

Graph35*

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

Louis Paternault
spalax(at)gresille(dot)org

29 novembre 2022

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

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

*Ce document décrit le paquet `graph35` v0.1.3, publié le 2022/11/29. Site web, tickets de suivi, etc. sur <http://framagit.org/spalax/graph35>.

B Ancres	9
B.1 Ancres des touches	9
B.2 Ancres de la touche REPLAY	9
B.3 Ancres de l'écran	9
B.4 Ancres du boîtier	9
C Pixel art	11
C.1 Menu	11
C.2 Fonctions	13
C.3 Batterie	24
D Touches	24
E Implementation	24
Liste des figures	42
Historique des changements	42
Index	43

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^AT_EX

Ce paquet est disponible à la fois dans les distributions T_EXLive et MiK_TE_X. 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=main>

— Décompresser l'archive.

— Compiler le fichier : `latex graph35.ins`

— Déplacer les fichiers `.sty` dans un répertoire du chemin de L^AT_EX.

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}{pixelart0}` 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 temps¹.

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 en plus une valeur `color=default`, pour spécifier explicitement l'utilisation de la couleur par défaut.

`\setgraphcolor` — 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}`.

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.

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.

`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.
`graph35SHIFTTEXT` : 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{graph35SHIFTTEXT}{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 ancrs (*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 :


$$\backslash\text{key}[\langle\text{color}, \text{prefix}, \text{suffix}, \text{scale}, \text{shift}, \text{alpha}\rangle]\{\langle\text{touche}\rangle\}.$$

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

Les arguments sont :

- $\{\langle\text{touche}\rangle\}$ 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.
- $[\langle\text{color}, \text{scale}\rangle]$ 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).
- $[\langle\text{shift}, \text{alpha}\rangle]$ 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`.
- $[\langle\text{prefix}, \text{suffix}\rangle]$ Des ancrs (*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 ancrs 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 ancrs définies pour les textes SHIFT et ALPHA.

 Sans options : ancrs `keyDEL`, `keyDELshift`, `keyDELalpha`.

 Avec options `prefix=tagada`, `suffix=tsoin` :
ancres `tagadaDELtsoin`, `tagadaDELtsoinshift`,
`tagadaDELtsoinalpha`.

Ces ancrs 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'un environnement TikZ. Sa signature est :

`\tikzkey[options]{touche}{coordonnées}`


Ses arguments sont :

- [*options*] les mêmes options que pour la commande `\key` ;
- *touche* le nom de la touche à dessiner ;
- *coordonnées* 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{icône}{raccourci}` 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}{coordonnée}` 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 `pixelart0` ; 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 à  à  (par exemple  ou ). 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 ). 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*.



FIGURE 1 – Calculatrice graph35+E.

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.

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.

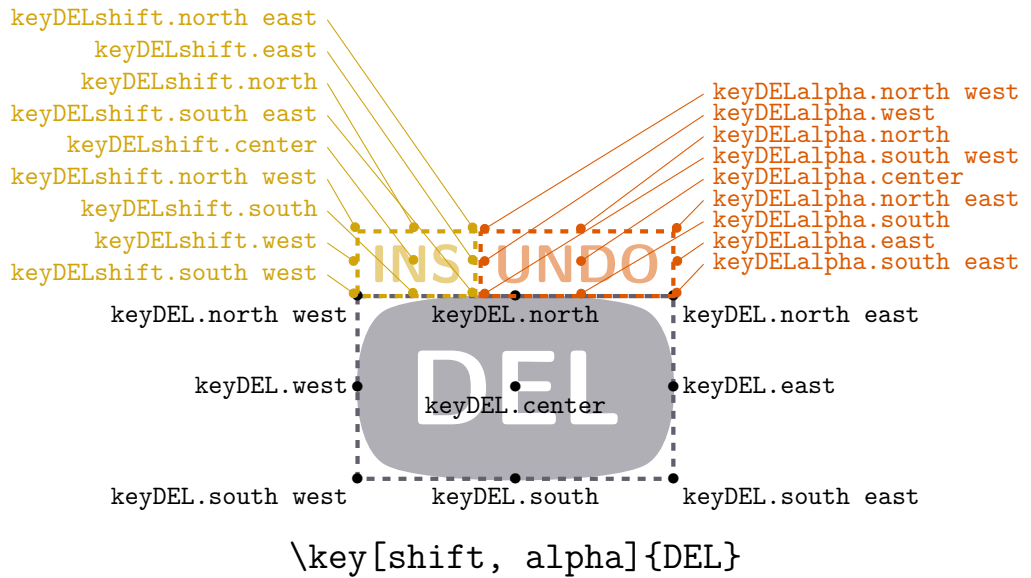


FIGURE 2 – Ancres des touches

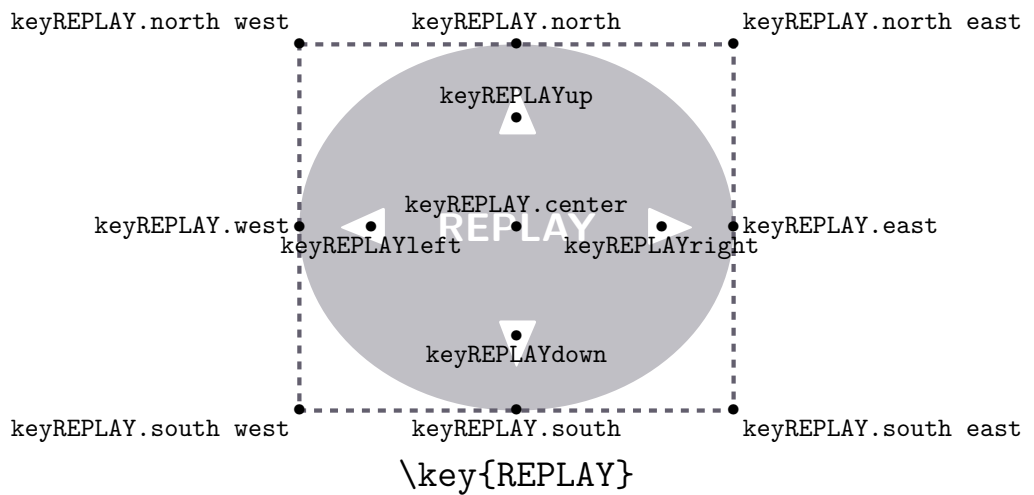


FIGURE 3 – Ancres de la touche REPLAY

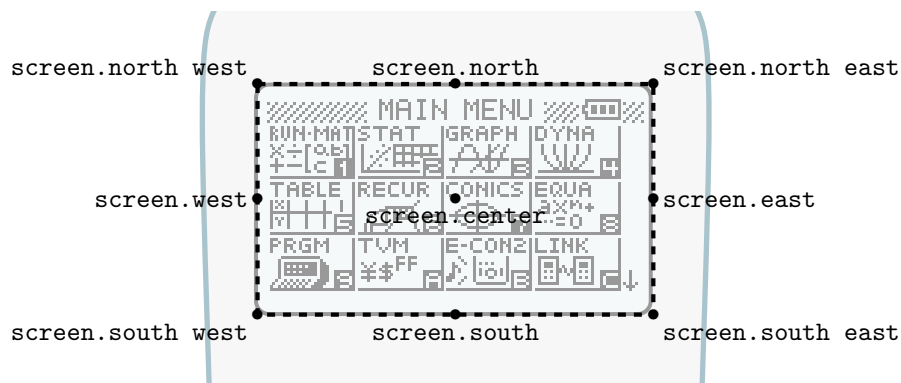























FIGURE 4 – Ancres de l'écran

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.

C.1.1 Icônes

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

C.1.2 Raccourcis

—  <code>\menu{black}{1}</code>	—  <code>\menu{black}{6}</code>
—  <code>\menu{black}{2}</code>	—  <code>\menu{black}{7}</code>
—  <code>\menu{black}{3}</code>	—  <code>\menu{black}{8}</code>
—  <code>\menu{black}{4}</code>	—  <code>\menu{black}{9}</code>
—  <code>\menu{black}{5}</code>	—  <code>\menu{black}{A}</code>

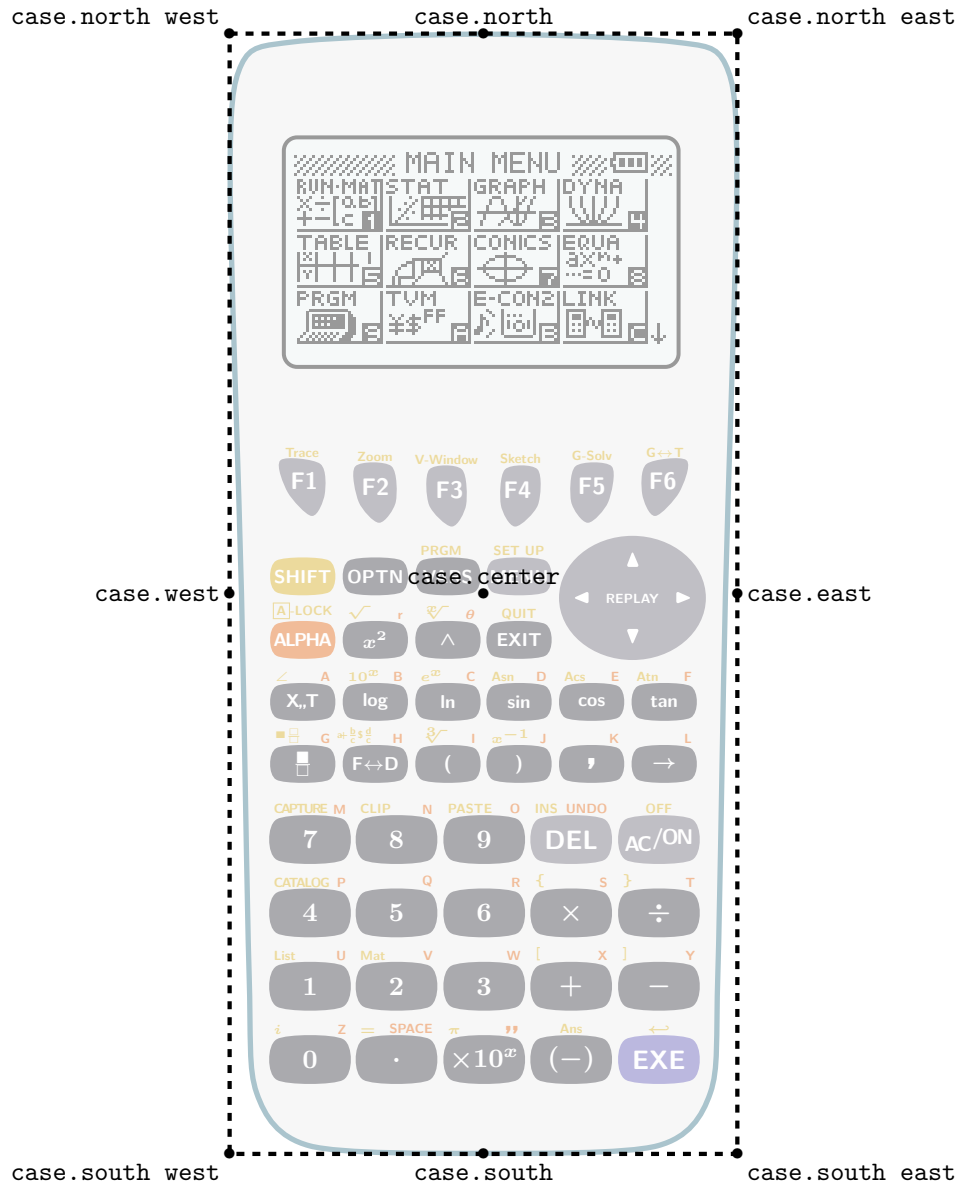


FIGURE 5 – Ancres du boîtier

	<code>\menu{black}{B}</code>		<code>\menu{black}{E}</code>
	<code>\menu{black}{black}</code>		<code>\menu{black}{F}</code>
	<code>\menu{black}{blank}</code>		<code>\menu{black}{G}</code>
	<code>\menu{black}{C}</code>		<code>\menu{black}{H}</code>
	<code>\menu{black}{D}</code>		

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.

<i>Vide</i>		question-b	200
		quote-b	200
blank		rightarrow	21
		Sigma-b	2x1
		square-b	22
		style1	2x2
		style2	2p
		style3	2P
		style4	2s
		style5	2S
		style6	2var
		style7	2VAR
		tilde-b	2VAR-b
b	1		2way
		1	2WAY
b	10		3
		10	3-b
		100	31
		100	3x1
	1p		33
		1P	3x3
	1s		38k
		1S	to38k
	1var		3pin
		1VAR	3PIN
		1VAR-b	4
	2		4-b
next		2	5
nextb		2-b	5-b

6	$\overline{6}$ 6-b	\overline{Sac} Sac	\overline{anPl} anPl-b
60	$\overline{60}$ 60	acn	anst
7400	$\overline{7400}$ 7400	\overline{Sacn} Sacn-b	\overline{anSt} anSt-b
9850	$\overline{9850}$ 9850	add	apl
9860	$\overline{9860}$ 9860	\overline{ADD} ADD	\overline{SaPl} SaPl-b
a	\overline{a} a-b	\overline{ADD} ADD-b	app
a0	$\overline{a0}$ a0	adf	\overline{APP} APP-b
	$\overline{a0-b}$ a0-b	\overline{Adf} Adf-b	apr
a1	$\overline{a1}$ a1	adv	\overline{APR} APR-b
	$\overline{a1-b}$ a1-b	\overline{ADV} ADV-b	\overline{tAPR} tAPR
a2	$\overline{a2}$ a2-b	aebx	area
aa	\overline{Aa} Aa	\overline{aebx} aebx	\overline{AREA} AREA-b
ab	\overline{ab} ab	$\overline{aebx-b}$ aebx-b	arg
	\overline{Sab} Sab	all	\overline{Arg} Arg-b
abc	\overline{ABC} ABC	\overline{ALL} ALL	as
abdf	\overline{ABdf} ABdf-b	\overline{ALL} ALL-b	\overline{AandS} AandS-b
abi	$\overline{tcomplexalgebraic-b}$ tcomplexalgebraic-b	always	asgn
abs	\overline{Abs} Abs-b	\overline{Alway} Alway	\overline{ASGN} ASGN
abt	\overline{ABT} ABT	amt	aug
abx	$\overline{a+bx}$ aplusbx	\overline{AMT} AMT-b	\overline{Aug} Aug-b
	$\overline{a+bx-b}$ aplusbx-b	an	auto
	$\overline{a*bx}$ atimesbx	\overline{an} an	\overline{AUTO} AUTO
	$\overline{a*bx-b}$ atimesbx-b	$\overline{an-b}$ an-b	\overline{Auto} Auto
ac	$\overline{a*c}$ ac	\overline{San} San	$\overline{Auto-2}$ Auto-2
		$\overline{San-b}$ San-b	$\overline{Auto-b}$ Auto-b
		an1	axb
		$\overline{an1}$ an1	$\overline{ax+b}$ axplusb
		$\overline{an1-b}$ an1-b	$\overline{ax+b-b}$ axplusb-b
		$\overline{an1-b2}$ an1-b2	b
		$\overline{San1-b}$ San1-b	\overline{b} b-b
		an2	b0
		$\overline{an+2}$ an2	$\overline{b0}$ b0-b
		$\overline{an+2}$ an2-b	b1
		$\overline{San2-b}$ San2-b	$\overline{b1}$ b1-b
		ancn	b2
		\overline{ancn} ancn-b	$\overline{b2}$ b2-b
		and	bal
		\overline{And} And-b	\overline{BAL} BAL
		angl	\overline{BAL} BAL-b
		\overline{ANGL} ANGL-b	bar
		anov	\overline{Bar} Bar-b
		\overline{ANOV} ANOV	base
		anpl	

BASE BASE	C c-b	CHNG CHNG
bc	c0	close
BC bc	C0 C0-b	CLOSE Close-b
SBC Sbc	c1	clr
bcd	C1 C1-b	CLR CLR
BCD Bcd	c2	CLR CLR-b
bdf	C2 C2-b	cls
BDF Bdf-b	cabl	CLS cls
bin	CABL CABL-b	CLS Cls-b
BIN BIN-b	calb	cmā
Bin Bin-b	CALB CALB-b	CMA CMA-b
binm	calc	cmp
BINM BINM-b	CALC CALC	CMF Cmp-b
bkup	CALC CALC-b	cmpd
BKUP BKVP-b	calib	CMPD CMPD-b
bn	CALIB CALIB	cmpr
Bn bn-b	capa	CMPR CMPR-b
Sbn Sbn-b	CAPA CAPA-b	cn
bn1	capt	Cn cn-b
Bn1 bn1-b	CAPT capt	Scn Scn-b
Sbn1 Sbn1-b	CAPT CAPT-b	cn1
bn2	cash	Cn1 cn1-b
Bn2 bn2-b	CASH CASH-b	Scn1 Scn1-b
Sbn2 Sbn2-b	casio	cn2
bnst	CASIO CASIO-b	Cn2 cn2-b
BnSt bnSt-b	ccd	Scn2 Scn2-b
bond	CCD Ccd	cnst
BOND BOND-b	cel	CnSt CnSt-b
bot	CEL CEL-b	cnt
BOT BOTbottom	cell	CNT cnt
BOT BOTright	CELL CELL	cnvt
box	ch1	CNVT CNVT-b
BOX BOX	CH1 CH1	col
Box Box-b	char	COL COL
bpd	CHAR CHAR-b	COL COL-b
BPD Bpd	chg	com
brk	CHG Chg-b	COM COM-b
Brk Brk-b	chi	conj
brkn	CHI CHI	CONJ Conj-b
Brkn Brkn-b	CHI CHI-b	conv
btm	Chi Chi-b	CONV CONV-b
BTM BTM	chng	copy
c		COPY COPY
		COPY COPY-b
		cosh

COSH cosh-b	DB DB	DrwC DrwC-b
cosh1	ddt	drwf
COSH1 cosh1-b	DDT ddt	DrwF DrwF-b
cost	ddx	drwn
COST COST	DDX ddx-b	DrwN DrwN-b
COST COST-b	defg	drwt
Cost Cost-b	DEFG DefG-b	Drwt Drwt-b
cpd	del	dsz
Cpd Cpd	DEL DEL	Dsz Dsz-b
cplx	DEL DEL-b	dx
CPLX CPLX-b	dela	Idx Idx
crcl	DELA DELA-b	Idx Idx-b
Crcl Crcl	dell	dyna
Crcl Crcl-b	DELL DELL-b	DYNA DYNA-b
crnt	depr	Dyna Dyna-b
CRNT CRNT-b	DEPR DEPR-b	e
cstm	det	E e-b
CSTM CSTM-b	DET Det-b	E Exa-b
ctgy	df	edf
CTGY CTGY-b	Df df-b	Edf Edf-b
ctl	diff	edit
CTL CTL-b	DIFF diff	EDIT EDIT
cuml	dim	EDIT EDIT-b
CUM1 Cum1-b	DIM DIM-b	eff
cut	Dim Dim-b	EFF EFF-b
CUT CUT	disp	TEFF tEFF
cy	DISP DISP-b	else
CY CY-b	dist	ELSE Else-b
d	DIST DIST-b	end
D d-b	dld	END End-b
d2dt2	dlminusD	eng
D2DT2 d2dt2	dlplusD	ENG ENGshiftleft
d2dx2	dms	ENG ENGshiftright
D2DX2 d2dx2-b	tDMS tDMS-b	engy
data	do	ENGY ENGY-b
DATA DATA-b	DO Do-b	entr
Data Data-b	dot	ENTR ENTR-b
percentDATA- b	DOT dot-b	equa
days	draw	EQUA EQUA-b
DAYS DAYS-b	DRAW DRAW	es
db	DRAW DRAW-b	ES EtS-b
	drwc	esym

ESYM ESYM-b
 exam
EXAM EXAM-b
 exe
EXE EXE
 exit
EXIT EXIT
EXIT EXIT-b
 exp
Exp Exp
EXP EXP-b
Exp Exp-b
Exp Exp-b2
 extd
Ext Ext
 f
F F
F F-b
F F-b2
f femto-b
 fa
Fa Fa-b
 fab
Fab Fab-b
 fact
FACT FACT-b
Fact Fact-b
 fast
Fast Fast
 fb
Fb Fb-b
 fcd
Fcd Fcd
 file
FILE FILE-b
 fill
FILL FILL-b
Fill Fill-b
 fline
FLine FLine
FLine FLine-b
 fmax
FMax FMax-b

fmin
FMin FMin-b
 for
FOR For-b
 forc
FORC FORC-b
 form
FORM FORM
FORM FORM-b
 fp
FP FP
FP FP-b
 fpd
Fpd Fpd
 frac
Frac Frac-b
 ftbl
FTbl FTbl-b
 full
FULL FULL
 furie
Furie Furie
 fv
FV FV
FV FV-b
 g
g g-b
G Giga-b
 gcd
Gcd Gcd
GCD GCD-b
 gcon
GCON GCON
Gcon Gcon-b
 gdx
GIdx GIdx-b
 geo
GEO GEO-b
 gmem
GMEM GMEM-b
 go
GO GO

gof
GOF GOF
 goto
Goto Goto-b
 gpd
Gpd Gpd
 gph1
GPH1 GPH1
GPH1 GPH1-b
 gph2
GPH2 GPH2
GPH2 GPH2-b
 gph3
GPH3 GPH3
GPH3 GPH3-b
 gplt
GPLT GPLT
GPlt GPlt-b
 grab
GRAB GRAB
 grph
GRPH GRPH
GRPH GRPH-b
GRPH Grph-b
 gslv
GSLV GSLV-b
 gtky
Gtky Gtky-b
 hcd
Hcd Hcd
 help
HELP HELP-b
 hgeo
HGEO HGEO-b
 hist
HIST Hist-b
 hpd
Hpd Hpd
 hyp
HYP HYP-b
 hztl
HZTL Hztl
HZTL Hztl-b

i	INV InvF	LGST Lgst
I i-b	invg	LGST Lgst-b
IP% Ipercent	INVG InvG	line
IP% Ipercent-b	invh	LINE Line
iden	INVH InvH	LINE LINE-b
IDEN Iden-b	invn	LINE Line-b
iend	INVN InvN	list
IEND IEnd-b	invp	LIST List
if	INVP InvP	LIST LIST-b
IF If-b	invt	LIST List-b
imp	INVT InvT	LIST tLIST-b
IMP Imp-b	io	lm
in	IO IO-b	LTM LtoM-b
IN IN	irr	lmem
init	IRR IRR	LMEM LMEM-b
INIT INIT	IRR IRR-b	load
inpt	isct	LOAD LOAD-b
INPT INPT-b	ISCT ISCT	log
input	isz	LOG Log
INPUT INPUT	ISZ Isz-b	LOG Log-b
ins	join	logab
INS INS	JOIN Join-b	LOGAB logab-b
INS INS-b	jump	logic
int	JUMP JUMP-b	LOGIC LOGIC-b
INT INT	k	lpw
INT INT-b	K kilo-b	LPW LpW-b
INT Int-b	lang	lwr
INT Intdiv-b	LANG LANG-b	LWR Lwr-b
SINT SINT	lbl	m
SINT SINT-b	LBL Lbl-b	M Mega-b
intg	lcm	m milli-b
INTG INTG	LCM LCM-b	main
INTG Intg-b	lcte	MAIN MAIN-b
intr	LCTE Lcte-b	man
INTR INTR-b	left	MAN Man
inv	LEFT Left-b	mark
INV Inv	len	MARK MARK-b
INV Inv-b	LEN Len-b	mass
invb	leng	MASS MASS-b
INVB InvB	LENG LENG-b	mat
invc	LENG Leng-b	MAT MAT-b
INVC InvC	lgst	MAT Mat-b
invf		MAT 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
 MtL MtoL-b
 mlti
 MLTI MLTI
 mn
 mn mxn-b
 mod
 MOD MOD-b
 Mod Mod-b

mode
 MODE MODE-b
 MODE MODExp-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
 NAME NAME-b
 nan
 Nan Nan-b
 ncd
 Ncd Ncd
 ncr
 nCr nCr-b
 ndis
 NDis NDis-b
 new
 NEW NEW-b
 next
 Next Next-b

nfv
 NFV NFV
 NFV NFV-b
 no
 NO NO
 none
 None None
 None None-b
 norm
 Norm Norm
 NORM NORM-b
 Norm Norm-b
 not
 Not Not-b
 npd
 Npd Npd
 npp
 NPP NPP-b
 npr
 nPr nPr-b
 npv
 NPV NPV
 NPV NPV-b
 num
 NUM NUM-b
 off
 Off Off
 Off Off-b
 on
 On On
 On On-b
 open
 OPEN OPEN-b
 Open Open-b
 opt
 OPT OPT
 OPT OPT-b
 or
 Or Or-b
 orig
 ORIG ORIG
 out
 OUT OUT
 p
 P P
 P p-b

P Peta-b
P phat-b
P pico-b
Pi Psnd-b

p1
P1 phat1-b

p2
P2 phat2-b

pa
Pa pa-b

pab
Pab pab-b

parm
PARM PARM
Parm parm
Parm Parm-b

pb
Pb pb-b

pbp
PBP PBP
PBP PBP-b

pcd
Pcd Pcd

pen
PEN PEN

pgdn
PgDn PgDn

pgup
PgUp PgUp

phas
PHAS PHAS

phase
Phase Phase-b

pie
Pie Pie-b

pitch
Pitch Pitch-b

pixl
PIXL PIXL-b

plchg
P1Chg P1Chg
P1Chg P1Chg-b

ploff
P1Off P1Off
P1Off P1Off-b

plon
P1On P1On
P1On P1On-b

plot
Plot Plot
PLOT PLOT-b
Plot Plot-b

pmt
PMT PMT
PMT PMT-b

poisn
POISN POISN-b

pol
POL POL
Pol Pol-b

poly
POLY POLY-b

ppd
Ppd Ppd

prc
PRC PRC
PRC PRC-b

prd
PRD PRD
PRD PRD-b

pre
PRE PRE

pres
PRES PRES-b

prn
PRN PRN
PRN PRN-b
SPRN SPRN
SPRN SPRN-b

prob
PROB PROB-b

prod
Prod Prod-b

prog
PROG PROG-b
Prog Prog-b

proj
Proj Proj

ptch
Ptch Ptch-b

pts
PTS PTS-b

pv
PV PV
PV PV-b

pwr
Pwr Pwr
PWR PWR-b
Pwr Pwr-b

py
PY PY-b

q
Q Qsnd-b

q1
Q1 Q1-b

q3
Q3 Q3-b

r
r r-b
r r-b2
r r-b3
r requal
r requal-b
R Rsnd-b
r tcomplexpolar-b

r2
r2 r2-b

r38k
R38k R38k-b

ran
Ran Ran-b

rand
RAND RAND-b

rang
RANG RANG-b

rcl
RCL RCL
RCL RCL-b

REL Rcl-b
 rdel **RDEL** RDEL
 rec **REC** Rec-b
 recal **RECAL** RECAL
 recr **RECR** RECR-b
 rect **RECT** RECT
 recv **RECV** RECV
Recv Recv
RECV Recv-b
 ref **REF** Ref-b
 reg **REG** REG
REG REG-b
 rel **REL** REL-b
 ren **REN** REN-b
 rep **REP** Rep-b
 rept **REPT** REPT
 reslt **RESLT** RESLT-b
Reslt Reslt-b
 right **RIGHT** Right-b
 rmdr **RMDR** Rmdr-b
 rnd **RND** RND
RND Rnd-b
 rndfi **RNDFI** RndFi-b
 rnf **RNF** RNF-b

root **ROOT** ROOT
 rop **ROP** ROP-b
 rot **ROT** Rot-b
 row **ROW** ROW
ROW ROW-b
 rref **RREF** Rref-b
 rset **RSET** RSET-b
 rt **RT** RT
RT RTtheta-b
 rtbl **RTBL** RTbl-b
 rtn **RTRN** Rtrn-b
 run **RUN** RUN
 rw **RW** Rwplus
 rx **RX** RX-b
 ry **RY** RY-b
 s38k **S38K** S38k-b
 save **SAVE** SAVE-b
 scal **SCAL** scal-b
 scat **SCAT** Scat-b
 sd **SD** SD-b
 sdev **SDEV** SDev-b
 se **SE** se-b

sel **SEL** SEL
SEL SEL-b
 sell **SELL** Sell-b
 sels **SELS** SELS-b
 send **SEND** Send-b
 seq **SEQ** SEQ-b
SEQ seq-b
 set **SET** SET-b
 sfv **SFV** SFV
SFV SFV-b
SFV SFV-b2
 shift **SHIFT** Shift-b
 si **SI** SI
SI SI-b
 siml **SIML** SIML-b
 simp **SIMP** Simp-b
SIMP Simp-b2
 sin **SIN** Sin
SIN Sin-b
 sinh **SINH** sinh-b
 sinh1 **SINH1** sinh1-b
 size **SIZE** SIZE-b
 sktch **SKTCH** SKTCH-b
 sl **SL** SL
 smem **SMEM** SMEM-b

smpl	SMPL SMPL-b	stick	STICK STICK-b	t	t-b2
snd	SND Snd	sto	STO STO-b Sto Sto-b	T	Tera-b
solv	SOLV SOLV SOLV SOLV-b	stop	STOP STOP Stop Stop-b	tsnd	tsnd-b
solve	Solve Solve	str	STR STR STR STR-b Str Str-b	Ttheta	Ttheta-b
solvN	SolvN SolvN-b	strp	STRP STRP-b	tabl	TABL TABL TABL TABL-b Tabl Tabl-b
sonic	Sonic sonic	strt	STRT STRT Strt Strt-b	tang	Tang Tang Tang Tang-b
sp	SP sp-b	stup	STUP STUP-b	tanh	TANH tanh-b
sqr	SQR SQR	styl	STYL STYL-b	tanh1	TANH1 tanh1-b
src	SRC SRC SRC SRC-b Src Src-b	sum	SUM Sum-b	tcd	TCD tcd
srtc	SRTC SRTC Srtc Srtc-b	svas	SVAS SVAS-b	test	TEST TEST-b Test Test-b
srtA	SRTA SRTA SrtA SrtA-b	swap	SWAP SWAP	text	TEXT TEXT Text Text Text Text-b
srtD	SRTD SRTD SrtD SrtD-b	sx	SX sx-b	then	THEN Then-b
ssa	SSA SSa-b	sx1	SX1 sx1-b	time	TIME TIME-b
ssab	SSAB SSab-b	sx2	SX2 sx2-b	tlow	TLOW tLow-b
ssb	SSB SSb-b	sy	SY sy-b	tmpr	TMPR TMPR-b
sse	SSE SSe-b	sybl	SYBL SYBL SYBL SYBL-b	to	TO To-b
stat	STAT STAT-b Stat Stat-b	syd	SYD SYD	tool	TOOL TOOL-b
std	STD STD	t	T T t t-b	top	TOP TOP TOP TOPleft TOP TOPtop
step	STEP Step-b			tpd	TPD tpd
				tran	

$\overline{\text{TRAN}}$ TRAN
 $\overline{\text{TRAN}}$ TRAN-b
 trig
 $\overline{\text{TRIG}}$ TRIG
 trn
 $\overline{\text{Trn}}$ Trn-b
 tup
 $\overline{\text{tUp}}$ tUp-b
 tvn
 $\overline{\text{TVM}}$ TVM-b
 type
 $\overline{\text{TYPE}}$ TYPE-b
 unit
 $\overline{\text{UNIT}}$ UNIT-b
 upr
 $\overline{\text{Upr}}$ Upr-b
 usb
 $\overline{\text{USB}}$ USB
 var
 $\overline{\text{var}}$ var
 $\overline{\text{VAR}}$ VAR-b
 $\overline{\text{Var}}$ Var-b
 vct
 $\overline{\text{VCT}}$ VCT-b
 velo
 $\overline{\text{VELO}}$ VELO-b
 ver
 $\overline{\text{VER}}$ VER-b
 vert
 $\overline{\text{Vert}}$ Vert
 $\overline{\text{Vert}}$ Vert-b
 vlum
 $\overline{\text{VLUM}}$ VLUM-b
 vnlk
 $\overline{\text{VNLK}}$ VNLK-b
 vrnr
 $\overline{\text{VRNR}}$ VRNR-b
 vwin
 $\overline{\text{VWIN}}$ VWIN-b
 $\overline{\text{VWin}}$ VWin-b
 wake
 $\overline{\text{WAKE}}$ WAKE-b

web
 $\overline{\text{WEB}}$ WEB
 $\overline{\text{Web}}$ Web-b
 wend
 $\overline{\text{WEnd}}$ WEnd-b
 while
 $\overline{\text{Whle}}$ Whle-b
 wiz
 $\overline{\text{WIZ}}$ WIZ-b
 x
 $\overline{\text{x!}}$ factorialx-
 b
 $\overline{\text{sigmax}}$ sigmax-b
 $\overline{\text{Sx}}$ Sx-b
 $\overline{\text{txequal}}$ txequal
 $\overline{\text{txgeq}}$ txgeq
 $\overline{\text{txgt}}$ txgt
 $\overline{\text{txleq}}$ txleq
 $\overline{\text{txlt}}$ txlt
 $\overline{\text{x}}$ x
 $\overline{\text{X}}$ X-b
 $\overline{\text{x}}$ x-b
 $\overline{\text{X}}$ X-b2
 $\overline{\text{X}}$ X-b3
 $\overline{\text{x}}$ xbar-b
 $\overline{\text{xequal}}$ xequal
 $\overline{\text{xequal-b}}$ xequal-b
 $\overline{\text{xgeq-b}}$ xgeq-b
 $\overline{\text{xgt-b}}$ xgt-b
 $\overline{\text{xhat-b}}$ xhat-b
 $\overline{\text{xleq-b}}$ xleq-b
 $\overline{\text{xlt-b}}$ xlt-b
 x1
 $\overline{\text{x1}}$ x1-b
 $\overline{\text{x1}}$ xbar1-b
 x1inv
 $\overline{\text{x1Inv}}$ x1Inv-b
 x2
 $\overline{\text{Sx2}}$ Sx2-b
 $\overline{\text{X2}}$ X2
 $\overline{\text{x2}}$ x2
 $\overline{\text{x2}}$ x2-b
 $\overline{\text{x2}}$ xbar2-b
 $\overline{\text{x2}}$ xpower2-b
 x2inv
 $\overline{\text{x2Inv}}$ x2Inv-b

x3
 $\overline{\text{X3}}$ X3
 $\overline{\text{x3}}$ x3
 $\overline{\text{x3}}$ x3-b
 $\overline{\text{x3}}$ xpower3-b
 x4
 $\overline{\text{X4}}$ X4
 $\overline{\text{x4}}$ x4
 $\overline{\text{x4}}$ xpower4-b
 xcal
 $\overline{\text{XCAL}}$ XCAL
 xfct
 $\overline{\text{Xfct}}$ Xfct-b
 xin
 $\overline{\text{xInv}}$ xInv-b
 xor
 $\overline{\text{Xor}}$ Xor-b
 xrw
 $\overline{\text{XRw}}$ XRw
 $\overline{\text{XRw}}$ XRwplus
 xt
 $\overline{\text{Xt}}$ Xt-b
 xy
 $\overline{\text{Sxy}}$ Sxy-b
 $\overline{\text{xy}}$ xy-b
 y
 $\overline{\text{sigmay}}$ sigmay-b
 $\overline{\text{Sy}}$ Sy-b
 $\overline{\text{tYequal}}$ tYequal
 $\overline{\text{tYgeq}}$ tYgeq
 $\overline{\text{tYgt}}$ tYgt
 $\overline{\text{tYleq}}$ tYleq
 $\overline{\text{tYlt}}$ tYlt
 $\overline{\text{Y}}$ Y
 $\overline{\text{Y}}$ Y-b
 $\overline{\text{Y}}$ Y-b2
 $\overline{\text{ybar-b}}$ ybar-b
 $\overline{\text{Yequal}}$ Yequal
 $\overline{\text{Yequal-b}}$ Yequal-b
 $\overline{\text{Ygeq-b}}$ Ygeq-b
 $\overline{\text{Ygt-b}}$ Ygt-b
 $\overline{\text{yhat-b}}$ yhat-b
 $\overline{\text{Yleq-b}}$ Yleq-b
 $\overline{\text{Ylt-b}}$ Ylt-b
 y1
 $\overline{\text{y1}}$ y1-b

y2	Yfct-b	z-b
Sy2-b	yicpt	zero
y2-b	YICPT	ZERO
y3	yld	zlow
y3-b	YLD	zLow-b
ycaL	YLD-b	zoom
YCAL	yt	ZOOM
yes	Yt-b	ZOOM-b
YES	z	zup
yfct	Z	zUp-b
	Z-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.

— \key{ACON}	— \key{XthetaT}	— \key{4}
— \key{DEL}	— \key{closeparen}	— \key{5}
— \key{ALPHA}	— \key{comma}	— \key{6}
— \key{EXE}	— \key{cos}	— \key{7}
— \key{F5}	— \key{fraction}	— \key{8}
— \key{F4}	— \key{ln}	— \key{9}
— \key{F1}	— \key{log}	— \key{divide}
— \key{F6}	— \key{openparen}	— \key{dot}
— \key{F3}	— \key{power}	— \key{minus}
— \key{F2}	— \key{rightarrow}	— \key{opposite}
— \key{MENU}	— \key{sin}	— \key{plus}
— \key{EXIT}	— \key{square}	— \key{times}
— \key{FD}	— \key{tan}	— \key{zero}
— \key{OPTN}	— \key{1}	— \key{REPLAY}
— \key{VARS}	— \key{10}	— \key{SHIFT}
	— \key{2}	
	— \key{3}	

E Implementation

Load some packages.

```
1 \RequirePackage{etoolbox}
2
```

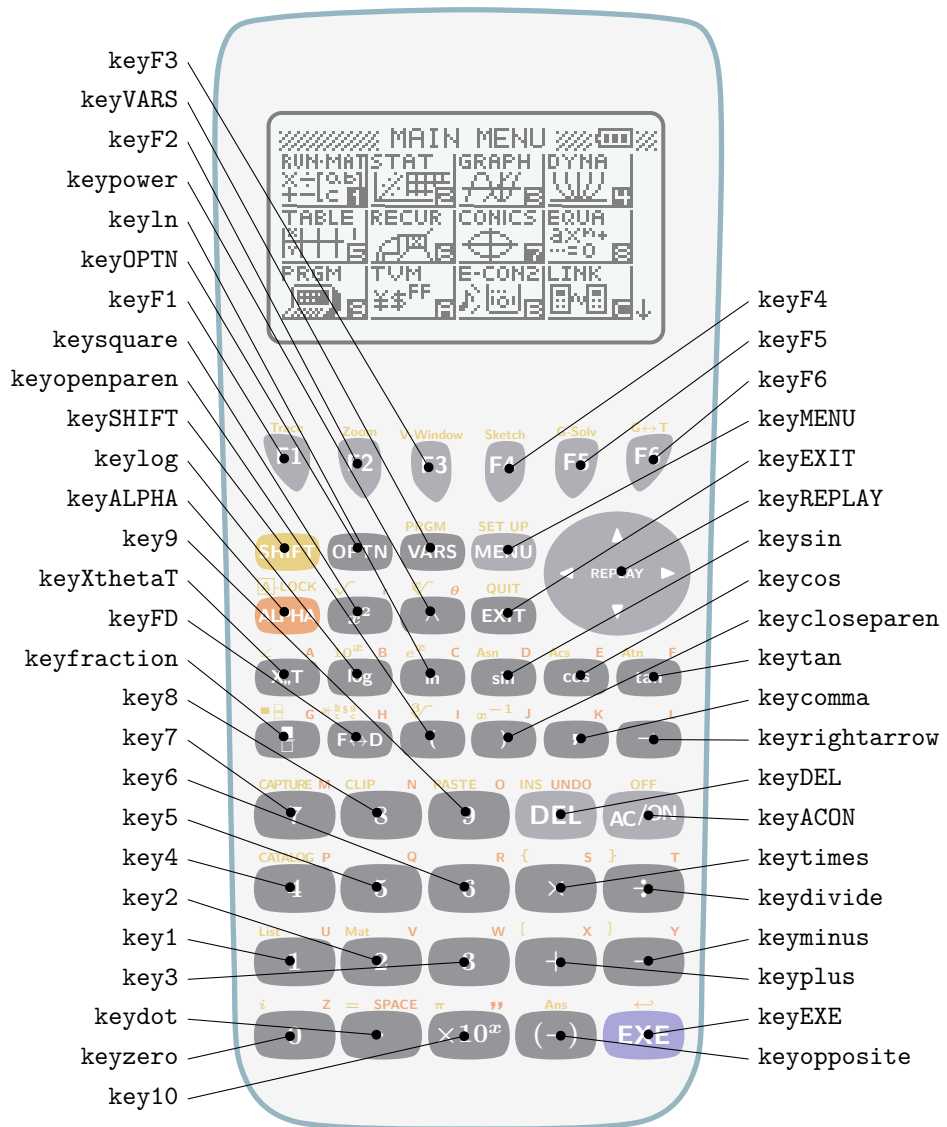



FIGURE 6 – Mots-clefs des touches

```

3 \RequirePackage{pixelart0}
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}
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 packa
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
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 defined 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
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@temp{text}},
174   type/formula/.code={\def\graph@temp{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
203   }%
204 }

```

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

```

205
206 \newcommand{\graph@tikzshifalpha}[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@an
223     }{
224       \node[inner sep=1pt, color=\pgfkeysvalueof{/GRAPH35/SHIFTALPHA/color}, anchor=\graph@an
225     }
226     \else\fi
227 }}

```

Define number keys (e.g. `1`).

```

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
232     \draw[very thick, color=graph35NUMBERBORDER, fill=graph35NUMBER] plot [smooth cycle] coord
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{/GRAPH3
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@tikzshiftalpha{,
246 show=\boolvalue{show@shift},
247 name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
248 position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
249 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
250 color=graph35SHIFTTEXT,
251 type={\pgfkeysvalueof{/GRAPH35/KEY/shift type}},
252 content={\pgfkeysvalueof{/GRAPH35/KEY/shift content}},
253 style={\tiny},
254 }
255 \graph@tikzshiftalpha{,
256 show=\boolvalue{show@alpha},
257 name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
258 position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
259 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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
272 \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina
273 (-.43, .25)
274 (.43, .25)
275 (.43, -.25)
276 (-.43, -.25)
277 ];
278 \node[minimum width=1.03cm, minimum height=.59cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH3
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}\pgfkeysval
288 position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
289 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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}\pgfkeysval
298     position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
299     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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 .

```

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
312     \draw[very thick, color=graph35EXEBORDER, fill=graph35EXE] plot [smooth cycle] coordinate
313     (-.43, .25)
314     (.43, .25)
315     (.43, -.25)
316     (-.43, -.25)
317   };
318   \node[minimum width=1.03cm, minimum height=.59cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH3
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}\pgfkeysval
328     position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
329     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
330     color=graph35SHIFTEXT,
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
342     \draw[very thick, color=graph35SHIFTBORDER, fill=graph35SHIFT] plot [smooth cycle] coordi
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/KEY/cont
349   \ifdefstring{\graph@type}{text}{
350     \node[color=graph35KEYTEXT] {\scriptsize\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/cont
351   }}{
352     \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/cont
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
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/cont
363   .64, 0) {} ;
364   \node[minimum width=0pt, minimum height=0pt, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/cont
365   \node[minimum width=0pt, minimum height=0pt, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/cont
366   .48) {} ;
367   \node[minimum width=1.9cm, minimum height=1.6cm, inner sep=0pt] (\pgfkeysvalueof{/GRAPH35/KEY/cont
368   \ifdefstring{\graph@type}{text}{
369     \node[color=graph35KEYTEXT] {\tiny\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/cont
370   }}{
371     \node[color=graph35KEYTEXT] {\tiny$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$}
372   }}
373   \begin{scope}[rounded corners=1pt, color=graph35KEYTEXT]
374     \fill (.58, .09) -- (.58, -.08) -- (.78, 0) -- cycle ;
375     \fill (-.58, .09) -- (-.58, -.08) -- (-.78, 0) -- cycle ;
376     \fill (-.08, .41) -- (.09, .41) -- (0, .62) -- cycle ;
377     \fill (-.08, -.41) -- (.09, -.41) -- (0, -.62) -- cycle ;
378   \end{scope}
379   \end{scope}
380   \setgraphcolor{\graph@color}%
381 }}

```

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
385   \draw[very thick, color=graph35ALPHABORDER, fill=graph35ALPHA] plot [smooth cycle] coordi
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/cont

```

```

392 \ifdefstring{\graph@type}{text}{
393   \node[color=graph35KEYTEXT] {\scriptsize\bfseries\ssfamly}\pgfkeysvalueof{/GRAPH35/KEY/cont
394 }{
395   \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/cont
396 }
397
398 \graph@tikzshiftalpha{
399   show=\boolvalue{show@shift},
400   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
401   position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
402   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
403   color=graph35SHIFTTEXT,
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
415     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina
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/
422   \ifdefstring{\graph@type}{text}{
423     \node[color=graph35KEYTEXT] {\scriptsize\bfseries\ssfamly}\pgfkeysvalueof{/GRAPH35/KEY/cont
424 }{
425     \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/cont
426 }
427
428 \graph@tikzshiftalpha{
429   show=\boolvalue{show@shift},
430   name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
431   position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
432   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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}\pgfkeysval
441   position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
442   node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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
455     \draw[very thick, color=graph35NUMBERBORDER, fill=graph35NUMBER] plot [smooth cycle] coord
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
462   \ifdefstring{\graph@type}{text}{
463     \node[color=graph35KEYTEXT] {\scriptsize\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KE
464   }}{
465     \node[color=graph35KEYTEXT] {\scriptsize$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/cont
466   }}
467   }
468   \graph@tikzshiftalpha{
469     show=\boolvalue{show@shift},
470     name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
471     position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
472     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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}\pgfkeysval
481     position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
482     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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
495     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina

```

```

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
.03, -.09} {});
503   \ifdefstring{\graph@type}{text}{
504     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/con
505   }}{
506     \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}}$
507   }
508
509   \graph@tikzshiftalpha{
510     show=\boolvalue{show@shift},
511     name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
512     position=center,
513     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
514     color=graph35SHIFTTTEXT,
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
526     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina
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
.005, -.08} {});
534   \ifdefstring{\graph@type}{text}{
535     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/con
536   }}{
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}\pgfkeysval
543     position=center,
544     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
545     color=graph35SHIFTTTEXT,

```

```

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
557     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina
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
565     .04, -.085}) {};
566   \ifdefstring{\graph@type}{text}{
567     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/con
568   }}{
569     \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}}$
570   }
571   \graph@tikzshiftalpha{
572     show=\boolvalue{show@shift},
573     name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
574     position=center,
575     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
576     color=graph35SHIFTEXT,
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
588     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina
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
.085}) {};
596 \ifdefstring{\graph@type}{text}{
597 \node[color=graph35KEYTEXT] {\small\bfseries\sffamily}\pgfkeysvalueof{/GRAPH35/KEY/con
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}\pgfkeysval
605 position=center,
606 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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
619 \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina
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
.08}) {};
627 \ifdefstring{\graph@type}{text}{
628 \node[color=graph35KEYTEXT] {\small\bfseries\sffamily}\pgfkeysvalueof{/GRAPH35/KEY/con
629 }{
630 \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$}
631 }
632
633 \graph@tikzshiftalpha{
634 show=\boolvalue{show@shift},
635 name={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
636 position=center,
637 node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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
650     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordina
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
658     .09) {} ;
659   \ifdefstring{\graph@type}{text}{
660     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/con
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}\pgfkeysval
667     position=center,
668     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysval
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
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] coordinates
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](-
727     2.64, 3) rectangle (2.58, 6.035);
728   \node[minimum width=5.22cm, minimum height=3.035cm, inner sep=0pt] (screen) at (-
729     .003, 4.5175) {};
730   \begin{scope}[shift={(-2.5, 5.8)}, scale=.039]
731     \foreach \i in {0, 3, ..., 33, 94, 97, ..., 103, 121, 124, ..., 127} {
732       \foreach \j in {1, 4, 7} {
733         \fill (\i, {-\j}) rectangle ++(1, 1);
734       }
735     }
736     \foreach \i in {2, 5, ..., 32, 96, 99, ..., 105, 123, 126, ..., 126} {
737       \foreach \j in {2, 5} {
738         \fill (\i, {-\j}) rectangle ++(1, 1);
739       }
740     }
741   \end{scope}

```



```

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 graph35+E	9
2	Ancres des touches	10
3	Ancres de la touche REPLAY	10
4	Ancres de l'écran	11
5	Ancres du boîtier	12
6	Mots-clefs des touches	25

Change History

v0.1.0		paquet utilisé pour dessiner les
General : First published version.	42	menus. 3
v0.1.2		
General : Changement du nom du		

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	270, 306, 310,	792, 793, 794,
\battery	<u>84</u>	336, 340, 355,	795, 796, 797,
		359, 379, 383,	798, 799, 800,
	C	409, 413, 449,	801, 802, 804,
\calculator	<u>690</u>	453, 489, 493,	805, 806, 807,
		520, 524, 551,	808, 809, 810,
	F	555, 582, 586,	811, 812, 813,
\function	<u>96</u>	613, 617, 644,	814, 815, 816,
		648, 675, 692, 696	817, 818, 819,
	K		820, 821, 822,
\key	<u>192</u>		823, 825, 826,
		T	827, 828, 829, 830
	M	\tikzbattery 86, <u>89</u> , 753	
\menu	<u>72</u>	\tikzcalculator 694, <u>698</u>	
		\tikzfunction .. 98, <u>101</u>	\tikzmenu
	S	\tikzkey 195, <u>198</u> , 781,	74, <u>77</u> , 755, 756,
\setgraphcolor		783, 784, 785,	757, 758, 760,
.. <u>69</u> , 230, 266,		786, 787, 788,	761, 762, 763,
		789, 790, 791,	765, 766, 767, 768