200 hires ou 160x200 logique multicolore, couleurs par cellule 8x8.                    |
| Imports           | PNG, JPEG et WebP convertis vers les couleurs actives.                                     |
| Exports           | BIN, ASM, données RLE et pack bitmap/screen/color.                                         |

### Contraintes multicolores C64

- Un caractère multicolore utilise quatre valeurs de pixels : fond partagé, deux couleurs partagées et couleur propre à la case.
- La Color RAM d'une case est limitée aux quatre bits bas.
- Un sprite multicolore emploie transparent, multicolore 1, couleur individuelle et multicolore 2.
- Le bitmap multicolore utilise deux bits par pixel logique et double la largeur visuelle.

### RLE RetroForge

Le format produit des couples [quantité, valeur]. Chaque couple décompresse la valeur autant de fois que la quantité. Une séquence de 5 octets \$20 devient \$05,\$20. La quantité maximale d'un couple est 255.

## 11. Disquette 1541 et D64

- Ajouter le dernier assemblage ou importer des fichiers externes.
- Choisir individuellement le type et la présence du header PRG.
- Paramétrer l'adresse de chargement de chaque fichier.
- Ajouter un lanceur BASIC SYS C64 ou VIC-20.
- Créer une image D64 avec BAM et répertoire.
- Importer et inspecter une image D64 existante.

### Les deux octets du header

Dans un PRG, les deux premiers octets contiennent l'adresse de chargement en little-endian : octet faible puis octet fort. Pour \$0801, le fichier commence par \$01,\$08. Cette option se règle fichier par fichier dans le module disquette.

| Type | Usage                                          | État                  |
|------|------------------------------------------------|-----------------------|
| PRG  | Programme avec adresse de chargement.          | Pris en charge        |
| SEQ  | Flux séquentiel.                               | Import/export de base |
| USR  | Fichier utilisateur.                           | Import/export de base |
| REL  | Enregistrements relatifs et secteurs latéraux. | Partiel               |

| Type | Usage                    | État                   |
|------|--------------------------|------------------------|
| D64  | Image 1541 de 35 pistes. | Création et inspection |

## 12. Cassette, T64 et cartouches CRT

| Format             | Description                                | Précaution                                    |
|--------------------|--------------------------------------------|-----------------------------------------------|
| BIN / ROM          | Octets bruts sans adresse.                 | Conserver séparément l'origine.               |
| T64                | Conteneur cassette avec entrée et adresse. | Tester le répertoire et le chargement.        |
| CRT 8/16 Kio       | Cartouche générique C64.                   | Code généralement placé à \$8000.             |
| Ocean 128/512 Kio  | Cartouche à banques.                       | Vérifier numéros et taille des CHIP.          |
| Magic Desk 512 Kio | 64 banques de 8 Kio.                       | La logique de commutation reste à programmer. |
| EasyFlash 1 Mio    | Conteneur bancaire étendu.                 | Support partiel.                              |
| Ultimax            | ROML et ROMH.                              | Respecter la cartographie Ultimax.            |
| VIC-20 CRT         | Blocs adaptés aux adresses VIC-20.         | Choisir l'extension mémoire correcte.         |

Les directives .banksize et .bank organisent le placement. Elles ne remplacent pas le code de commutation propre au matériel de la cartouche.

## 13. Éditeur hexadécimal

- Ouvrir un BIN, PRG, ROM, CRT ou le dernier assemblage.
- Modifier les octets tout en consultant leur représentation ASCII.
- Définir une adresse de base pour afficher les adresses mémoire.
- Chercher et remplacer une séquence à taille constante.
- Exporter une plage en BIN ou sous forme de directives ASM.
- Calculer somme 16 bits et XOR 8 bits.
- Réinjecter la ressource modifiée dans le projet.

## 14. Émulateur et débogueur

| Outil           | Description                                                                      |
|-----------------|----------------------------------------------------------------------------------|
| CPU 6502/6510   | Pas-à-pas, exécution bornée, registres, pile et drapeaux.                        |
| Points d'arrêt  | Adresse du PC ou condition sur une cellule mémoire.                              |
| Mémoire         | Inspection, surveillance et recherche de séquences.                              |
| Écran texte     | Prévisualisation C64 ou VIC-20 depuis la RAM.                                    |
| Bitmap VIC-II   | Prévisualisation hires ou multicolore.                                           |
| Sprites VIC-II  | Affichage des huit sprites et couleurs.                                          |
| ROM utilisateur | Chargement légal des ROM BASIC, KERNAL et caractères fournies par l'utilisateur. |

L'émulateur n'est pas encore cycle-exact. VIC-II, SID, CIA, interruptions complexes, timing raster et périphériques ne remplacent pas un émulateur de validation tel que VICE.

## 15. Exportation et headers

| Sortie | Header                               | Usage                               |
|--------|--------------------------------------|-------------------------------------|
| PRG    | 2 octets d'adresse little-endian     | Chargement Commodore standard.      |
| BIN    | Aucun                                | Insertion à une adresse connue.     |
| D64    | Structure BAM/répertoire et secteurs | Disquette virtuelle 1541.           |
| T64    | Header archive et entrées            | Cassette virtuelle.                 |
| CRT    | Header global + paquets CHIP         | Cartouches C64/VIC-20.              |
| ASM    | Labels et directives de données      | Réutilisation dans un autre source. |

### Headers BASIC

- Mode automatique : suit C64/VIC-20 et l'extension mémoire.
- Adresse personnalisée : utile pour une organisation mémoire précise.
- PRG ON : ajoute les deux octets de chargement.
- PRG OFF : produit un BIN brut.
- Aperçu : contrôle profil, début, fin, taille et octets du header avant export.

## 16. Diagnostics et dépannage

| Symptôme                                | Action recommandée                                                              |
|-----------------------------------------|---------------------------------------------------------------------------------|
| Un menu ne s'ouvre plus                 | Fermer les autres menus, recharger la build 41 et lancer le diagnostic interne. |
| Une erreur JavaScript apparaît          | Copier le message complet et préciser l'action effectuée.                       |
| F5 n'assemble pas                       | Vérifier le mode ASM/BASIC, le fichier d'entrée et l'audit ASM.                 |
| INCLUDE introuvable                     | Contrôler le chemin relatif au fichier appelant et la casse du nom.             |
| INCBIN introuvable                      | Ajouter la ressource au projet et contrôler offset/taille.                      |
| Fichier dans le mauvais dossier         | Utiliser clic droit Déplacer vers ou glisser-déposer.                           |
| Projet perdu après nettoyage navigateur | Restaurer le ZIP ou le projet complet RFP JSON.                                 |
| PRG ne démarre pas                      | Vérifier origine, header et adresse SYS.                                        |
| D64 non lisible                         | Inspecter BAM/répertoire puis valider dans un émulateur externe.                |

## 17. Raccourcis et aide-mémoire

| Raccourci   | Commande                     |
|-------------|------------------------------|
| F5          | Compiler / assembler         |
| Ctrl+S      | Sauvegarder le fichier actif |
| Ctrl+Maj+S  | Sauvegarder tout le projet   |
| Ctrl+F      | Rechercher / remplacer       |
| Ctrl+G      | Aller à une ligne            |
| Ctrl+Espace | Complétion                   |

| Raccourci           | Commande                |
|---------------------|-------------------------|
| Ctrl+/'             | Commenter / décommenter |
| Alt+Flèche haut/bas | Déplacer la ligne       |
| Ctrl+Maj+P          | Palette de commandes    |
| Tab / Maj+Tab       | Indenter / désindenter  |

## 18. Limites connues

- Cette édition est une version test publique, pas une version de production garantie.
- L'émulation VIC-II/SID/CIA n'est pas cycle-exacte.
- Le format REL complet et ses secteurs latéraux restent partiels.
- Certaines cartouches bancarisées exigent un pilote de commutation écrit par le programmeur.
- Le compilateur BASIC doit être validé commande par commande pour les programmes importants.
- Les accès persistants au disque dépendent des fonctions autorisées par le navigateur.
- Le stockage local peut être supprimé par le navigateur ; conserver des exports externes.

Fin du manuel - RetroForge 64 v0.1.0 build 41 - version test publique.