

# Graph35\*

Paquet permettant de tracer des touches et menus  
de calculatrices CASIO.

Louis Paternault  
spalax+ctan(at)gresille(dot)org

18 avril 2018

## Résumé

Ce paquet fournit les commandes pour tracer des touches et menus de certaines calculatrices CASIO (parmi lesquelles les GRAPH25, GRAPH35, GRAPH75 et d'autres...).

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	License . . . . .	2
1.2	Sommaire . . . . .	3
<b>2</b>	<b>Téléchargement et installation</b>	<b>3</b>
2.1	Distributions GNU/Linux . . . . .	3
2.2	Distribution L <sup>A</sup> T <sub>E</sub> X . . . . .	3
2.3	Installation manuelle . . . . .	3
<b>3</b>	<b>Utilisation</b>	<b>3</b>
3.1	Calculatrices prises en charge . . . . .	3
3.2	Options du paquet . . . . .	4
3.3	Couleurs . . . . .	4
3.4	Calculatrices . . . . .	5
3.5	Touches . . . . .	6
3.6	Écran . . . . .	7
3.7	Changement d'échelle . . . . .	8
<b>4</b>	<b>Binares</b>	<b>9</b>

---

\*Ce document décrit le paquet `graph35` v0.1.1, publié le 2018/04/18. Site web, tickets de suivi, etc. sur <http://framagit.org/spalax/graph35>.

<b>A</b>	<b>Calculatrices</b>	<b>9</b>
<b>B</b>	<b>Ancre</b>	<b>9</b>
	B.1 Ancres des touches . . . . .	10
	B.2 Ancres de la touche <b>REPLAY</b> . . . . .	10
	B.3 Ancres de l'écran . . . . .	10
	B.4 Ancres du boîtier . . . . .	10
<b>C</b>	<b>Pixel art</b>	<b>10</b>
	C.1 Menu . . . . .	10
	C.2 Fonctions . . . . .	14
	C.3 Batterie . . . . .	26
<b>D</b>	<b>Touches</b>	<b>26</b>
<b>E</b>	<b>Implementation</b>	<b>26</b>
	Liste des figures	45
	Historique des changements	46
	Index	46

## 1 Introduction

Ce document présente le paquet `graph35`, permettant de tracer des boutons et menus de certaines calculatrices CASIO.

### 1.1 License

This work may be distributed and/or modified under the conditions of the  $\LaTeX$  Project Public License, either version 1.3 of this license or (at your option) any later version.

Further information can be found in the `.dtx` file used to build this document.

Traduction (sans valeur légale)

L'ensemble de ce travail peut être publié et/ou modifié en respectant les conditions de la  *$\LaTeX$  Project Public License* (License publique du projet  $\LaTeX$ ), au choix dans la version 1.3 de cette licence, ou une licence ultérieure.

## 1.2 Sommaire

L'installation est décrite dans la partie 2. Les options et macros sont décrites dans la partie 3. La partie 4 décrit quelques logiciels utilisés ou associés à ce paquet. Les annexes A à D contiennent la liste des calculatrices, touches, menus disponibles, ainsi que des illustrations des différentes macros mises en œuvre dans ce paquet. Enfin, la dernière partie E contient le code du paquet.

## 2 Téléchargement et installation

### 2.1 Distributions Gnu/Linux

Sur les distributions pour lesquelles `graph35` a été empaqueté, la manière la plus simple de l'installer est d'utiliser votre gestionnaire de paquet. Sous Debian (et ses dérivées, comme Ubuntu), il est empaqueté dans le paquet `texlive-pictures` à partir de la version 2018.20180404-1. Il est donc possible de l'installer en utilisant la commande suivante.

```
sudo apt install texlive-pictures
```

### 2.2 Distribution L<sup>A</sup>T<sub>E</sub>X

Ce paquet est disponible à la fois dans les distributions T<sub>E</sub>XLive et MiK<sub>T</sub>E<sub>X</sub>. Il peut être installé en utilisant leur gestionnaire de paquets respectif.

### 2.3 Installation manuelle

— Télécharger l'archive.

**Version stable** <http://mirrors.ctan.org/graphics/graph35.zip>

**Version de développement** <https://framagit.org/spalax/graph35/repository/archive.zip?ref=master>

— Décompresser l'archive.

— Compiler le fichier : `latex graph35.ins`

— Déplacer les fichiers `.sty` dans un répertoire du chemin de L<sup>A</sup>T<sub>E</sub>X.

## 3 Utilisation

### 3.1 Calculatrices prises en charge

**Touches et calculatrices** Les macros permettent de dessiner la calculatrice et les touches de la GRAPH35 uniquement.

**Écran** Les macros permettent de dessiner les éléments de menus affichés à l'écran pour, entre autres, les GRAPH25, GRAPH35, GRAPH75, FX-9860GII, FX-9750GII et sans doute d'autres.

## 3.2 Options du paquet

Le paquet admet une unique option `color`, qui vaut par défaut `color=real`.


Cette option peut prendre deux valeurs : `real` et `blackandwhite`, qui définit la couleur par défaut des dessins (calculatrice et touches). Voir la section suivante pour les détails.


Ce n'est pas à proprement parler une option du paquet, mais pour accélérer la compilation, il est possible d'ajouter la ligne `\PassOptionsToPackage{draft}{pixelart}` avant de charger le paquet `graph35` : ceci a pour effet de désactiver l'affichage de toutes les images en pixelart (principalement les macros `\function`, voir partie C.2). En effet, la compilation de ces commandes peut être très long, et la désactivation temporaire peut faire gagner du temps<sup>1</sup>.

## 3.3 Couleurs

### 3.3.1 Couleurs prédéfinies

Il est possible de personnaliser les couleurs des touches, de la calculatrice, etc. en utilisant des jeux de couleur prédéfinies (ou personnalisées ; voir plus loin). Ces jeux de couleurs prédéfinis sont :

**real**  Couleur réaliste, mais qui risque d'être difficilement lisible dans un document imprimé en noir et blanc.

**blackandwhite**  Noir et blanc, à fort contraste, qui sera plus lisible à l'impression.

### 3.3.2 Choix des couleurs

Plusieurs méthodes permettent de choisir parmi un jeu de couleur prédéfinies.

- L'argument `color` du paquet définit la couleur par défaut à utiliser (qui peut être ensuite modifiée au cas par cas avec l'option `color` des macros). Par exemple, pour que tous les dessins soient en noir et blanc, charger le paquet en utilisant `\usepackage[color=blackandwhite]{graph35}`. Par défaut, les couleurs réalistes sont utilisées (`color=real`).
- Les commandes `\key` et `\calculator` acceptent en plus une valeur `color` pour définir la couleur de cette commande uniquement. Par défaut, la couleur définie lors du chargement du paquet est utilisée. Ces commandes acceptent un plus une valeur `color=default`, pour spécifier explicitement l'utilisation de la couleur par défaut.
- Enfin, il est possible de redéfinir la couleur par défaut en utilisant la macro `\setgraphcolor{<color>}`. Par exemple, si le paquet a été chargé avec l'option `color=blackandwhite`, pour utiliser les couleurs réalistes pour la suite du document, utiliser `\setgraphcolor{real}`.


`\setgraphcolor`

---


1. Par exemple, sur mon ordinateur, ajouter cette option à ce fichier rend la compilation trente fois plus rapide, la faisant passer de huit minutes à 16 secondes.

### 3.3.3 Couleurs personnalisées


Il est aussi possible d'utiliser des couleurs arbitraires en définissant les couleurs suivantes.

`graph35ACON` : Touche ACON .

`graph35ACONBORDER` : Bordure de la touche ACON.

`graph35ALPHA` : Touche ALPHA .

`graph35ALPHABORDER` : Bordure de la touche ALPHA.

`graph35SHIFT` : Touche SHIFT .


`graph35SHIFTBORDER` : Bordure de la touche SHIFT.

`graph35SCREEN` : Pixels de l'écran.

`graph35SCREENBG` : Arrière-plan de l'écran.

`graph35CASE` : Boîtier.

`graph35CASEBORDER` : Bordure du boîtier.

`graph35EXE` : Touche EXE .

`graph35EXEBORDER` : Bordure de la touche EXE.


`graph35NUMBER` : Touches numériques.

`graph35NUMBERBORDER` : Bordures des touches numériques.

`graph35KEYTEXT` : Texte sur les touches.

`graph35ALPHATEXT` : Texte *alpha* au dessus des touches.

`graph35SHIFTEXT` : Texte *shift* au dessus des touches.

Ces couleurs sont des noms de couleur au sens du paquet `xcolor`, et peuvent être redéfinies en utilisant les commandes de ce paquet. Par exemple, pour produire la touche , utiliser le code suivant.

```
1 \colorlet{graph35KEYTEXT}{green}
2 \colorlet{graph35SHIFTEXT}{orange}
3 \definecolor{graph35ALPHATEXT}{RGB}{0, 0, 255}
4 \definecolor{graph35NUMBER}{RGB}{200, 200, 200}
5 \colorlet{graph35NUMBERBORDER}{graph35NUMBER}
6
7 \key[shift, alpha]{7}
```

## 3.4 Calculatrices

`\calculator` Pour le moment, il n'est possible d'afficher qu'un seul modèle de calculatrice : la GRAPH35+. La syntaxe est `\calculator[color, scale]{modèle}`.

- `{modèle}` Voir la liste des modèles disponibles dans l'annexe A (page 9).
- `[color]` Permet de changer la couleur du dessin ; voir la partie précédente (3.3).

- [*scale*] Permet de modifier l'échelle du dessin. Le résultat produit n'est pas forcément celui que vous attendiez ; voir la partie 3.7 pour plus d'informations.

Par exemple, la commande `\calculator[color=real]{graph35+E}` produit une version dix fois plus grande du dessin suivant (une version plus grande est visible dans l'annexe A, page 9).



`\tikzcalculator` Il est aussi possible d'inclure une calculatrice dans une figure TikZ, avec la commande `\tikzcalculator{modèle}`. Cette commande ne prend aucun autre argument que le modèle, et trace une calculatrice autour des coordonnées (0;0). Pour dessiner ailleurs, avec une autre échelle, utilisez un environnement `scope`, comme dans l'exemple suivant.

```

1 \begin{tikzpicture}
2   \begin{scope}[shift={(1, 2)}, scale=.5]
3     \tikzcalculator{graph35+E}
4   \end{scope}
5 \end{tikzpicture}



```

Des ancres (*anchors*) sont définies pour chacune des touches, les bords de la calculatrice, ainsi que l'écran, pour pouvoir y faire référence dans vos tracés TikZ. Voir la partie B pour plus d'informations.





### 3.5 Touches

`\key` Pour inclure une touche de la calculatrice, utiliser :

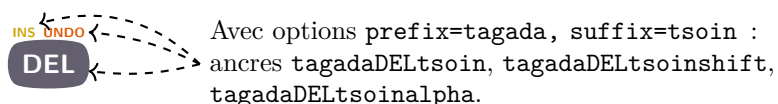
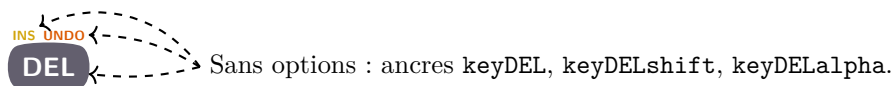
`\key[color, prefix, suffix, scale, shift, alpha]{touche}`.

Par exemple `\key[color=blackandwhite]{DEL}` produira  tandis que `\key[shift, alpha]{DEL}` produira .

Les arguments sont :

- `{touche}` Nom de la touche à afficher (par exemple 1 pour , ou EXE pour ). Le nom de la touche est plus ou moins ce qui est écrit dessus. La liste des touches est visible sous forme de liste dans l'annexe D, ou comme légende d'une calculatrice dans la figure 6.
- [*color, scale*] Couleur et échelle de la touche. Ces arguments prennent les mêmes arguments et ont les mêmes limitations que pour la commande `calculator` (voir section 3.3 pour les couleurs, et 3.7 pour l'échelle).
- [*shift, alpha*] Ces options affichent ou masquent les textes jaunes et rouges décrivant la fonction de la touche si elle a été pressée après les touches  ou . Par défaut, ces textes sont masqués (ce qui est équivalent à `shift=false, alpha=false`); pour l'activer, utiliser `shift=true` ou `alpha=true` ou plus simplement `shift` ou `alpha`.

- [*prefix, suffix*] Des ancres (*anchors*) sont créées avec chaque touche, pour permettre d’y faire référence dans les tracés TikZ (elles sont utilisées par exemple pour tracer la figure 6). Par défaut, ces ancres ont pour nom `key` suivi du nom de la touche (par exemple `keyDEL` pour la touche DEL). Ces options permettent de définir le nom de l’ancre, comme dans la figure suivante. Cela permet d’avoir plusieurs fois la même touche sur la même figure, et de faire référence aux deux touches de manière distincte. Ces options contrôlent aussi le nom des ancres définies pour les textes SHIFT et ALPHA.



Ces ancres sont illustrées dans les annexes B.1 et B.2.

- En regardant le code source, vous pouvez voir que d’autres options sont disponibles. Elles ne sont pas décrites ici car elles ne sont pas destinées à être utilisées par l’utilisateur final, et peuvent changer dans une prochaine version sans préavis.

`\tikzkey` Comme pour `\calculator` et `\tikzcalculator`, la macro `\tikzkey` a la même fonction que `\key`, sauf qu’elle est prévue pour être appelée à l’intérieur d’environnement TikZ. Sa signature est :

$$\text{\tikzkey}[\langle options \rangle]{\langle touche \rangle}{\langle coordonnées \rangle}$$


Ses arguments sont :

- [*options*] les mêmes options que pour la commande `\key` ;
- $\langle touche \rangle$  le nom de la touche à dessiner ;
- $\langle coordonnées \rangle$  les coordonnées autour desquelles dessiner la touche.

## 3.6 Écran

Des macros permettent de tracer des éléments de menu qui apparaissent à l’écran. Il y en a trois : pour tracer les éléments du menu, les légendes des touches de fonction, et le niveau des batteries.

### 3.6.1 Menus

`\menu` La macro `\menu{\langle icône \rangle}{\langle raccourci \rangle}` affiche l’icône du menu de la calculatrice. Par exemple, `\menu{RUNMAT}{A}` affiche . Le raccourci (le caractère apparaissant en bas à droite de l’icône) est indépendant de l’icône car selon le modèle et la mise à jour de la calculatrice, il peut varier.

La liste des icônes et raccourcis disponibles est disponible dans l’annexe C.1.

`\tikzmenu` La macro `\tikzmenu`, dont la signature est `\tikzmenu[options]{icône}{raccourci}{coordonnées}` dessine une icône du menu dans un environnement TikZ. Ses arguments sont :

- `{icône}` et `{raccourci}` : mêmes arguments que pour `\menu` ;
- `{coordonnées}` : coordonnées du coin supérieur gauche du menu ;
- `[options]` : options, qui sont passées telles quelles à la macro `\bwpixelart` du paquet `pixelart` ; elles permettent de régler l'échelle et la couleur du tracé (par exemple `scale=.5`, `color=red`).

### 3.6.2 Fonctions

`\function` La macro `\function{fonction}` affiche la légende des touches de fonctions F1 à F6 (par exemple `\function{F1}` ou `\function{F6}`). La liste des dessins disponibles se trouve à l'annexe C.2.

`\tikzfunction` La macro `\tikzfunction[options]{fonction}{coordonnées}` a la même utilité que la macro `\function`, mais dans un environnement TikZ. Son argument `{fonction}` est le même que pour `\function` ; ses arguments `[options]` et `{coordonnées}` sont les mêmes que pour `\tikzmenu`.

### 3.6.3 Batterie

`\battery` La macro `\battery{charge}` affiche le niveau de la batterie (par exemple `\battery{100}`). La liste des dessins disponibles se trouve à l'annexe C.3.

`\tikzbattery` La macro `\tikzbattery[options]{charge}{coordonnées}` a la même utilité que la macro `\battery`, mais dans un environnement TikZ. Son argument `{charge}` est le même que pour `\battery` ; ses arguments `[options]` et `{coordonnées}` sont les mêmes que pour `\tikzmenu`.

## 3.7 Changement d'échelle

L'option `scale` utilisée pour modifier la taille des calculatrices et des touches ne modifie pas l'épaisseur des traits ni le rayon des coins arrondis. Ceci a pour effet indésirable le tracé suivant d'une calculatrice à l'échelle  $1/10$  : remarquez que le bord du cadre (en vert) est trop épais, et l'écran est quasiment elliptique.



Pour remédier à cela, plusieurs solutions existent, dont aucune n'est idéale, sans quoi elle serait mise en œuvre par défaut.

- S'accomoder de ces défauts. En effets, pour des petites mises à l'échelle, ils sont peu visibles.
- Enrober le tracé dans une commande `\scalebox` ou `\resizebox`, comme le code `\resizebox{.1}{\calculator{graph35+E}}` qui produit le tracé suivant.





- Utiliser l’option `transform canvas` de `pgf` (lors de l’appel à `TikZ`, comme par exemple `\begin{tikzpicture}[scale=.1, transform canvas={scale=.1}]...` Ceci réduit correctement l’ensemble du tracé, mais ne change pas les limites dudit tracé, et ne modifie pas les coordonnées en accord (donc les ancres deviennent inutilisables).

Enfin, si vous incluez des tracés dans un environnement `tikzpicture` avec l’option `scale`, n’oubliez pas d’ajouter l’option `transform shape` pour que les contours de l’image soient aussi modifiés, et que votre petite image ne se retrouve pas perdue au milieu d’un grand espace vide.

## 4 Binaires

Quelques logiciels (écrits en Python3) sont maintenus en même temps que ce paquet `LATEX`, afin d’aider à son développement. Ils ne sont par contre pas distribués avec ce paquet, et doivent être téléchargés sur la forge logicielle pour être utilisés. Ils sont assez spécialisés pour accompagner le développement de ce paquet, mais si quelqu’un leur trouve une autre utilité, tant mieux.

La plupart de ces logiciels manipulent des fichiers au format `.pxl`, qui est un format créé pour l’occasion, codant une image en *pixel art*. Chaque icône de menu, ou fonction, est enregistrée dans un tel fichier avant d’être convertie en code `LATEX` et intégrée à ce paquet.

- `catpxl` Affiche dans le terminal, de manière lisible, un fichier `.pxl`.
- `completefunctionchars` Si ce n’est pas déjà fait, associe à chaque *pixel art* des touches de fonction la liste des caractères apparaissant dessus (ceci est utile pour ensuite indexer ces *pixel art* dans l’annexe C.2).
- `generate.keys` et `generate.pixelart` À partir de différents fichiers du dépôt, génère le code `LATEX` distribué sous la forme de ce paquet, et une partie de la documentation.
- `screenshot2pixelart` Extrait d’une capture d’écran de la calculatrice ses *pixel art*.

## A Calculatrices

Liste des calculatrices représentées, ainsi que leur mot-clef (utilisé comme argument des macros `\calculator` et `\tikzcalculator`).

- `graph35+E` : figure 1.

## B Ancres

Illustration des différentes ancres définies sur les différents tracés.

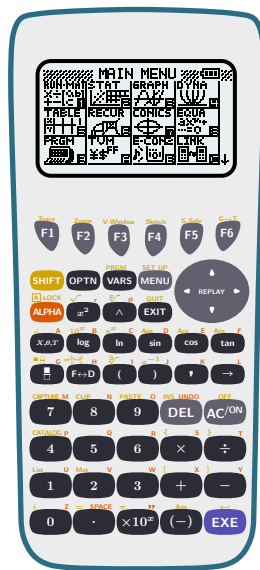


FIGURE 1 – Calculatrice graph35+E.

## B.1 Ancres des touches

Chaque touche définit les ancres illustrées à la figure 2.

## B.2 Ancres de la touche REPLAY

La touche REPLAY définit des ancres supplémentaires, pour chacune des flèches. Elles sont illustrées figure 3.

## B.3 Ancres de l'écran

Il est possible de faire référence à l'écran en utilisant les ancres de la figure 4.

## B.4 Ancres du boîtier

Il est possible de faire référence au boîtier en utilisant les ancres de la figure 5.

# C Pixel art

## C.1 Menu

En plus des icônes et raccourcis disponibles sur les calculatrices, deux valeurs spéciales sont disponibles : **black**, qui produit une figure de même taille, mais noire ; et **blank**, qui ne produit aucune figure.

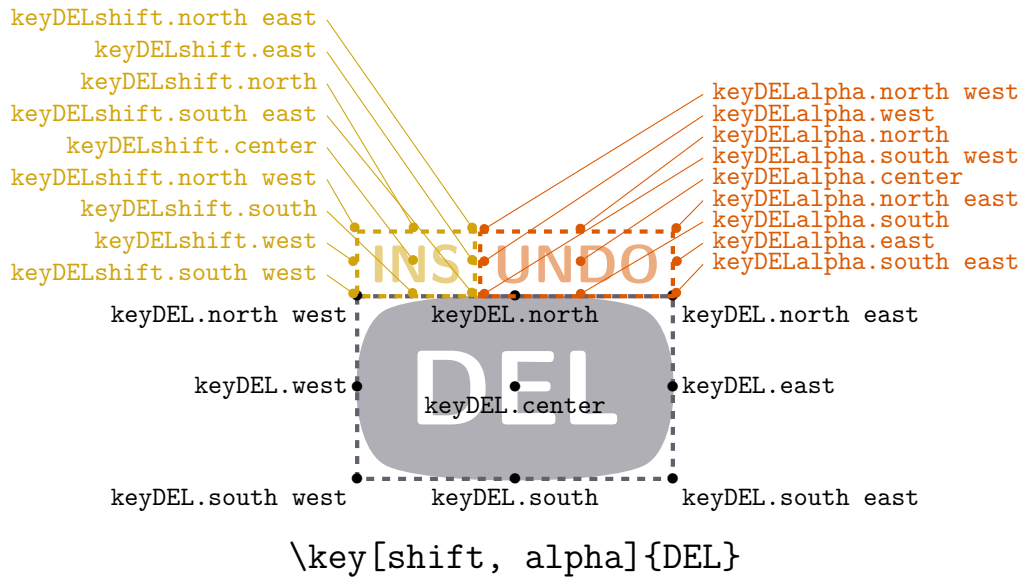


FIGURE 2 – Ancres des touches

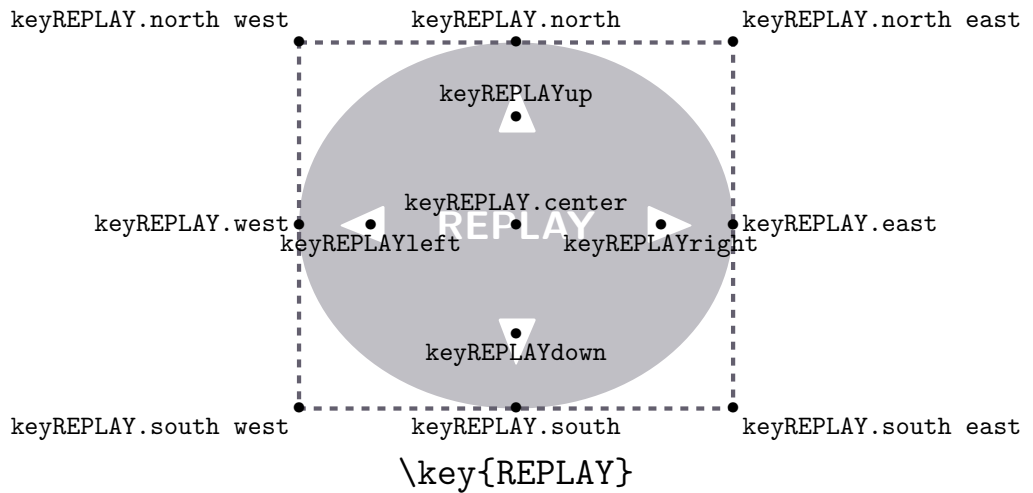


FIGURE 3 – Ancres de la touche REPLAY

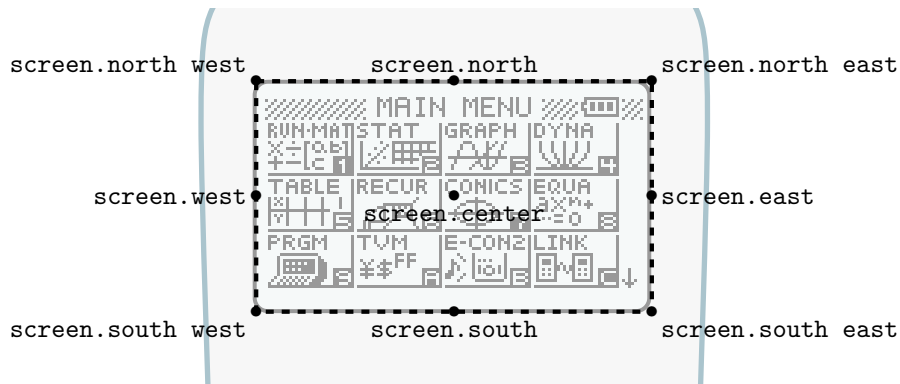





































FIGURE 4 – Ancres de l'écran

### C.1.1 Icônes

—  \menu{black}{black}	—  \menu{MEMORY}{black}
—  \menu{blank}{black}	—  \menu{PRGM}{black}
—  \menu{CONICS}{black}	—  \menu{RECUR}{black}
—  \menu{DYNA}{black}	—  \menu{RUN}{black}
—  \menu{eACT}{black}	—  \menu{RUNMAT}{black}
—  \menu{ECON2}{black}	—  \menu{SSHT}{black}
—  \menu{eCON3}{black}	—  \menu{STAT}{black}
—  \menu{EQUA}{black}	—  \menu{SYSTEM}{black}
—  \menu{GEOM}{black}	—  \menu{TABLE}{black}
—  \menu{GRAPH}{black}	—  \menu{TVM}{black}
—  \menu{LINK}{black}	

### C.1.2 Raccourcis

—  \menu{black}{1}	—  \menu{black}{8}
—  \menu{black}{2}	—  \menu{black}{9}
—  \menu{black}{3}	—  \menu{black}{A}
—  \menu{black}{4}	—  \menu{black}{B}
—  \menu{black}{5}	—  \menu{black}{black}
—  \menu{black}{6}	—  \menu{black}{blank}
—  \menu{black}{7}	—  \menu{black}{C}

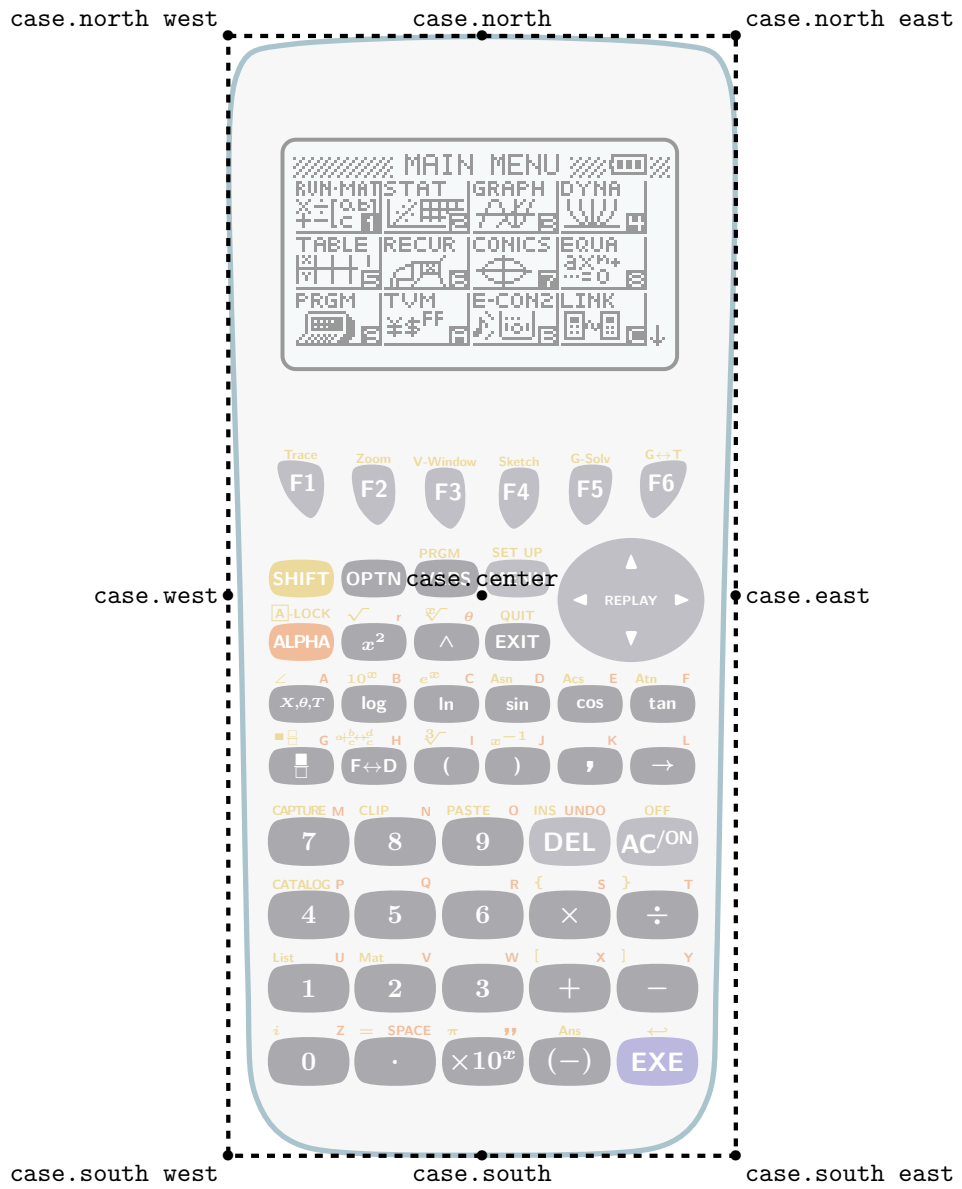







FIGURE 5 – Ancres du boîtier





























































—  \menu{black}{D}	—  \menu{black}{G}
—  \menu{black}{E}	—  \menu{black}{H}
—  \menu{black}{F}	

## C.2 Fonctions

Les images disponibles sont triées selon les caractères visibles (lettres latines et chiffres). Pour trouver quel code produit l'image que vous désirez, regardez quels caractères sont visibles, et retrouvez votre image dans la partie correspondante de l'index.

Par exemple, sur  ou , aucun caractère n'est visible (en effet, les lettres de  sont des lettres grecques, pas latines) ; sur , les lettres *acn* sont visibles ; sur , seul la lettre *r* est visible ; etc.

### Vide

 battery	 question-b	 2-b
blank	 quote-b	200
 colon-b	 rightarrow	 200
 contrast-b	 Sigma-b	21
 degree-b	 square-b	 2x1
 Delta-b	 style1	22
 different	 style2	 2x2
 different-b	 style3	2p
 dms	 style4	 2P
 dms-b	 style5	2s
 dollar-b	 style6	 2S
 doublequote-b	 style7	2var
 doublerightarrow-b	 tilde-b	 2VAR
 equal-b	 1	 2VAR-b
 geq-b	 10	2way
 GREEK	 10	 2WAY
 greek	 100	3
 gt	 100	 3-b
 gt-b	1p	31
 key	 1P	 3x1
 leq-b	1s	33
 lt	 1S	 3x3
 lt-b	1var	38k
 micro-b	 1VAR	 to38k
next	 1VAR-b	3pin
nextb	2	 3PIN
output-b	 2	
percent-b		
period-b		

**4**  
 $\overline{4}$  4-b  
**5**  
 $\overline{5}$  5-b  
**6**  
 $\overline{6}$  6-b  
**60**  
 $\overline{60}$  60  
**7400**  
 $\overline{7400}$  7400  
**9850**  
 $\overline{9850}$  9850  
**9860**  
 $\overline{9860}$  9860  
**a**  
 $\overline{a}$  a-b  
**a0**  
 $\overline{a0}$  a0  
 $\overline{a0}$  a0-b  
**a1**  
 $\overline{a1}$  a1  
 $\overline{a1}$  a1-b  
**a2**  
 $\overline{a2}$  a2-b  
**aa**  
 $\overline{aa}$  Aa  
**ab**  
 $\overline{a-b}$  ab  
 $\overline{Sab}$  Sab  
**abc**  
 $\overline{ABC}$  ABC  
**abdf**  
 $\overline{ABdf}$  ABdf-b  
**abi**  
 $\overline{a+b}$  tcomplexalgebraic-b  
**abs**  
 $\overline{Abs}$  Abs-b  
**abt**  
 $\overline{ABT}$  ABT  
**abx**  
 $\overline{a+bx}$  aplusbx  
 $\overline{a+bx}$  aplusbx-b  
 $\overline{a \cdot bx}$  atimesbx  
 $\overline{a \cdot bx}$  atimesbx-b  
**ac**  
 $\overline{a \cdot c}$  ac  
 $\overline{a \cdot c}$  Sac  
**acn**  
 $\overline{Sacn}$  Sacn-b  
**add**  
 $\overline{ADD}$  ADD  
 $\overline{ADD}$  ADD-b  
**adf**  
 $\overline{Adf}$  Adf-b  
**adv**  
 $\overline{ADV}$  ADV-b  
**aebx**  
 $\overline{aebx}$  aebx  
 $\overline{aebx}$  aebx-b  
**all**  
 $\overline{ALL}$  ALL  
 $\overline{ALL}$  ALL-b  
**always**  
 $\overline{Always}$  Always  
**amt**  
 $\overline{AMT}$  AMT-b  
**an**  
 $\overline{an}$  an  
 $\overline{an}$  an-b  
 $\overline{San}$  San  
 $\overline{San}$  San-b  
**an1**  
 $\overline{an1}$  an1  
 $\overline{an1}$  an1-b  
 $\overline{an1}$  an1-b2  
 $\overline{San1}$  San1-b  
**an2**  
 $\overline{an2}$  an2  
 $\overline{an2}$  an2-b  
 $\overline{San2}$  San2-b  
**ancn**  
 $\overline{ancn}$  ancn-b  
**and**  
 $\overline{And}$  And-b  
**angl**  
 $\overline{ANGL}$  ANGL-b  
**anov**  
 $\overline{ANOV}$  ANOV  
**anpl**  
 $\overline{anPl}$  anPl-b  
**anst**  
 $\overline{anSt}$  anSt-b  
**apl**  
 $\overline{SaPl}$  SaPl-b  
**app**  
 $\overline{APP}$  APP-b  
**apr**  
 $\overline{APR}$  APR-b  
 $\overline{tAPR}$  tAPR  
**area**  
 $\overline{AREA}$  AREA-b  
**arg**  
 $\overline{Arg}$  Arg-b  
**as**  
 $\overline{AandS}$  AandS-b  
**asgn**  
 $\overline{ASGN}$  ASGN  
**aug**  
 $\overline{Aug}$  Aug-b  
**auto**  
 $\overline{AUTO}$  AUTO  
 $\overline{Auto}$  Auto  
 $\overline{Auto}$  Auto-2  
 $\overline{Auto}$  Auto-b  
**axb**  
 $\overline{ax+bx}$  axplusb  
 $\overline{ax+bx}$  axplusb-b  
**b**  
 $\overline{b}$  b-b  
**b0**  
 $\overline{b0}$  b0-b

b1	<b>BOT</b> BOTbottom <b>BOT</b> BOTright	ccd	<b>Ccd</b> Ccd
<b>b1</b> b1-b		cel	<b>CEL</b> CEL-b
b2	<b>BOX</b> BOX <b>Box</b> Box-b	cell	<b>CELL</b> CELL
<b>b2</b> b2-b		ch1	<b>CHI</b> CH1
bal	<b>BAL</b> BAL <b>BAL</b> BAL-b	char	<b>CHAR</b> CHAR-b
bar	<b>Bar</b> Bar-b	chg	<b>CHG</b> Chg-b
base	<b>BASE</b> BASE	chi	<b>CHI</b> CHI <b>CHI</b> CHI-b <b>Chi</b> Chi-b
bc	<b>b-c</b> bc <b>Sbc</b> Sbc	chng	<b>CHNG</b> CHNG
bcd	<b>Bcd</b> Bcd	close	<b>Close</b> Close-b
bdf	<b>Bdf</b> Bdf-b	clr	<b>CLR</b> CLR <b>CLR</b> CLR-b
bin	<b>BIN</b> BIN-b <b>Bin</b> Bin-b	cls	<b>cls</b> cls <b>Cls</b> Cls-b
binm	<b>BINM</b> BINM-b	cma	<b>CMA</b> CMA-b
bkup	<b>BKVP</b> BKVP-b	cmp	<b>Cmp</b> Cmp-b
bn	<b>bn</b> bn-b <b>Sbn</b> Sbn-b	cmpd	<b>CMPD</b> CMPD-b
bn1	<b>bn1</b> bn1-b <b>Sbn1</b> Sbn1-b	cmpr	<b>CMPR</b> CMPR-b
bn2	<b>bn2</b> bn2-b <b>Sbn2</b> Sbn2-b	cn	<b>Cn</b> cn-b <b>Scn</b> Scn-b
bnst	<b>bnSt</b> bnSt-b	cn1	<b>Cn1</b> cn1-b <b>Scn1</b> Scn1-b
bond	<b>BOND</b> BOND-b	cn2	<b>Cn2</b> cn2-b
bot			



<b>Scn2</b> Scn2-b	<b>CTGY</b> CTGY-b	<b>Det</b> Det-b
cnst	ctl	df
<b>CnSt</b> CnSt-b	<b>CTL</b> CTL-b	<b>df</b> df-b
cnt	cuml	diff
<b>cnt</b> cnt	<b>Cuml</b> Cuml-b	<b>diff</b> diff
cnvt	cut	dim
<b>CNVT</b> CNVT-b	<b>CUT</b> CUT	<b>DIM</b> DIM-b
col	cy	<b>Dim</b> Dim-b
<b>COL</b> COL	<b>CY</b> CY-b	disp
<b>COL</b> COL-b	d	<b>DISP</b> DISP-b
com	<b>d</b> d-b	dist
<b>COM</b> COM-b	d2dt2	<b>DIST</b> DIST-b
conj	<b>d2dt2</b> d2dt2	dld
<b>Conj</b> Conj-b	d2dx2	<b>dl-</b> dlminusD
conv	<b>d2dx2</b> d2dx2-b	<b>dl+</b> dlplusD
<b>CONV</b> CONV-b	data	dms
copy	<b>DATA</b> DATA-b	<b>tDMS</b> tDMS-b
<b>COPY</b> COPY	<b>Data</b> Data-b	do
<b>COPY</b> COPY-b	<b>percentDATA</b> percentDATA-b	<b>Do</b> Do-b
cosh	days	dot
<b>cosh</b> cosh-b	<b>DAYS</b> DAYS-b	<b>dot</b> dot-b
cosh1	db	draw
<b>cosh1</b> cosh1-b	<b>DB</b> DB	<b>DRAW</b> DRAW
cost	ddt	<b>DRAW</b> DRAW-b
<b>COST</b> COST	<b>ddt</b> ddt	drwc
<b>COST</b> COST-b	ddx	<b>DrwC</b> DrwC-b
<b>Cost</b> Cost-b	<b>ddx</b> ddx-b	drwf
cpd	defg	<b>DrwF</b> DrwF-b
<b>Cpd</b> Cpd	<b>DefG</b> DefG-b	drwn
cplx	del	<b>DrwN</b> DrwN-b
<b>CPLX</b> CPLX-b	<b>DEL</b> DEL	drwt
crcl	<b>DEL</b> DEL-b	<b>Drwt</b> Drwt-b
<b>Crcl</b> Crcl	dela	dsz
<b>Crcl</b> Crcl-b	<b>DELA</b> DELA-b	<b>Dsz</b> Dsz-b
crnt	dell	dx
<b>CRNT</b> CRNT-b	<b>DELL</b> DELL-b	<b>Idx</b> Idx
cstm	depr	<b>Idx</b> Idx-b
<b>CSTM</b> CSTM-b	<b>DEPR</b> DEPR-b	dyna
ctgy	det	<b>DYNA</b> DYNA-b
		<b>Dyna</b> Dyna-b
		e
		<b>e</b> e-b

<b>E</b> Exa-b	<b>ExtD</b> ExtD	<b>FP</b> FP
edf	f	<b>FP</b> FP-b
<b>Edf</b> Edf-b	<b>F</b> F	fpd
edit	<b>F</b> F-b	<b>Fpd</b> Fpd
<b>EDIT</b> EDIT	<b>F</b> F-b2	frac
<b>EDIT</b> EDIT-b	<b>f</b> femto-b	<b>Frac</b> Frac-b
eff	fa	ftbl
<b>EFF</b> EFF-b	<b>Fa</b> Fa-b	<b>FTbl</b> FTbl-b
<b>EFF</b> tEFF	fab	full
else	<b>Fab</b> Fab-b	<b>FULL</b> FULL
<b>Else</b> Else-b	fact	furie
end	<b>FACT</b> FACT-b	<b>Furie</b> Furie
<b>End</b> End-b	<b>Fact</b> Fact-b	fv
eng	fast	<b>FV</b> FV
<b>ENG</b> ENGshiftleft	<b>Fast</b> Fast	<b>FV</b> FV-b
<b>ENG</b> ENGshiftright	fb	g
engy	<b>Fb</b> Fb-b	<b>g</b> g-b
<b>ENGY</b> ENGY-b	fcd	<b>G</b> Giga-b
entr	<b>Fcd</b> Fcd	gcd
<b>ENTR</b> ENTR-b	file	<b>Gcd</b> Gcd
equa	<b>FILE</b> FILE-b	<b>GCD</b> GCD-b
<b>EQUA</b> EQUA-b	fill	gcon
es	<b>FILL</b> FILL-b	<b>GCON</b> GCON
<b>ES</b> EtS-b	<b>FILL</b> Fill-b	<b>GCON</b> Gcon-b
esym	fine	gdx
<b>ESYM</b> ESYM-b	<b>FLine</b> FLine	<b>GIdx</b> GIdx-b
exam	<b>FLine</b> FLine-b	geo
<b>EXAM</b> EXAM-b	fmax	<b>GEO</b> GEO-b
exe	<b>FMax</b> FMax-b	gmem
<b>EXE</b> EXE	fmin	<b>GMEM</b> GMEM-b
exit	<b>FMin</b> FMin-b	go
<b>EXIT</b> EXIT	for	<b>GO</b> GO
<b>EXIT</b> EXIT-b	<b>FOR</b> For-b	gof
exp	forc	<b>GOF</b> GOF
<b>Exp</b> Exp	<b>FORC</b> FORC-b	goto
<b>EXP</b> EXP-b	form	<b>GOTO</b> Goto-b
<b>EXP</b> Exp-b	<b>FORM</b> FORM	gpd
<b>EXP</b> Exp-b2	<b>FORM</b> FORM-b	<b>Gpd</b> Gpd
extd	fp	gph1

<b>GPH1</b> GPH1	<b>iden</b>	<b>InvF</b> InvF
<b>GPH1-b</b>	<b>Iden</b> Iden-b	<b>invg</b>
<b>gph2</b>	<b>iend</b>	<b>InvG</b> InvG
<b>GPHE</b> GPH2	<b>IEnd</b> IEnd-b	<b>invh</b>
<b>GPH2-b</b>	<b>if</b>	<b>InvH</b> InvH
<b>gph3</b>	<b>If</b> If-b	<b>invn</b>
<b>GPHE</b> GPH3	<b>imp</b>	<b>InvN</b> InvN
<b>GPH3-b</b>	<b>Imp</b> Imp-b	<b>invp</b>
<b>gplt</b>	<b>in</b>	<b>InvP</b> InvP
<b>GFLT</b> GPLT	<b>IN</b> IN	<b>invt</b>
<b>GPLT-b</b>	<b>init</b>	<b>InvT</b> InvT
<b>grab</b>	<b>INIT</b> INIT	<b>io</b>
<b>GRAB</b> GRAB	<b>inpt</b>	<b>IO</b> IO-b
<b>grph</b>	<b>INPT</b> INPT-b	<b>irr</b>
<b>GRPH</b> GRPH	<b>input</b>	<b>IRR</b> IRR
<b>GRPH-b</b>	<b>INPUT</b> INPUT	<b>IRR-b</b>
<b>Grph-b</b>	<b>ins</b>	<b>isct</b>
<b>gslv</b>	<b>INS</b> INS	<b>ISCT</b> ISCT
<b>GSLV-b</b>	<b>INS-b</b>	<b>isz</b>
<b>gtk</b>	<b>int</b>	<b>Isz</b> Isz-b
<b>GTK-b</b>	<b>INT</b> INT	<b>join</b>
<b>hcd</b>	<b>INT-b</b>	<b>Join</b> Join-b
<b>Hcd</b> Hcd	<b>Int</b> Int-b	<b>jump</b>
<b>help</b>	<b>Int</b> Intdiv-b	<b>JUMP</b> JUMP-b
<b>HELP</b> HELP-b	<b>SINT</b>	<b>k</b>
<b>hgeo</b>	<b>SINT-b</b>	<b>K</b> kilo-b
<b>HGEO</b> HGEO-b	<b>intg</b>	<b>lang</b>
<b>hist</b>	<b>INTG</b> INTG	<b>LANG</b> LANG-b
<b>HIST</b> Hist-b	<b>Intg</b> Intg-b	<b>lbl</b>
<b>hpd</b>	<b>intr</b>	<b>Lbl</b> Lbl-b
<b>Hpd</b> Hpd	<b>INTR</b> INTR-b	<b>lcm</b>
<b>hyp</b>	<b>inv</b>	<b>LCM</b> LCM-b
<b>HYP</b> HYP-b	<b>Inv</b> Inv	<b>lcte</b>
<b>hztl</b>	<b>Inv</b> Inv-b	<b>Lcte</b> Lcte-b
<b>HZTL</b> Hzt1	<b>invb</b>	<b>left</b>
<b>Hzt1</b> Hzt1-b	<b>InvB</b> InvB	<b>Left</b> Left-b
<b>i</b>	<b>invc</b>	<b>len</b>
<b>I</b> i-b	<b>InvC</b> InvC	<b>Len</b> Len-b
<b>I%</b> Ipercent	<b>invf</b>	
<b>I%-b</b> Ipercent-b		

**leng**  
**LENG** LENG-b  
**Leng** Leng-b  
**lgst**  
**Lgst** Lgst  
**Lgst** Lgst-b  
**line**  
**Line** Line  
**LINE** LINE-b  
**Line** Line-b  
**list**  
**List** List  
**LIST** LIST-b  
**List** List-b  
**tLIST** tLIST-b  
**lm**  
**LtoM** LtoM-b  
**lmem**  
**LMEM** LMEM-b  
**load**  
**LOAD** LOAD-b  
**log**  
**Log** Log  
**Log** Log-b  
**logab**  
**logab** logab-b  
**logic**  
**LOGIC** LOGIC-b  
**lpw**  
**LpW** LpW-b  
**lwr**  
**Lwr** Lwr-b  
**m**  
**M** Mega-b  
**m** milli-b  
**main**  
**MAIN** MAIN-b  
**man**  
**Man** Man  
**mark**  
**MARK** MARK-b  
**mass**  
**MASS** MASS-b  
**mat**  
**MAT** MAT-b  
**Mat** Mat-b  
**tMAT** tMAT-b  
**math**  
**MATH** MATH  
**Math** Math  
**MATH** MATH-b  
**max**  
**MAX** MAX  
**Max** Max-b  
**max** max-b  
**maxx**  
**maxX** maxX-b  
**maxy**  
**maxY** maxY-b  
**mean**  
**Mean** Mean-b  
**med**  
**Med** Med  
**Med** Med-b  
**mem**  
**Mem** Mem  
**MEM** MEM-b  
**memo**  
**MEMO** MEMO  
**menu**  
**MENU** MENU-b  
**Menu** Menu-b  
**mid**  
**Mid** Mid-b  
**min**  
**MIN** MIN  
**Min** Min-b  
**min** min-b  
**minx**  
**minX** minX-b  
**miny**  
**minY** minY-b  
**mkf**  
**MKF** MKF-b  
**ml**  
**MtoL** MtoL-b  
**mlti**  
**MLTI** MLTI  
**mn**  
**mxn** mxn-b  
**mod**  
**MOD** MOD-b  
**Mod** Mod-b  
**mode**  
**MODE** MODE-b  
**MODE** MODEExp-b  
**move**  
**MOVE** MOVE  
**mrg**  
**Mrg** Mrg  
**Mrg** Mrg-b  
**ms**  
**MandS** MandS-b  
**msa**  
**MSa** MSa-b  
**msab**  
**MSab** MSab-b  
**msb**  
**Msb** Msb-b  
**mse**  
**Mse** Mse-b  
**mv**  
**MV** MV  
**n**  
**n** n  
**n** n-b  
**n** nano-b  
**n1**  
**n1** n1-b  
**n2**  
**n2** n2-b  
**name**

<del>NAME</del> NAME-b	<del>Off</del> Off	<del>PBP</del> PBP-b
nan	<del>Off</del> Off-b	pcd
<del>Nan</del> Nan-b	on	<del>Pcd</del> Pcd
ncd	<del>On</del> On	pen
<del>Ncd</del> Ncd	<del>On</del> On-b	<del>PEN</del> PEN
ncr	open	pgdn
<del>nCr</del> nCr-b	<del>OPEN</del> OPEN-b	<del>PgDn</del> PgDn
ndis	<del>Open</del> Open-b	pgup
<del>NDis</del> NDis-b	opt	<del>PgUp</del> PgUp
new	<del>OPT</del> OPT	phas
<del>NEW</del> NEW-b	<del>OPT</del> OPT-b	<del>PHAS</del> PHAS
next	or	phase
<del>Next</del> Next-b	<del>Or</del> Or-b	<del>Phase</del> Phase-b
nfv	orig	pie
<del>NFV</del> NFV	<del>ORIG</del> ORIG	<del>Pie</del> Pie-b
<del>NFV</del> NFV-b	out	pitch
no	<del>OUT</del> OUT	<del>Pitch</del> Pitch-b
<del>NO</del> NO	p	pixl
none	<del>P</del> P	<del>PIXL</del> PIXL-b
<del>None</del> None	<del>P</del> p-b	plchg
<del>None</del> None-b	<del>P</del> Peta-b	<del>PlChg</del> PlChg
norm	<del>P</del> phat-b	<del>PlChg</del> PlChg-b
<del>Norm</del> Norm	<del>P</del> pico-b	ploff
<del>NORM</del> NORM-b	<del>P</del> Psnd-b	<del>Ploff</del> Ploff
<del>Norm</del> Norm-b	p1	<del>Ploff</del> Ploff-b
not	<del>Phat1</del> phat1-b	plon
<del>Not</del> Not-b	p2	<del>PlOn</del> PlOn
npd	<del>Phat2</del> phat2-b	<del>PlOn</del> PlOn-b
<del>Npd</del> Npd	pa	plot
npp	<del>Pa</del> pa-b	<del>Plot</del> Plot
<del>NPP</del> NPP-b	pab	<del>PLOT</del> PLOT-b
npr	<del>Pab</del> pab-b	<del>Plot</del> Plot-b
<del>nPr</del> nPr-b	parm	pmt
npv	<del>PARM</del> PARM	<del>PMT</del> PMT
<del>NPV</del> NPV	<del>parm</del> parm	<del>PMT</del> PMT-b
<del>NPV</del> NPV-b	<del>Parm</del> Parm-b	poisn
num	pb	<del>POISN</del> POISN-b
<del>NUM</del> NUM-b	<del>Pb</del> pb-b	pol
off	pbp	<del>POL</del> POL
	<del>PBP</del> PBP	<del>POL</del> Pol-b
		poly
		<del>POLY</del> POLY-b

ppd	<b>Q1</b> Qsnd-b	<b>RECV</b> Recv-b
<b>PPD</b> Ppd		
prc	<b>Q1</b> Q1-b	<b>REF</b> Ref-b
<b>PRC</b> PRC		
<b>PRC</b> PRC-b		
prd	<b>Q3</b> Q3-b	<b>REG</b> REG
<b>PRD</b> PRD		<b>REG</b> REG-b
<b>PRD</b> PRD-b		
pre	<b>r</b>	<b>REL</b> REL-b
<b>PRE</b> PRE	<b>r</b> r-b	
	<b>r</b> r-b2	<b>REN</b> REN-b
	<b>r</b> r-b3	
pres	<b>R=</b> requal	<b>REP</b> Rep-b
<b>PRES</b> PRES-b	<b>R=</b> requal-b	
	<b>RSND</b> Rsnd-b	
	<b>RTCP</b> tcomplexpolar-b	<b>REPT</b> REPT
prn	<b>r2</b>	
<b>PRN</b> PRN	<b>r2</b> r2-b	<b>RESLT</b> RESULT-b
<b>PRN</b> PRN-b		<b>RESLT</b> Reslt-b
<b>SPRN</b> SPRN	<b>r38k</b>	
<b>SPRN</b> SPRN-b	<b>R38K</b> R38k-b	
prob	<b>ran</b>	<b>RIGHT</b> Right-b
<b>PROB</b> PROB-b	<b>RAN</b> Ran-b	
prod	<b>rand</b>	<b>RMDR</b> Rmdr-b
<b>PROD</b> Prod-b	<b>RAND</b> RAND-b	
prog	<b>rang</b>	<b>RND</b> RND
<b>PROG</b> PROG-b	<b>RANG</b> RANG-b	<b>RND</b> Rnd-b
<b>PROG</b> Prog-b		
proj	<b>rcl</b>	<b>RNDFI</b> RndFi-b
<b>PROJ</b> Proj	<b>RCL</b> RCL	
	<b>RCL</b> RCL-b	<b>RNF</b> RNF-b
	<b>RCL</b> Rcl-b	
ptch	<b>rdel</b>	<b>ROOT</b> ROOT
<b>PTCH</b> Ptch-b	<b>RDEL</b>	
pts	<b>rec</b>	<b>ROP</b> ROP-b
<b>PTS</b> PTS-b	<b>REC</b> Rec-b	
pv	<b>recal</b>	<b>ROT</b> Rot-b
<b>PV</b> PV	<b>RECAL</b>	
<b>PV</b> PV-b		
pwr	<b>recr</b>	<b>ROW</b> ROW
<b>PWR</b> Pwr	<b>RECR</b> RECR-b	<b>ROW</b> ROW-b
<b>PWR</b> PWR-b		
<b>PWR</b> Pwr-b	<b>RECT</b>	
py	<b>recv</b>	<b>RREF</b> Rref-b
<b>PY</b> PY-b	<b>RECV</b>	
	<b>RECV</b> Recv	<b>RSET</b> RSET-b
q		

rt	<b>RT</b> RT	<b>SEQ</b> SEQ-b	<b>SOLV</b> SOLV
	<b>RTtheta</b> RTtheta-b	<b>seq</b> seq-b	<b>SOLV-b</b> SOLV-b
rtbl			<b>Solve</b> Solve
	<b>RTbl</b> RTbl-b	<b>SET</b> SET-b	<b>SolvN</b> SolvN-b
rtrn			<b>sonic</b> sonic
	<b>Rtrn</b> Rtrn-b	<b>SFV</b> SFV	<b>sp</b> sp-b
run		<b>SFV-b</b> SFV-b	<b>sqr</b> SQR
	<b>RUN</b> RUN	<b>SFV-b2</b> SFV-b2	<b>src</b> SRC
rw			<b>SRC-b</b> SRC-b
	<b>Rwplus</b> Rwplus	<b>Shift</b> Shift-b	<b>Src</b> Src-b
rx			<b>SRTA</b> SRTA
	<b>RX</b> RX-b	<b>SI</b> SI	<b>SrtA-b</b> SrtA-b
ry		<b>SI-b</b> SI-b	<b>SRTD</b> SRTD
	<b>RY</b> RY-b	<b>SIML</b> SIML-b	<b>SrtD-b</b> SrtD-b
s38k			<b>SSA</b> SSA-b
	<b>S38k</b> S38k-b	<b>SIMP</b> SIMP-b	<b>SSab</b> SSab-b
save		<b>SIMP-b2</b> SIMP-b2	<b>SSb</b> SSb-b
	<b>SAVE</b> SAVE-b	<b>SIN</b> SIN	<b>SSE</b> SSE-b
scal		<b>SIN-b</b> SIN-b	<b>STAT</b> STAT-b
	<b>scal</b> scal-b	<b>sinh</b> sinh-b	<b>Stat</b> Stat-b
scat		<b>sinh1</b> sinh1-b	<b>STD</b> STD
	<b>Scat</b> Scat-b	<b>SIZE</b> SIZE-b	<b>step</b> Step-b
sd		<b>SKTCH</b> SKTCH-b	<b>stick</b> STICK-b
	<b>SD</b> SD-b	<b>SL</b> SL	
sdev		<b>SMEM</b> SMEM-b	
	<b>SDev</b> SDev-b	<b>SMPL</b> SMPL-b	
se		<b>SND</b> SND	
	<b>se</b> se-b	<b>Snd</b> Snd	
sel		<b>Solv</b> Solv	
	<b>SEL</b> SEL		
	<b>SEL-b</b> SEL-b		
sell			
	<b>Sell</b> Sell-b		
sels			
	<b>SELS</b> SELS-b		
send			
	<b>Send</b> Send-b		





sto	<b>T</b> T	<b>TOPL</b> TOPleft
<b>STO</b> STO-b	<b>t</b> t-b	<b>TOPT</b> TOPTop
<b>Sto</b> Sto-b	<b>t2</b> t-b2	tpd
stop	<b>T</b> Tera-b	<b>TPD</b> tpd
<b>STOP</b> STOP	<b>tsnd</b> tsnd-b	tran
<b>Stop</b> Stop-b	<b>Ttheta</b> Ttheta-b	<b>TRAN</b> TRAN
str	tabl	<b>TRAN</b> TRAN-b
<b>STR</b> STR	<b>TABL</b> TABL	trig
<b>STR</b> STR-b	<b>TABL</b> TABL-b	<b>TRIG</b> TRIG
<b>Str</b> Str-b	<b>Tabl</b> Tabl-b	trn
strp	tang	<b>TRN</b> Trn-b
<b>STRP</b> STRP-b	<b>Tang</b> Tang	tup
strt	<b>Tang</b> Tang-b	<b>tUp</b> tUp-b
<b>STRT</b> STRT	tanh	tvm
<b>Strt</b> Strt-b	<b>TANH</b> tanh-b	<b>TVM</b> TVM-b
stup	tanh1	type
<b>STUP</b> STUP-b	<b>TANH1</b> tanh1-b	<b>TYPE</b> TYPE-b
styl	tcd	unit
<b>STYL</b> STYL-b	<b>tcd</b> tcd	<b>UNIT</b> UNIT-b
sum	test	upr
<b>SUM</b> Sum-b	<b>TEST</b> TEST-b	<b>UPR</b> Upr-b
svas	<b>Test</b> Test-b	usb
<b>SVAS</b> SVAS-b	text	<b>USB</b> USB
swap	<b>TEXT</b> TEXT	var
<b>SWAP</b> SWAP	<b>Text</b> Text	<b>VAR</b> var
sx	<b>Text</b> Text-b	<b>VAR</b> VAR-b
<b>SX</b> sx-b	then	<b>Var</b> Var-b
sx1	<b>THEN</b> Then-b	vct
<b>SX1</b> sx1-b	time	<b>VCT</b> VCT-b
sx2	<b>TIME</b> TIME-b	velo
<b>SX2</b> sx2-b	tlow	<b>VELO</b> VELO-b
sy	<b>tLow</b> tLow-b	ver
<b>SY</b> sy-b	tmpr	<b>VER</b> VER-b
sybl	<b>TMPR</b> Tmpr-b	vert
<b>SYBL</b> SYBL	to	<b>VERT</b> Vert
<b>SYBL</b> SYBL-b	<b>To</b> To-b	<b>VERT</b> Vert-b
syd	tool	vlum
<b>SYD</b> SYD	<b>TOOL</b> TOOL-b	<b>VLUM</b> VLUM-b
t	top	vnlk
	<b>TOP</b> TOP	<b>VNLK</b> VNLK-b
		vrnr
		<b>VRNR</b> VRNR-b



**vwin**  
 $\overline{VWIN}$  VWIN-b  
 $\overline{VWin}$  VWin-b  
**wake**  
 $\overline{WAKE}$  WAKE-b  
**web**  
 $\overline{WEB}$  WEB  
 $\overline{Web}$  Web-b  
**wend**  
 $\overline{WEnd}$  WEnd-b  
**while**  
 $\overline{While}$  While-b  
**wiz**  
 $\overline{WIZ}$  WIZ-b  
**x**  
 $\overline{X!}$  factorialx-b  
 $\overline{Sx}$  sigmax-b  
 $\overline{Sx}$  Sx-b  
 $\overline{tx=}$  txequal  
 $\overline{tx\ge}$  txgeq  
 $\overline{tx>}$  txgt  
 $\overline{tx\le}$  txleq  
 $\overline{tx<}$  txlt  
 $\overline{x}$  x  
 $\overline{X}$  X-b  
 $\overline{x}$  x-b  
 $\overline{X}$  X-b2  
 $\overline{X}$  X-b3  
 $\overline{x}$  xbar-b  
 $\overline{tx=}$  xequal  
 $\overline{tx\ge}$  xequal-b  
 $\overline{tx\ge}$  xgeq-b  
 $\overline{tx>}$  xgt-b  
 $\overline{x}$  xhat-b  
 $\overline{tx\le}$  xleq-b  
 $\overline{tx<}$  xlt-b  
**x1**  
 $\overline{x1}$  x1-b  
 $\overline{x1}$  xbar1-b  
**x1inv**  
 $\overline{x1Inv}$  x1Inv-b  
**x2**  
 $\overline{Sx2}$  Sx2-b  
 $\overline{X2}$  X2  
 $\overline{x2}$  x2  
 $\overline{x2}$  x2-b  
 $\overline{x2}$  xbar2-b  
 $\overline{x2}$  xpower2-b  
**x2inv**  
 $\overline{x2Inv}$  x2Inv-b  
**x3**  
 $\overline{X3}$  X3  
 $\overline{x3}$  x3  
 $\overline{x3}$  x3-b  
 $\overline{x3}$  xpower3-b  
**x4**  
 $\overline{X4}$  X4  
 $\overline{x4}$  x4  
 $\overline{x4}$  xpower4-b  
**xcal**  
 $\overline{XCAL}$  XCAL  
**xfct**  
 $\overline{Xfct}$  Xfct-b  
**xinv**  
 $\overline{xInv}$  xInv-b  
**xor**  
 $\overline{Xor}$  Xor-b  
**xrw**  
 $\overline{XRw}$  XRw  
 $\overline{XRw+}$  XRwplus  
**xt**  
 $\overline{Xt}$  Xt-b  
**xy**  
 $\overline{Sxy}$  Sxy-b  
 $\overline{xy}$  xy-b  
**y**  
 $\overline{sigmay}$  sigmay-b  
 $\overline{Sy}$  Sy-b  
 $\overline{tY=}$  tYequal  
 $\overline{tY\ge}$  tYgeq  
 $\overline{tY>}$  tYgt  
 $\overline{tY\le}$  tYleq  
 $\overline{tY<}$  tYlt  
 $\overline{Y}$  Y  
 $\overline{Y}$  Y-b  
 $\overline{Y}$  Y-b2  
 $\overline{y}$  ybar-b  
 $\overline{Y=}$  Yequal  
 $\overline{Y=}$  Yequal-b  
 $\overline{Y\ge}$  Ygeq-b  
 $\overline{Y>}$  Ygt-b  
 $\overline{y}$  yhat-b  
 $\overline{Y\le}$  Yleq-b  
 $\overline{Y<}$  Ylt-b  
**y1**  
 $\overline{y1}$  y1-b  
**y2**  
 $\overline{Sy2}$  Sy2-b  
 $\overline{y2}$  y2-b  
**y3**  
 $\overline{y3}$  y3-b  
**ycal**  
 $\overline{YCAL}$  YCAL  
**yes**  
 $\overline{YES}$  YES  
**yfct**  
 $\overline{Yfct}$  Yfct-b  
**yicpt**  
 $\overline{YICPT}$  YICPT  
**yld**  
 $\overline{YLD}$  YLD  
 $\overline{YLD}$  YLD-b  
**yt**  
 $\overline{Yt}$  Yt-b  
**z**  
 $\overline{Z}$  Z  
 $\overline{Z}$  Z-b  
 $\overline{Z}$  z-b  
**zero**  
 $\overline{ZERO}$  ZERO  
**zlow**  
 $\overline{zLow}$  zLow-b  
**zoom**  
 $\overline{ZOOM}$  ZOOM  
 $\overline{ZOOM}$  ZOOM-b  
**zup**  
 $\overline{zUp}$  zUp-b
















































### C.3 Batterie

Liste des indicateurs de charge de batterie.

-  \battery{empty}
-  \battery{low}
-  \battery{high}
-  \battery{medium}

## D Touches

Les touches sont ici classées dans un ordre arbitraire. Pour les retrouver sur la calculatrice, voir la figure 6.

- |  |  |   |
|--|--|---|
| <ul style="list-style-type: none"> <li>—  \key{ACON}</li> <li>—  \key{DEL}</li> <li>—  \key{ALPHA}</li> <li>—  \key{EXE}</li> <li>—  \key{F5}</li> <li>—  \key{F4}</li> <li>—  \key{F1}</li> <li>—  \key{F6}</li> <li>—  \key{F3}</li> <li>—  \key{F2}</li> <li>—  \key{MENU}</li> <li>—  \key{EXIT}</li> <li>—  \key{FD}</li> <li>—  \key{OPTN}</li> <li>—  \key{VARs}</li> </ul> | <ul style="list-style-type: none"> <li>—  \key{XthetaT}</li> <li>—  \key{closeparen}</li> <li>—  \key{comma}</li> <li>—  \key{cos}</li> <li>—  \key{fraction}</li> <li>—  \key{ln}</li> <li>—  \key{log}</li> <li>—  \key{openparen}</li> <li>—  \key{power}</li> <li>—  \key{rightarrow}</li> <li>—  \key{sin}</li> <li>—  \key{square}</li> <li>—  \key{tan}</li> <li>—  \key{1}</li> <li>—  \key{10^n}</li> <li>—  \key{2}</li> <li>—  \key{3}</li> </ul> | <ul style="list-style-type: none"> <li>—  \key{4}</li> <li>—  \key{5}</li> <li>—  \key{6}</li> <li>—  \key{7}</li> <li>—  \key{8}</li> <li>—  \key{9}</li> <li>—  \key{divide}</li> <li>—  \key{dot}</li> <li>—  \key{minus}</li> <li>—  \key{opposite}</li> <li>—  \key{plus}</li> <li>—  \key{times}</li> <li>—  \key{zero}</li> <li>—  \key{REPLAY}</li> <li>—  \key{SHIFT}</li> </ul> |
|--|--|---|

## E Implementation

Load some packages.

```

1 \RequirePackage{etoolbox}
2
3 \RequirePackage{pixelart}
4 \RequirePackage{tikz}
5 \usetikzlibrary{calc}
6 \RequirePackage{pgfkeys}
7 \RequirePackage{etoolbox}
8 \RequirePackage{amssymb}
9 \RequirePackage{amsbsy}
10 \RequirePackage{sansmath}
11 \RequirePackage{letterspace}
12 \RequirePackage{pgfopts}

```

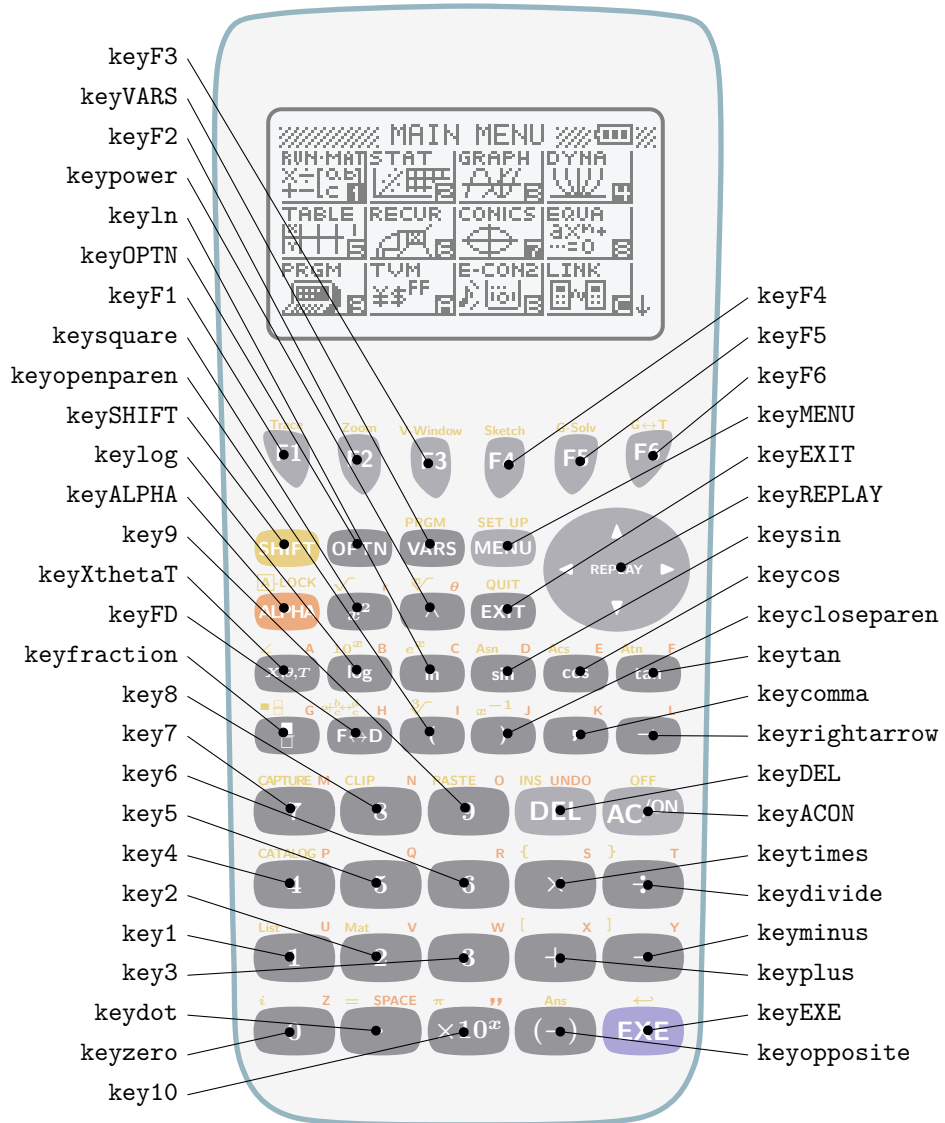


FIGURE 6 – Mots-clefs des touches

```

13
14 \RequirePackage{graph35-pixelart}
15 \RequirePackage{graph35-keys}
    Define package arguments
16 \pgfkeys{
17   % color option
18   /GRAPH35/.cd,
19   color/.is choice,
20   color/real/.code={\def\graph@color{real}},
21   color/blackandwhite/.code={\def\graph@color{blackandwhite}},
22   color/.value required,
23   color={real},
24 }
25 \ProcessPgfPackageOptions{/GRAPH35}
    Define colors.
26 \newcommand{\setgraphcolor@default}{}
27 \newcommand{\setgraphcolor@blackandwhite}{
28   \colorlet{graph35ACON}{white}
29   \colorlet{graph35ACONBORDER}{black}
30   \colorlet{graph35SHIFT}{white}
31   \colorlet{graph35SHIFTTEXT}{black}
32   \colorlet{graph35SHIFTBORDER}{black}
33   \colorlet{graph35CASE}{white}
34   \colorlet{graph35CASEBORDER}{black}
35   \colorlet{graph35ALPHA}{white}
36   \colorlet{graph35ALPHATEXT}{black}
37   \colorlet{graph35ALPHABORDER}{black}
38   \colorlet{graph35EXE}{white}
39   \colorlet{graph35EXEBORDER}{black}
40   \colorlet{graph35WHITE}{white}
41   \colorlet{graph35KEYTEXT}{black}
42   \colorlet{graph35SCREEN}{black}
43   \colorlet{graph35NUMBER}{white}
44   \colorlet{graph35NUMBERBORDER}{black}
45   \colorlet{graph35SCREENBG}{white}
46 }
47 \newcommand{\setgraphcolor@real}{
48   \colorlet{graph35KEYTEXT}{white}
49   \definecolor{graph35ACON}{RGB}{99, 95, 110}
50   \colorlet{graph35ACONBORDER}{graph35ACON}
51   \definecolor{graph35SHIFT}{RGB}{209, 164, 11}
52   \definecolor{graph35SHIFTTEXT}{RGB}{209, 164, 11}
53   \colorlet{graph35SHIFTBORDER}{graph35SHIFT}
54   \definecolor{graph35CASE}{RGB}{236, 236, 236}
55   \definecolor{graph35CASEBORDER}{RGB}{43, 108, 130}
56   \definecolor{graph35ALPHA}{RGB}{220, 89, 1}
57   \definecolor{graph35ALPHATEXT}{RGB}{220, 89, 1}
58   \colorlet{graph35ALPHABORDER}{graph35ALPHA}

```

```

59 \definecolor{graph35EXE}{RGB}{86, 78, 177}
60 \colorlet{graph35EXEBORDER}{graph35EXE}
61 \definecolor{graph35WHITE}{RGB}{255, 255, 255}
62 \definecolor{graph35SCREEN}{RGB}{0, 0, 0}
63 \definecolor{graph35NUMBER}{RGB}{44, 44, 54}
64 \colorlet{graph35NUMBERBORDER}{graph35NUMBER}
65 \definecolor{graph35SCREENBG}{RGB}{234,240,241}
66 }
67
68 \csuse{setgraphcolor@\graph@color}

\setgraphcolor Define \setgraphcolor, to set default color set.
69 \newcommand{\setgraphcolor}[1]{
70   \csuse{setgraphcolor@#1}%
71 }

\menu Define \menu, which is basically a call to \tikzmenu.
72 \newcommand{\menu}[2]{%
73   \begin{tikzpicture}[baseline=-2.7ex]%
74     \tikzmenu[scale=.03]{#1}{#2}{(0, 0)}
75   \end{tikzpicture}%
76 }

\tikzmenu Define \tikzmenu, which is a wrapper to the \tikzmenu@* macros defined in
graph35-pixelart.sty.
77 \newcommand{\tikzmenu}[4][]{%
78   \ifcsdef{tikzmenu@#2}{%
79     \csuse{tikzmenu@#2}[#1]{#3}{#4}
80   }{%
81     \ClassError{graph35}{Unknown menu '#2'}{Look for the list of available menus in the package}
82   }%
83 }

\battery Define \battery, which is basically a call to \tikzbattery.
84 \newcommand{\battery}[1]{%
85   \begin{tikzpicture}[baseline=-1.3ex]%
86     \tikzbattery[scale=.03]{#1}{(0, 0)}
87   \end{tikzpicture}%
88 }

\tikzbattery Define \tikzbattery, which is a wrapper to the \tikzbattery@* macros defined
in graph35-pixelart.sty.
89 \newcommand{\tikzbattery}[3][]{%
90   \ifcsdef{tikzbattery@#2}{%
91     \csuse{tikzbattery@#2}[#1]{#3}
92   }{%
93     \ClassError{graph35}{Unknown battery status '#2'}{Look for the list of available battery st
94   }%
95 }

```

```

\function Define \function, which is basically a call to \tikzfunction.
96 \newcommand{\function}[1]{%
97   \begin{tikzpicture}[baseline=-1.3ex]%
98     \tikzfunction[scale=.03]{#1}{(0, 0)}
99   \end{tikzpicture}%
100 }

\tikzfunction Define \tikzfunction, which is a wrapper to the \tikzfunction@* macros defi-
ned in graph35-pixelart.sty.
101 \newcommand{\tikzfunction}[3] []{%
102   \ifcsdef{tikzfunction@#2}{%
103     \csuse{tikzfunction@#2}[#1]{#3}%
104   }{%
105     \ClassError{graph35}{Unknown function menu '#2'}{Look for the list of available function me
106   }%
107 }

    Parsing \key arguments (and \graph@tikzshiftalpha, used in \key to draw
    the shift and alpha texts).
108 \newif\ifshow@shift
109 \newif\ifshow@alpha
110 \newcommand{\boolvalue}[1]{\csuse{if#1>true\else false\fi}
111 \pgfkeys{
112   /GRAPH35/KEY/.is family,
113   /GRAPH35/KEY,
114   prefix/.code={\pgfkeyssetvalue{/GRAPH35/KEY/prefix}{#1}},
115   prefix/.value required,
116   prefix={key},
117   suffix/.code={\pgfkeyssetvalue{/GRAPH35/KEY/suffix}{#1}},
118   suffix/.value required,
119   suffix={},
120   name/.code={\pgfkeyssetvalue{/GRAPH35/KEY/name}{#1}},
121   name/.value required,
122   name={},
123   scale/.code={\pgfkeyssetvalue{/GRAPH35/KEY/scale}{#1}},
124   scale/.value required,
125   scale=1,
126   type/.is choice,
127   type/text/.code={\def\graph@type{text}},
128   type/formula/.code={\def\graph@type{formula}},
129   type/.value required,
130   content/.code={\pgfkeyssetvalue{/GRAPH35/KEY/content}{#1}},
131   content/.value required,
132   content={},
133   shift/.is if=show@shift,
134   shift=false,
135   shift type/.code={\pgfkeyssetvalue{/GRAPH35/KEY/shift type}{#1}},
136   shift type/.value required,
137   shift type={text},

```

```

138 shift content/.code={\pgfkeyssetvalue{/GRAPH35/KEY/shift content}{#1}},
139 shift content/.value required,
140 shift content={},
141 shift position/.code={\pgfkeyssetvalue{/GRAPH35/KEY/shift position}{#1}},
142 shift position/.value required,
143 shift position=left,
144 alpha/.is if=show@alpha,
145 alpha=false,
146 alpha type/.code={\pgfkeyssetvalue{/GRAPH35/KEY/alpha type}{#1}},
147 alpha type/.value required,
148 alpha type={text},
149 alpha content/.code={\pgfkeyssetvalue{/GRAPH35/KEY/alpha content}{#1}},
150 alpha content/.value required,
151 alpha content={},
152 alpha position/.code={\pgfkeyssetvalue{/GRAPH35/KEY/alpha position}{#1}},
153 alpha position/.value required,
154 alpha position=right,
155 color/.is choice,
156 color/real/.code={\def\graph@tempcolor{real}},
157 color/blackandwhite/.code={\def\graph@tempcolor{blackandwhite}},
158 color/default/.code={\def\graph@tempcolor{default}},
159 color/.value required,
160 color={default},
161 }
162
163 \newif\if@show
164 \pgfkeys{
165 /GRAPH35/SHIFTALPHA/.is family,
166 /GRAPH35/SHIFTALPHA,
167 show/.is if=@show,
168 show=false,
169 name/.code={\pgfkeyssetvalue{/GRAPH35/SHIFTALPHA/name}{#1}},
170 name/.value required,
171 name={graph35@tempname},
172 type/.is choice,
173 type/text/.code={\def\graph@type{text}},
174 type/formula/.code={\def\graph@type{formula}},
175 type/.value required,
176 content/.code={\pgfkeyssetvalue{/GRAPH35/SHIFTALPHA/content}{#1}},
177 content/.value required,
178 content={},
179 style/.code={\pgfkeyssetvalue{/GRAPH35/SHIFTALPHA/style}{#1}},
180 style/.value required,
181 style={},
182 position/.is choice,
183 position/left/.code={\def\temp@position{left}},
184 position/right/.code={\def\temp@position{right}},
185 position/center/.code={\def\temp@position{center}},
186 node/.code={\pgfkeyssetvalue{/GRAPH35/SHIFTALPHA/node}{#1}},
187 node/.value required,

```

```

188 color/.code={\pgfkeyssetvalue{/GRAPH35/SHIFTALPHA/color}{#1}},
189 color/.value required,
190 color=graph35SCREEN,
191 }

```

`\key` Definition of `\key`, which is a wrapper to `\tikzkey`.

```

192 \newcommand{\key}[2] [] {%
193   \begin{tikzpicture}[baseline=(@origin.base), scale=.5, transform shape, every node/.style={
194     \node (@origin) at (0, 0) {\strut};
195     \tikzkey[#1]{#2}{(0, 0)}%
196   \end{tikzpicture}%
197 }

```

`\tikzkey` Definition of `\tikzkey`, which is a wrapper to a bunch of `\tikzkey@*` macros, one for each key.

```

198 \newcommand{\tikzkey}[3] [] {%
199   \ifcsdef{tikzkey@#2}{%
200     \csuse{tikzkey@#2}[#1]{#3}%
201   }{%
202     \ClassError{graph35}{Unknown key '#2'}{Look for the list of available keys in the package d
203   }%
204 }

```


Define `\graph@tikzshiftalpha`, used internally to draw shift and alpha text or symbols.

```

205
206 \newcommand{\graph@tikzshiftalpha}[1]{%
207   \pgfkeys{/GRAPH35/SHIFTALPHA, #1}
208   \if@show
209     \ifdefstring{\temp@position}{left}{
210       \def\graph@anchor{south west}
211       \coordinate (graph@node) at (\pgfkeysvalueof{/GRAPH35/SHIFTALPHA/node}.north west) ;
212     }{}
213     \ifdefstring{\temp@position}{right}{
214       \def\graph@anchor{south east}
215       \coordinate (graph@node) at (\pgfkeysvalueof{/GRAPH35/SHIFTALPHA/node}.north east) ;
216     }{}
217     \ifdefstring{\temp@position}{center}{
218       \def\graph@anchor{south}
219       \coordinate (graph@node) at (\pgfkeysvalueof{/GRAPH35/SHIFTALPHA/node}.north) ;
220     }{}
221     \ifdefstring{\graph@type}{text}{
222       \node[inner sep=1pt, color=\pgfkeysvalueof{/GRAPH35/SHIFTALPHA/color}, anchor=\graph@anch
223     ]{
224       \node[inner sep=1pt, color=\pgfkeysvalueof{/GRAPH35/SHIFTALPHA/color}, anchor=\graph@anch
225     }
226     \else\fi
227 }}

```



Define number keys (e.g. ) .

```

228 \newcommand{\graph@tikzgenerickeynumber}[2] [] {{
229   \pgfkeys{/GRAPH35/KEY, #1}%
230   \setgraphcolor{\graph@tempcolor}%
231   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
232     \draw[very thick, color=graph35NUMBERBORDER, fill=graph35NUMBER] plot [smooth cycle] coordi
233       (-.43, .25)
234       (.43, .25)
235       (.43, -.25)
236       (-.43, -.25)
237     };
238     \node[minimum width=1.03cm, minimum height=.59cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/
239     \ifdefstring{\graph@type}{text}{
240       \node[color=graph35KEYTEXT] {\bfseries\sffamily\pgfkeysvalueof{/GRAPH35/KEY/content}} ;
241     }{
242       \node[color=graph35KEYTEXT] {$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$} ;
243     }
244
245     \graph@tikzshifitalpha{,
246       show=\boolvalue{show@shift},
247       name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
248       position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
249       node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
250       color=graph35SHIFTEXT,
251       type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
252       content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
253       style={\tiny},
254     }
255     \graph@tikzshifitalpha{,
256       show=\boolvalue{show@alpha},
257       name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
258       position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
259       node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
260       color=graph35ALPHATEXT,
261       type={\pgfkeysvalueof{/GRAPH35/KEY/alpha type}},
262       content={\pgfkeysvalueof{/GRAPH35/KEY/alpha content}},
263       style={\tiny},
264     }
265   \end{scope}
266   \setgraphcolor{\graph@color}%
267 }}

```

Define ACON key  .

```

268 \newcommand{\graph@tikzgenerickeyACON}[2] [] {{
269   \pgfkeys{/GRAPH35/KEY, #1}%
270   \setgraphcolor{\graph@tempcolor}%
271   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
272     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
273       (-.43, .25)
274       (.43, .25)

```

```

275     (.43, -.25)
276     (-.43, -.25)
277 };
278 \node[minimum width=1.03cm, minimum height=.59cm, inner sep=Opt] (\pgfkeysvalueof{/GRAPH35/
279 \ifdefstring{\graph@type}{text}{
280   \node[color=graph35KEYTEXT] {\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/content}}};
281 }{
282   \node[color=graph35KEYTEXT] {$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$};
283 }
284
285 \graph@tikzshiftalpha{
286   show=\boolvalue{show@shift},
287   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
288   position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
289   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
290   color=graph35SHIFTTEXT,
291   type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
292   content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
293   style={\tiny},
294 }
295 \graph@tikzshiftalpha{
296   show=\boolvalue{show@alpha},
297   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
298   position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
299   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
300   color=graph35ALPHATEXT,
301   type={\pgfkeysvalueof{/GRAPH35/KEY/alpha type}},
302   content={\pgfkeysvalueof{/GRAPH35/KEY/alpha content}},
303   style={\tiny},
304 }
305 \end{scope}
306 \setgraphcolor{\graph@color}%
307 }}

```

Define EXE key EXE.

```


308 \newcommand{\graph@tikzgenerickeyexe}[2] [] {{
309   \pgfkeys{/GRAPH35/KEY, #1}%
310   \setgraphcolor{\graph@tempcolor}%
311   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
312     \draw[very thick, color=graph35EXEBORDER, fill=graph35EXE] plot [smooth cycle] coordinates
313     (-.43, .25)
314     (.43, .25)
315     (.43, -.25)
316     (-.43, -.25)
317   };
318   \node[minimum width=1.03cm, minimum height=.59cm, inner sep=Opt] (\pgfkeysvalueof{/GRAPH35/
319   \ifdefstring{\graph@type}{text}{
320     \node[color=graph35KEYTEXT] {\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/content}}};
321   }{
322     \node[color=graph35KEYTEXT] {$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$};

```

```

323   }
324
325   \graph@tikzshiftalpha{,
326     show=\boolvalue{show@shift},
327     name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
328     position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
329     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
330     color=graph35SHIFTTEXT,
331     type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
332     content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
333     style={\tiny},
334   }
335 \end{scope}
336 \setgraphcolor{\graph@color}%
337 }}

```

Define SHIFT key .

```

338 \newcommand{\graph@tikzgenerickeyshift}[2] [] {{
339   \pgfkeys{/GRAPH35/KEY, #1}%
340   \setgraphcolor{\graph@tempcolor}%
341   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
342     \draw[very thick, color=graph35SHIFTBORDER, fill=graph35SHIFT] plot [smooth cycle] coordina
343     (-.34, .19)
344     (.34, .19)
345     (.34, -.19)
346     (-.34, -.19)
347   };
348   \node[minimum width=0.8cm, minimum height=.45cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K
349   \ifdefstring{\graph@type}{text}{
350     \node[color=graph35KEYTEXT] {\scriptsize\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/
351   }}
352     \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content
353   }
354 \end{scope}
355 \setgraphcolor{\graph@color}%
356 }}

```

Define REPLAY key .

```

357 \newcommand{\graph@tikzgenerickeyreplay}[2] [] {{
358   \pgfkeys{/GRAPH35/KEY, #1}%
359   \setgraphcolor{\graph@tempcolor}%
360   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
361     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] (0, 0) ellipse (.95 and .80) ;
362     \node[minimum width=0pt, minimum height=0pt, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/p
363     .64, 0) {} ;
364     \node[minimum width=0pt, minimum height=0pt, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/p
365     \node[minimum width=0pt, minimum height=0pt, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/p
366     .48) {} ;
367     \node[minimum width=1.9cm, minimum height=1.6cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K

```

```

367 \ifdefstring{\graph@type}{text}{
368 \node[color=graph35KEYTEXT] {\tiny\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/content}}
369 }{
370 \node[color=graph35KEYTEXT] {\tiny$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$} ;
371 }
372 \begin{scope}[rounded corners=1pt, color=graph35KEYTEXT]
373 \fill (.58, .09) -- (.58, -.08) -- (.78, 0) -- cycle ;
374 \fill (-.58, .09) -- (-.58, -.08) -- (-.78, 0) -- cycle ;
375 \fill (-.08, .41) -- (.09, .41) -- (0, .62) -- cycle ;
376 \fill (-.08, -.41) -- (.09, -.41) -- (0, -.62) -- cycle ;
377 \end{scope}
378 \end{scope}
379 \setgraphcolor{\graph@color}%
380 }}


```

Define ALPHA key .

```

381 \newcommand{\graph@tikzgenerickeyalpha}[2] [] {{
382 \pgfkeys{/GRAPH35/KEY, #1}%
383 \setgraphcolor{\graph@tempcolor}%
384 \begin{scope}[shift=#2], scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every node/.style={
385 \draw[very thick, color=graph35ALPHABORDER, fill=graph35ALPHA] plot [smooth cycle] coordinates
386 (-.34, .19)
387 (.34, .19)
388 (.34, -.19)
389 (-.34, -.19)
390 } ;
391 \node[minimum width=0.8cm, minimum height=.45cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/content})
392 \ifdefstring{\graph@type}{text}{
393 \node[color=graph35KEYTEXT] {\scriptsize\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/content}}
394 }{
395 \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$}
396 }
397 }
398 \graph@tikzshiftalpha{,
399 show=\boolvalue{show@shift},
400 name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalueof{/GRAPH35/KEY/shift position}},
401 position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
402 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
403 color=graph35SHIFTEXT,
404 type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
405 content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
406 style={\tiny},
407 }
408 \end{scope}
409 \setgraphcolor{\graph@color}%
410 }}

```

Define menu key .

```


411 \newcommand{\graph@tikzgenerickeymenu}[2] [] {{
412 \pgfkeys{/GRAPH35/KEY, #1}%
413 \setgraphcolor{\graph@tempcolor}%

```

```

414 \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
415 \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
416 (-.34, .19)
417 (.34, .19)
418 (.34, -.19)
419 (-.34, -.19)
420 };
421 \node[minimum width=.8cm, minimum height=.45cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/
422 \ifdefstring{\graph@type}{text}{
423 \node[color=graph35KEYTEXT] {\scriptsize\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/
424 }}{
425 \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content
426 }}
427 }
428 \graph@tikzshiftalpha{,
429 show=\boolvalue{show@shift},
430 name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
431 position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
432 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
433 color=graph35SHIFTTEXT,
434 type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
435 content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
436 style={\tiny},
437 }
438 \graph@tikzshiftalpha{,
439 show=\boolvalue{show@alpha},
440 name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
441 position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
442 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
443 color=graph35ALPHATEXT,
444 type={\pgfkeysvalueof{/GRAPH35/KEY/alpha type}},
445 content={\pgfkeysvalueof{/GRAPH35/KEY/alpha content}},
446 style={\tiny},
447 }
448 \end{scope}
449 \setgraphcolor{\graph@color}%
450 }}

```

Define miscallaenous keys (e.g. ) .

```


451 \newcommand{\graph@tikzgenerickeymisc}[2][ ]{{
452 \pgfkeys{/GRAPH35/KEY, #1}%
453 \setgraphcolor{\graph@tempcolor}%
454 \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
455 \draw[very thick, color=graph35NUMBERBORDER, fill=graph35NUMBER] plot [smooth cycle] coordi
456 (-.34, .19)
457 (.34, .19)
458 (.34, -.19)
459 (-.34, -.19)
460 };
461 \node[minimum width=0.8cm, minimum height=.45cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K

```

```

462 \ifdefstring{\graph@type}{text}{
463   \node[color=graph35KEYTEXT] {\scriptsize\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/
464 }}{
465   \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content
466 }}
467 }
468 \graph@tikzshiftalpha{,
469   show=\boolvalue{show@shift},
470   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
471 position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
472   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
473 color=graph35SHIFTTEXT,
474   type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
475   content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
476   style={\tiny},
477 }
478 \graph@tikzshiftalpha{,
479   show=\boolvalue{show@alpha},
480   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
481 position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
482   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
483 color=graph35ALPHATEXT,
484   type={\pgfkeysvalueof{/GRAPH35/KEY/alpha type}},
485   content={\pgfkeysvalueof{/GRAPH35/KEY/alpha content}},
486   style={\tiny},
487 }
488 \end{scope}
489 \setgraphcolor{\graph@color}%
490 }}

```

Define F1 key .

```


491 \newcommand{\graph@tikzgenerickeyfone}[2][{}]{%
492   \pgfkeys{/GRAPH35/KEY, #1}%
493   \setgraphcolor{\graph@tempcolor}%
494   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
495     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
496       (-.3, .25)
497       (.2, .20)
498       (.25, -.18)
499       (.1, -.45)
500       (-.20, -.15)
501   };
502   \node[minimum width=.56cm, minimum height=.70cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K
503 .03, -.09) {};
504   \ifdefstring{\graph@type}{text}{
505     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/conte
506 }}{
507     \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}}$};
508   }

```

```

509 \graph@tikzshiftalpha{,
510   show=\boolvalue{show@shift},
511   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
512   position=center,
513   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
514   color=graph35SHIFTTEXT,
515   type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
516   content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
517   style={\tiny},
518 }
519 \end{scope}
520 \setgraphcolor{\graph@color}%
521 }}


```

Define F2 key .

```

522 \newcommand{\graph@tikzgenerickeyftwo}[2][{}]{%
523 \pgfkeys{/GRAPH35/KEY, #1}%
524 \setgraphcolor{\graph@tempcolor}%
525 \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
526 \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
527 (-.25, .25)
528 (.2, .25)
529 (.25, -.20)
530 (.05, -.46)
531 (-.20, -.20)
532 ];
533 \node[minimum width=.52cm, minimum height=.74cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K
534 .005, -.08) {};
535 \ifdefstring{\graph@type}{text}{
536 \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/conte
537 \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$};
538 }
539 }
540 \graph@tikzshiftalpha{,
541   show=\boolvalue{show@shift},
542   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
543   position=center,
544   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
545   color=graph35SHIFTTEXT,
546   type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
547   content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
548   style={\tiny},
549 }
550 \end{scope}
551 \setgraphcolor{\graph@color}%
552 }}

```

Define F3 key .

```


553 \newcommand{\graph@tikzgenerickeyfthree}[2][{}]{%

```

```

554 \pgfkeys{/GRAPH35/KEY, #1}%
555 \setgraphcolor{\graph@tempcolor}%
556 \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
557 \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
558 (.15, .25)
559 (-.25, .25)
560 (-.25, -.20)
561 (.03, -.47)
562 (.20, -.20)
563 };
564 \node[minimum width=.48cm, minimum height=.75cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K
.04, -.085) {};
565 \ifdefstring{\graph@type}{text}{
566 \node[color=graph35KEYTEXT] {\small\bfseries\sffamily}\pgfkeysvalueof{/GRAPH35/KEY/conte
567 }{
568 \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$};
569 }
570
571 \graph@tikzshiftalpha{
572 show=\boolvalue{show@shift},
573 name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
574 position=center,
575 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
576 color=graph35SHIFTTEXT,
577 type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
578 content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
579 style={\tiny},
580 }
581 \end{scope}
582 \setgraphcolor{\graph@color}%
583 }}

```

Define F4 key .

```

584 \newcommand{\graph@tikzgenerickeyffour}[2][\{}{
585 \pgfkeys{/GRAPH35/KEY, #1}%
586 \setgraphcolor{\graph@tempcolor}%
587 \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
588 \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
589 (-.15, .25)
590 (.25, .25)
591 (.25, -.20)
592 (-.03, -.47)
593 (-.20, -.20)
594 };
595 \node[minimum width=.48cm, minimum height=.75cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K
.085) {};
596 \ifdefstring{\graph@type}{text}{
597 \node[color=graph35KEYTEXT] {\small\bfseries\sffamily}\pgfkeysvalueof{/GRAPH35/KEY/conte
598 }{
599 \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$};


```



```

600   }
601
602   \graph@tikzshiftalpha{,
603     show=\boolvalue{show@shift},
604     name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
605     position=center,
606     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
607     color=graph35SHIFTTEXT,
608     type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
609     content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
610     style={\tiny},
611   }
612 \end{scope}
613 \setgraphcolor{\graph@color}%
614 }}


```

Define F5 key .

```

615 \newcommand{\graph@tikzgenerickeyffive}[2] [] { {%
616   \pgfkeys{/GRAPH35/KEY, #1}%
617   \setgraphcolor{\graph@tempcolor}%
618   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
619     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
620     (.25, .25)
621     (-.2, .25)
622     (-.25, -.20)
623     (-.05, -.46)
624     (.20, -.20)
625   };
626   \node[minimum width=.52cm, minimum height=.74cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K
627   .08) {} ;
628   \ifdefstring{\graph@type}{text}{
629     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/conte
630   }{
631     \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$} ;
632   }
633   \graph@tikzshiftalpha{,
634     show=\boolvalue{show@shift},
635     name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
636     position=center,
637     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
638     color=graph35SHIFTTEXT,
639     type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
640     content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
641     style={\tiny},
642   }
643 \end{scope}
644 \setgraphcolor{\graph@color}%
645 }}

```

Define F6 key .

```

646 \newcommand{\graph@tikzgenerickeyfsix}[2][\]{%
647   \pgfkeys{/GRAPH35/KEY, #1}%
648   \setgraphcolor{\graph@tempcolor}%
649   \begin{scope}[shift={#2}, scale=\pgfkeysvalueof{/GRAPH35/KEY/scale}, transform shape, every n
650     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinate
651     (.3, .25)
652     (-.2, .20)
653     (-.25, -.18)
654     (-.1, -.45)
655     (.20, -.15)
656   };
657   \node[minimum width=.56cm, minimum height=.70cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/K
.09) {} ;
658   \ifdefstring{\graph@type}{text}{
659     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/conte
660   }}{
661     \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$} ;
662   }
663
664   \graph@tikzshiftalpha{
665     show=\boolvalue{show@shift},
666     name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
667     position=center,
668     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
669     color=graph35SHIFTTEXT,
670     type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
671     content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
672     style={\tiny},
673   }
674   \end{scope}
675   \setgraphcolor{\graph@color}%
676 }}

```

Parse `\calculator` and `\tikzcalculator` options.

```

677 \pgfkeys{
678   /GRAPH35/CALCULATOR/.is family,
679   /GRAPH35/CALCULATOR,
680   scale/.value required,
681   scale/.code={\pgfkeyssetvalue{/GRAPH35/CALCULATOR/scale}{#1}},
682   scale=1,
683   color/.is choice,
684   color/real/.code={\def\graph@tempcolor{real}},
685   color/blackandwhite/.code={\def\graph@tempcolor{blackandwhite}},
686   color/default/.code={\def\graph@tempcolor{default}},
687   color/.value required,
688   color={default},
689 }

```

`\calculator` Define `\calculator`, which is a wrapper to `\tikzcalculator`.

```

690 \newcommand{\calculator}[2] [] {%
691   \pgfkeys{/GRAPH35/CALCULATOR, #1}%
692   \setgraphcolor{\graph@tempcolor}%
693   \begin{tikzpicture}[scale=\pgfkeysvalueof{/GRAPH35/CALCULATOR/scale}, transform shape]%
694     \tikzcalculator{#2}
695   \end{tikzpicture}%
696   \setgraphcolor{\graph@color}%
697 }}

\tikzcalculator Define \tikzcalculator, to draw calculators (see annex A). This is a wrapper to
the \tikzcalculator@* macros, defined later.
698 \newcommand{\tikzcalculator}[1]{%
699   \ifcsdef{tikzcalculator@#1}{%
700     \csuse{tikzcalculator@#1}%
701   }{%
702     \ClassError{graph35}{Unknown calculator '#1'}{Look for the list of available calculators in
703   }%
704 }

Define macro to draw the Graph35+E calculator.
705 \csdef{tikzcalculator@graph35+E}{%
706   %\node{\includegraphics[scale=.59]{graph35-original}};
707   \draw[color=graph35CASEBORDER, fill=graph35CASE, line width=2pt] plot [smooth cycle] coordi
708     (-3.2, 0)
709     (-3.35, 5.5)
710     (-3.2, 7.0)
711     (-2.5, 7.3)
712     (0, 7.4)
713     (2.5, 7.3)
714     (3.2, 7.0)
715     (3.35, 5.5)
716     (3.2, 0)
717     (3.1, -5)
718     (3.0, -6.6)
719     (2.3, -7.3)
720     (0, -7.4)
721     (-2.3, -7.3)
722     (-3.0, -6.6)
723     (-3.1, -5)
724   };
725   \node[minimum width=6.7cm, minimum height=14.8cm, inner sep=0pt] (case) at (0, 0) {};
726   \draw[draw=graph35SCREEN, line width=1.5pt, fill=graph35SCREENBG, rounded corners=5](-
2.64, 3) rectangle (2.58, 6.035);
727   \node[minimum width=5.22cm, minimum height=3.035cm, inner sep=0pt] (screen) at (-
.003, 4.5175) {};
728   \begin{scope}[shift={(-2.5, 5.8)}, scale=.039]
729     \foreach \i in {0, 3, ..., 33, 94, 97, ..., 103, 121, 124, ..., 127} {
730       \foreach \j in {1, 4, 7} {
731         \fill (\i, {-\j}) rectangle ++(1, 1);
732       }

```

```

733     }
734     \foreach \i in {2, 5, ..., 32, 96, 99, ..., 105, 123, 126, ..., 126} {
735         \foreach \j in {2, 5} {
736             \fill (\i, {-\j}) rectangle ++(1, 1);
737         }
738     }
739     \foreach \i in {1, 4, ..., 31, 95, 98, ..., 104, 122, 125, ..., 125} {
740         \foreach \j in {3, 6} {
741             \fill (\i, {-\j}) rectangle ++(1, 1);
742         }
743     }
744     \tikzbpixelart{(38, 0)}{%
745         1000100111000111001000100000010001011111010001010001
746         1101101000100010001000100000011011010000010001010001
747         1010101000100010001100100000010101010000011001010001
748         1010101111100010001010100000010101011110010101010001
749         1000101000100010001001100000010001010000010011010001
750         1000101000100010001000100000010001010000010001010001
751         1000101000100111001000100000010001011111010001001110
752     }
753     \tikzbattery{high}{(107, 0)}
754     % Row 1
755     \tikzmenu{RUNMAT}{1}{(0, -8)}
756     \tikzmenu{STAT}{2}{(30, -8)}
757     \tikzmenu{GRAPH}{3}{(60, -8)}
758     \tikzmenu{DYNA}{4}{(90, -8)}
759     % Row 2
760     \tikzmenu{TABLE}{5}{(0, -27)}
761     \tikzmenu{RECUR}{6}{(30, -27)}
762     \tikzmenu{CONICS}{7}{(60, -27)}
763     \tikzmenu{EQUA}{8}{(90, -27)}
764     % Row 3
765     \tikzmenu{PRGM}{9}{(0, -46)}
766     \tikzmenu{TVM}{A}{(30, -46)}
767     \tikzmenu{ECON2}{B}{(60, -46)}
768     \tikzmenu{LINK}{C}{(90, -46)}
769     % Right arrow
770     \tikzbpixelart{(121, -58)}{%
771         00100
772         00100
773         00100
774         00100
775         10101
776         01110
777         00100
778     }
779     \end{scope}
780
781     \tikzkey[shift, alpha]{REPLAY}{(1.97, -.055)}
782

```

```

783 \tikzkey[shift, alpha]{SHIFT}{(-2.39, .23)}
784 \tikzkey[shift, alpha]{OPTN}{(-1.43, .23)}
785 \tikzkey[shift, alpha]{VARS}{(-.47, .23)}
786 \tikzkey[shift, alpha]{MENU}{(.47, .23)}
787 \tikzkey[shift, alpha]{ALPHA}{(-2.39, -.61)}
788 \tikzkey[shift, alpha]{square}{(-1.43, -.61)}
789 \tikzkey[shift, alpha]{power}{(-.47, -.61)}
790 \tikzkey[shift, alpha]{EXIT}{(.47, -.61)}
791 \tikzkey[shift, alpha]{XthetaT}{(-2.39, -1.43)}
792 \tikzkey[shift, alpha]{log}{(-1.43, -1.43)}
793 \tikzkey[shift, alpha]{ln}{(-.47, -1.43)}
794 \tikzkey[shift, alpha]{sin}{(.47, -1.43)}
795 \tikzkey[shift, alpha]{cos}{(1.43, -1.43)}
796 \tikzkey[shift, alpha]{tan}{(2.39, -1.43)}
797 \tikzkey[shift, alpha]{fraction}{(-2.39, -2.25)}
798 \tikzkey[shift, alpha]{FD}{(-1.43, -2.25)}
799 \tikzkey[shift, alpha]{openparen}{(-.47, -2.25)}
800 \tikzkey[shift, alpha]{closeparen}{(.47, -2.25)}
801 \tikzkey[shift, alpha]{comma}{(1.43, -2.25)}
802 \tikzkey[shift, alpha]{rightarrow}{(2.39, -2.25)}
803
804 \tikzkey[shift, alpha]{DEL}{(1.16, -3.25)}
805 \tikzkey[shift, alpha]{ACON}{(2.32, -3.25)}
806 \tikzkey[shift, alpha]{times}{(1.16, -4.22)}
807 \tikzkey[shift, alpha]{divide}{(2.32, -4.22)}
808 \tikzkey[shift, alpha]{plus}{(1.16, -5.19)}
809 \tikzkey[shift, alpha]{minus}{(2.32, -5.19)}
810 \tikzkey[shift, alpha]{dot}{(-1.15, -6.159)}
811 \tikzkey[shift, alpha]{opposite}{(1.16, -6.159)}
812 \tikzkey[shift, alpha]{zero}{(-2.294, -6.159)}
813 \tikzkey[shift, alpha]{1}{(-2.294, -5.19)}
814 \tikzkey[shift, alpha]{2}{(-1.15, -5.19)}
815 \tikzkey[shift, alpha]{3}{(.01, -5.19)}
816 \tikzkey[shift, alpha]{4}{(-2.294, -4.22)}
817 \tikzkey[shift, alpha]{5}{(-1.15, -4.22)}
818 \tikzkey[shift, alpha]{6}{(.01, -4.22)}
819 \tikzkey[shift, alpha]{7}{(-2.294, -3.25)}
820 \tikzkey[shift, alpha]{8}{(-1.15, -3.25)}
821 \tikzkey[shift, alpha]{9}{(.01, -3.25)}
822 \tikzkey[shift, alpha]{10}{(.01, -6.159)}
823 \tikzkey[shift, alpha]{EXE}{(2.32, -6.159)}
824
825 \tikzkey[shift, alpha]{F1}{(-2.37, 1.5)}
826 \tikzkey[shift, alpha]{F2}{(-1.43, 1.42)}
827 \tikzkey[shift, alpha]{F3}{(-.45, 1.38)}
828 \tikzkey[shift, alpha]{F4}{(.45, 1.38)}
829 \tikzkey[shift, alpha]{F5}{(1.43, 1.42)}
830 \tikzkey[shift, alpha]{F6}{(2.37, 1.5)}
831 }

```

## Table des figures

1	Calculatrice <code>graph35+E</code> . . . . .	10
2	Ancres des touches . . . . .	11
3	Ancres de la touche <code>REPLAY</code> . . . . .	11
4	Ancres de l'écran . . . . .	12
5	Ancres du boîtier . . . . .	13
6	Mots-clefs des touches . . . . .	27

## Change History

v0.1.0

General : First published version. 46

## Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

<b>B</b>	<b>S</b>	<b>T</b>
<code>\battery</code> . . . . . <u>84</u>	<code>\setgraphcolor</code> . . . . .	<code>\tikzbattery</code> 86, <u>89</u> , 753
<b>C</b>	. . <u>69</u> , 230, 266,	<code>\tikzcalculator</code> 694, <u>698</u>
<code>\calculator</code> . . . . . <u>690</u>	270, 306, 310,	<code>\tikzfunction</code> . . 98, <u>101</u>
<b>F</b>	336, 340, 355,	<code>\tikzkey</code> . . 195, <u>198</u> ,
<code>\function</code> . . . . . <u>96</u>	359, 379, 383,	781, 783–802,
<b>K</b>	409, 413, 449,	804–823, 825–830
<code>\key</code> . . . . . <u>192</u>	453, 489, 493,	<code>\tikzmenu</code> . . . . . 74,
<b>M</b>	520, 524, 551,	<u>77</u> , 755–758,
<code>\menu</code> . . . . . <u>72</u>	555, 582, 586,	760–763, 765–768
	613, 617, 644,	
	648, 675, 692, 696	