

Graph35*

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

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

22 mars 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

1	Introduction	2
1.1	License	2
1.2	Sommaire	2
2	Téléchargement et installation	3
2.1	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	8
4	Binares	9
A	Calculatrices	9

*Ce document décrit le paquet `graph35` v0.1.0, publié le 2018/03/21. 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	11
B.4 Ancres du boîtier	11
C Pixel art	13
C.1 Menu	13
C.2 Fonctions	13
C.3 Batterie	25
D Touches	26
E Implementation	26
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 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^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}{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 temps¹.


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 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.
- 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`

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.
`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 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

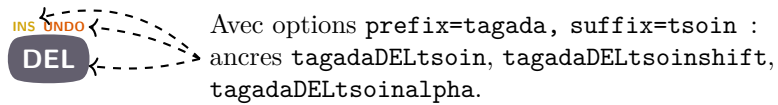
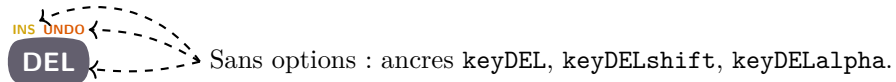
`\key` Pour inclure une touche de la calculatrice, utiliser :

`\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 ancrés 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 :

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


Ses arguments sont :

- $[\langle options \rangle]$ 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[\langle options \rangle]{\langle icône \rangle}{\langle raccourci \rangle}{\langle coordonnées \rangle}` dessine une icône du menu dans un environnement TikZ. Ses arguments sont :

- $\{\langle icône \rangle\}$ et $\{\langle raccourci \rangle\}$: mêmes arguments que pour `\menu` ;
- $\{\langle coordonnées \rangle\}$: coordonnées du coin supérieur gauche du menu ;
- $[\langle options \rangle]$: 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  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 ancrs 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.

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.



FIGURE 1 – Calculatrice graph35+E.

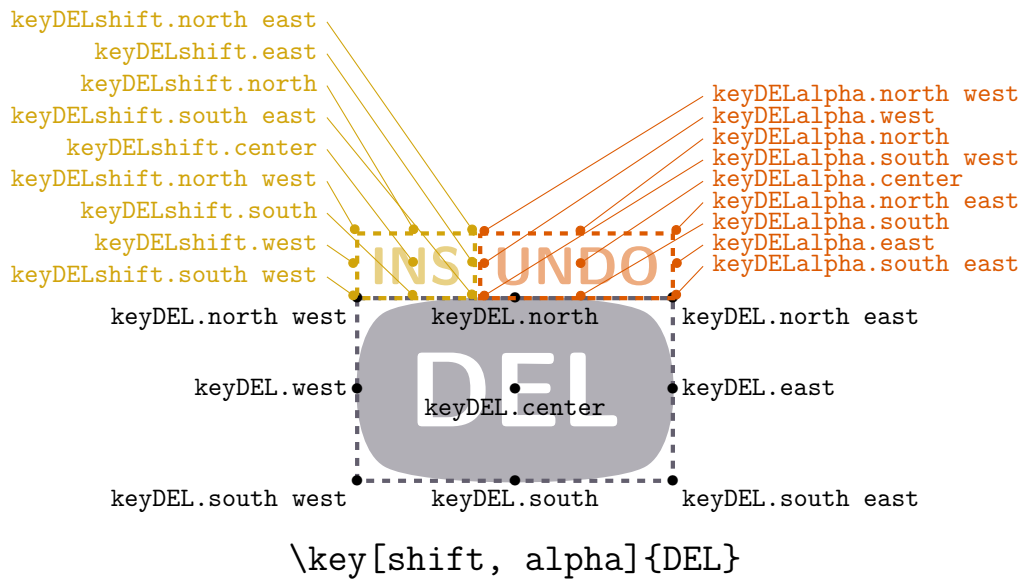


FIGURE 2 – Ancres des touches

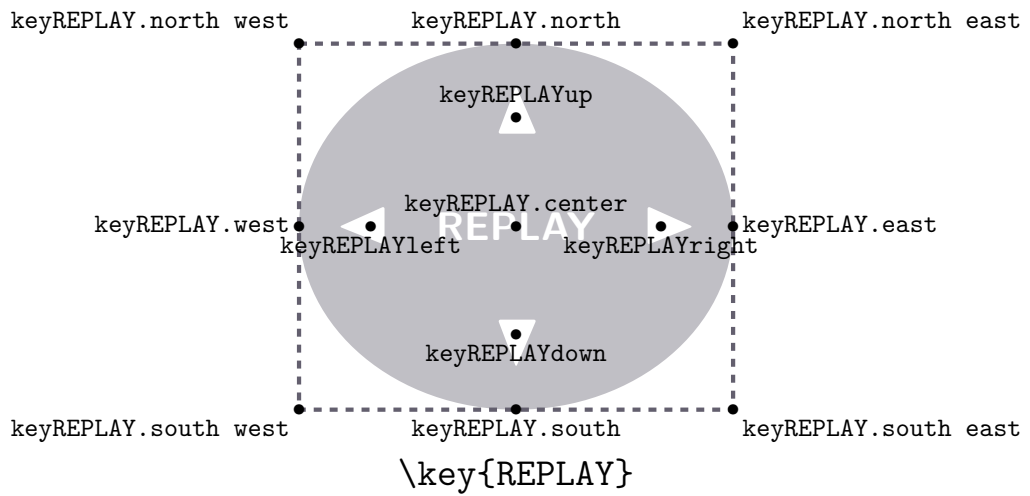


FIGURE 3 – Ancres de la touche REPLAY

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.

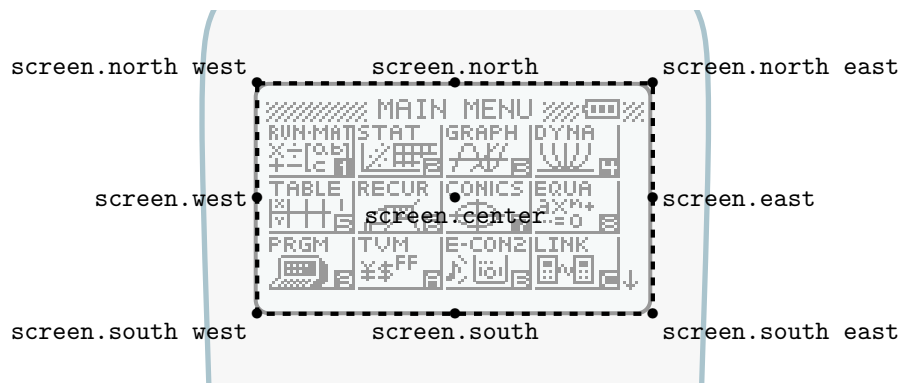


FIGURE 4 – Ancres de l'écran

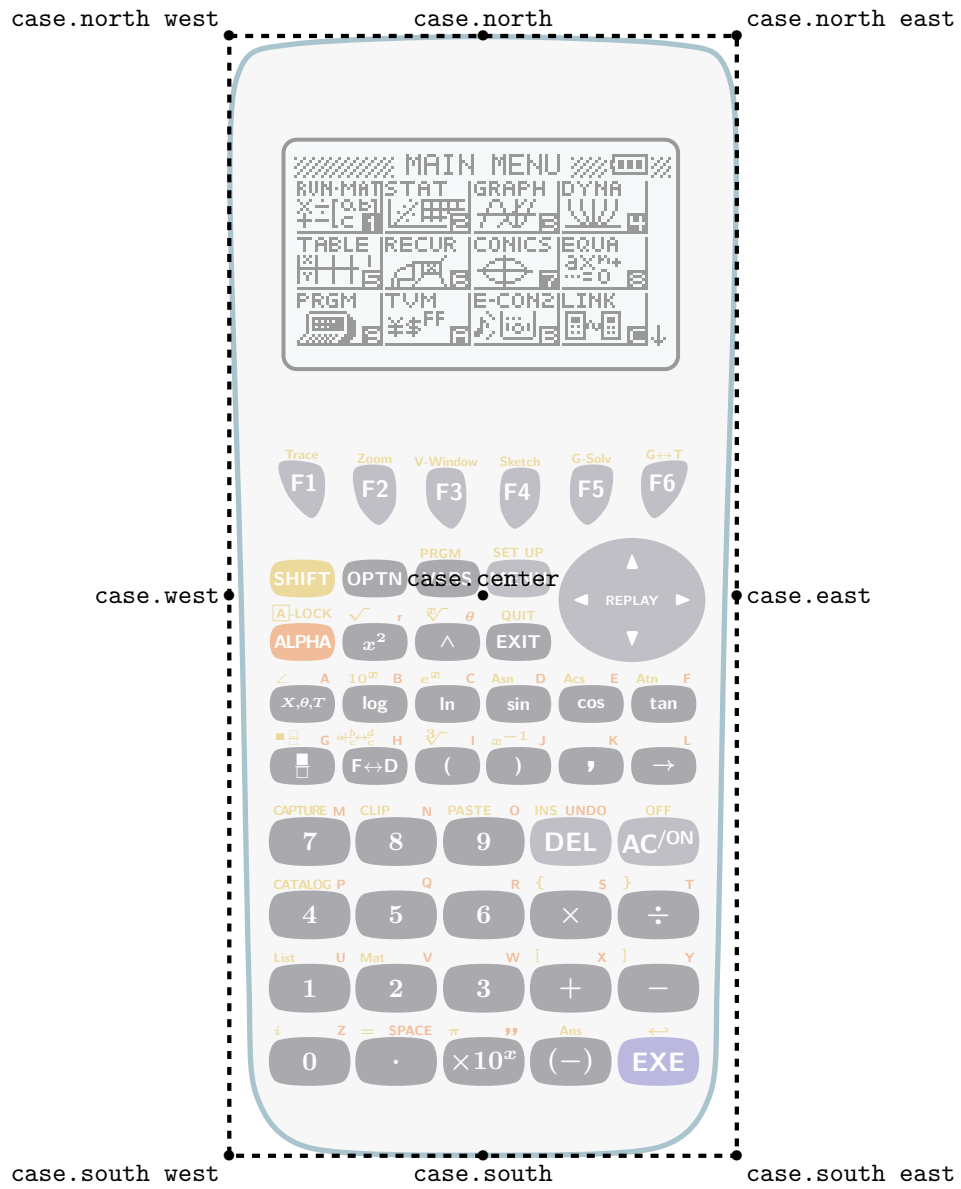























FIGURE 5 – Ancres du boîtier

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}{B}</code>
— 	<code>\menu{black}{2}</code>	— 	<code>\menu{black}{black}</code>
— 	<code>\menu{black}{3}</code>	— 	<code>\menu{black}{blank}</code>
— 	<code>\menu{black}{4}</code>	— 	<code>\menu{black}{C}</code>
— 	<code>\menu{black}{5}</code>	— 	<code>\menu{black}{D}</code>
— 	<code>\menu{black}{6}</code>	— 	<code>\menu{black}{E}</code>
— 	<code>\menu{black}{7}</code>	— 	<code>\menu{black}{F}</code>
— 	<code>\menu{black}{8}</code>	— 	<code>\menu{black}{G}</code>
— 	<code>\menu{black}{9}</code>	— 	<code>\menu{black}{H}</code>
— 	<code>\menu{black}{A}</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.

Vide

	battery		style5		2WAY
	blank		style6		3
	colon-b		style7		3-b
	contrast-b		tilde-b		31
	degree-b		1		3x1
	Delta-b		10		33
	different		10		3x3
	different-b		100		38k
	dms		100		to38k
	dms-b		1p		3pin
	dollar-b		1P		3PIN
	doublequote-b		1S		4
	doublerightarrow		1var		4-b
	equal-b		1VAR		5
	geq-b		1VAR-b		5-b
	GREEK		2		6
	greek		2-b		6-b
	gt		200		60
	gt-b		200		7400
	key		21		7400
	leq-b		2x1		9850
	lt		22		9860
	lt-b		2x2		9860
	micro-b		2p		a
	next		2P		a-b
	nextb		2s		a0
	output-b		2S		a0
	percent-b		2var		a0-b
	period-b		2VAR		a1
	question-b		2VAR-b		a1
	quote-b		2way		a1-b
	rightarrow				
	Sigma-b				
	square-b				
	style1				
	style2				
	style3				
	style4				

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

BINM BINM-b	CABL CABL-b	CLR CLR-b
bkup	calb	cls
BKVP BKVP-b	CALB CALB-b	CLS cls
bn	calc	CLS Cls-b
Bn bn-b	CALC CALC	cma
Sbn Sbn-b	CALC CALC-b	CMA CMA-b
bn1	calib	cmp
Bn1 bn1-b	CALIB CALIB	Cmp Cmp-b
Sbn1 Sbn1-b	capa	cmpd
bn2	CAPA CAPA-b	CMPD CMPD-b
Bn2 bn2-b	capt	cmpr
Sbn2 Sbn2-b	Capt capt	CMPR CMPR-b
bnst	CAPT CAPT-b	cn
BnSt bnSt-b	cash	Cn cn-b
bond	CASH CASH-b	Scn Scn-b
BOND BOND-b	casio	cn1
bot	CASIO CASIO-b	Cn1 cn1-b
BOTb BOTbottom	ccd	Scn1 Scn1-b
BOTr BOTright	Ccd Ccd	cn2
box	cel	Cn2 cn2-b
BOX BOX	CEL CEL-b	Scn2 Scn2-b
Box Box-b	cell	cnst
bpd	CELL CELL	CnSt CnSt-b
Bpd Bpd	ch1	cnt
brk	CH1 CH1	Cnt cnt
Brk Brk-b	char	cnvt
brkn	CHAR CHAR-b	CNVT CNVT-b
Brkn Brkn-b	chg	col
btm	CHG Chg-b	COL COL
BTM BTM	chi	COL COL-b
c	CHI CHI	com
C c-b	CHI CHI-b	COM COM-b
c0	Chi Chi-b	conj
C0 C0-b	chng	Conj Conj-b
c1	CHNG CHNG	conv
C1 C1-b	close	CONV CONV-b
c2	CLOSE Close-b	copy
C2 C2-b	clr	COPY COPY
cabl	CLR CLR	COPY COPY-b
		cosh
		COSH cosh-b
		cosh1

~~COSH~~ cosh1-b
cost
~~COST~~ COST
~~COST~~ COST-b
~~Cost~~ Cost-b
cpd
~~Cpd~~ Cpd
cplx
~~CPLX~~ CPLX-b
crcl
~~Crcl~~ Crcl
~~Crcl~~ Crcl-b
crnt
~~CRNT~~ CRNT-b
cstm
~~CSTM~~ CSTM-b
ctgy
~~CTGY~~ CTGY-b
ctl
~~CTL~~ CTL-b
cuml
~~Cuml~~ Cuml-b
cut
~~CUT~~ CUT
cy
~~CY~~ CY-b
d
~~d~~ d-b
d2dt2
~~d2dt2~~ d2dt2
d2dx2
~~d2dx2~~ d2dx2-b
data
~~DATA~~ DATA-b
~~Data~~ Data-b
~~percentDATA~~ percentDATA-b
days
~~DAYS~~ DAYS-b
db

~~DB~~ DB
ddt
~~ddt~~ ddt
ddx
~~ddx~~ ddx-b
defg
~~DefG~~ DefG-b
del
~~DEL~~ DEL
~~DEL~~ DEL-b
dela
~~DELA~~ DELA-b
dell
~~DELL~~ DELL-b
depr
~~DEPR~~ DEPR-b
det
~~DET~~ Det-b
df
~~df~~ df-b
diff
~~diff~~ diff
dim
~~DIM~~ DIM-b
~~Dim~~ Dim-b
disp
~~DISP~~ DISP-b
dist
~~DIST~~ DIST-b
dld
~~dlminusD~~ dlminusD
~~dlplusD~~ dlplusD
dms
~~tDMS~~ tDMS-b
do
~~DO~~ Do-b
dot
~~DOT~~ dot-b
draw

~~DRAW~~ DRAW
~~DRAW~~ DRAW-b
drwc
~~DrwC~~ DrwC-b
drwf
~~DrwF~~ DrwF-b
drwn
~~DrwN~~ DrwN-b
drwt
~~Drwt~~ Drwt-b
dsz
~~Dsz~~ Dsz-b
dx
~~Idx~~ Idx
~~Idx~~ Idx-b
dyna
~~DYNA~~ DYNA-b
~~Dyna~~ Dyna-b
e
~~e~~ e-b
~~E~~ Exa-b
edf
~~Edf~~ Edf-b
edit
~~EDIT~~ EDIT
~~EDIT~~ EDIT-b
eff
~~EFF~~ EFF-b
~~tEFF~~ tEFF
else
~~Else~~ Else-b
end
~~End~~ End-b
eng
~~ENG~~ ENGshiftleft
~~ENG~~ ENGshiftright
engy
~~ENGY~~ ENGY-b
entr
~~ENTR~~ ENTR-b
equa
~~EQUA~~ EQUA-b

es	ELL FILL-b	ECN GCON
ES EtS-b	ELL Fill-b	ECN Gcon-b
esym	eline	gdx
ESYM ESYM-b	ELINE FLine	GDx GIdx-b
exam	ELINE FLine-b	geo
EXAM EXAM-b	fmax	GE0 GEO-b
exe	FMX FMax-b	gmem
EXE EXE	fmin	GMEM GMEM-b
exit	FMN FMin-b	go
EXIT EXIT	for	GO GO
EXIT EXIT-b	FOR For-b	gof
exp	forc	GOF GOF
Exp Exp	FORC FORC-b	goto
EXP EXP-b	form	GOTO Goto-b
Exp Exp-b	FORM FORM	gpd
Exp Exp-b2	FORM FORM-b	GPD Gpd
extd	fp	gph1
EXTd Extd	FP FP	GPH1 GPH1
f	FP FP-b	GPH1 GPH1-b
F F	fpd	gph2
F F-b	Fpd Fpd	GPH2 GPH2
F F-b2	frac	GPH2 GPH2-b
f femto-b	FRAC Frac-b	gph3
fa	ftbl	GPH3 GPH3
FA Fa-b	FTbl FTbl-b	GPH3 GPH3-b
fab	full	gplt
FAB Fab-b	FULL FULL	GPLT GPLT
fact	furie	GPlt GPlt-b
FACT FACT-b	FURIE Furie	grab
Fact Fact-b	fv	GRAB GRAB
fast	FV FV	grph
FAST Fast	FV FV-b	GRPH GRPH
fb	g	GRPH GRPH-b
Fb Fb-b	G g-b	GRPH Grph-b
fcd	G Giga-b	gslv
Fcd Fcd	gcd	GSLV GSLV-b
file	Gcd Gcd	gtk
FILE FILE-b	GCD GCD-b	Gtk Gtk-b
fill	gcon	hcd

Hcd Hcd	INT INT-b	Join Join-b
help	Int Int-b	jump
HELP HELP-b	Intdiv Intdiv-b	JUMP JUMP-b
hgeo	SINT SINT	k
HGEO HGEO-b	SINT-b SINT-b	kilo kilo-b
hist	intg	lang
HIST Hist-b	INTG INTG	LANG LANG-b
hpd	Intg-b Intg-b	lbl
Hpd Hpd	intr	Lbl Lbl-b
hyp	INTR INTR-b	lcm
HYP HYP-b	inv	LCM LCM-b
hztl	Inv Inv	lcte
HZTL Hzt1	Inv-b Inv-b	Lcte Lcte-b
HZTL Hzt1-b	invb	left
i	InvB InvB	Left Left-b
i i-b	invc	len
I% Ipercent	InvC InvC	Len Len-b
I% Ipercent-b	invf	leng
iden	InvF InvF	LENG LENG-b
IDEN Iden-b	invg	LENG Leng-b
iend	InvG InvG	lgst
IEND IEnd-b	invh	Lgst Lgst
if	InvH InvH	Lgst Lgst-b
If If-b	invn	line
imp	InvN InvN	Line Line
IMP Imp-b	invp	LINE LINE-b
in	InvP InvP	Line Line-b
IN IN	invt	list
init	InvT InvT	List List
INIT INIT	io	LIST LIST-b
inpt	IO IO-b	List List-b
INPT INPT-b	irr	tLIST tLIST-b
input	IRR IRR	lm
INPUT INPUT	IRR IRR-b	LtoM LtoM-b
ins	isct	lmem
INS INS	ISCT ISCT	LMEM LMEM-b
INS INS-b	isz	load
int	Isz Isz-b	LOAD LOAD-b
INT INT	join	log

~~LOG~~ Log
~~LOG~~ Log-b
 logab
~~LOGAB~~ logab-b
 logic
~~LOGIC~~ LOGIC-b
 lpw
~~LPLW~~ 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
~~MtL~~ 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

~~MRS~~ Mrg-b
 ms
~~MS~~ 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
~~Psnd~~ 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
~~PlChg~~ PlChg
~~PlChg~~ PlChg-b
 ploff
~~PLOff~~ PLOff
~~PLOff~~ PLOff-b
 plon
~~PLOn~~ PLOn
~~PLOn~~ PLOn-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	r2 r2-b	REPT REPT
SPRN SPRN	r38k	reslt
SPRN SPRN-b	R38k R38k-b	RESLT RESULT-b
prob	ran	Reslt Reslt-b
PROB PROB-b	Ran Ran-b	right
prod	rand	Right Right-b
Prod Prod-b	RAND RAND-b	rmdr
prog	RANG RANG-b	Rmdr Rmdr-b
PROG PROG-b	rcl	rnd
Prog Prog-b	RCL RCL	RND RND
proj	RCL RCL-b	Rnd Rnd-b
Proj Proj	Rcl Rcl-b	rndfi
ptch	rdel	RndFi RndFi-b
Ptch Ptch-b	RDEL RDEL	rnf
pts	rec	RNF RNF-b
PTS PTS-b	Rec Rec-b	root
pv	recal	ROOT ROOT
PV PV	RECAL RECAL	rop
PV PV-b	recr	ROP ROP-b
pwr	RECT RECT	rot
Pwr Pwr	recv	Rot Rot-b
PWR PWR-b	RECV RECV	row
Pwr Pwr-b	Recv Recv	ROW ROW
py	Recv Recv-b	ROW ROW-b
PY PY-b	ref	rref
q	REF Ref-b	Rref Rref-b
Q Qsnd-b	reg	rset
q1	REG REG	RSET RSET-b
Q1 Q1-b	REG REG-b	rt
q3	rel	RT RT
Q3 Q3-b	REL REL-b	RTtheta RTtheta-b
r	ren	rtbl
r r-b	REN REN-b	RTbl RTbl-b
r r-b2	rep	rtrn
r r-b3	REP Rep-b	Rtrn Rtrn-b
r= requal	rept	run
r= requal-b		RUN RUN
R Rsnd-b		rw
Rz* tcomplexpolar-b		Rw+ Rwplus

rx
 ~~R-X~~ RX-b
 ry
 ~~R-Y~~ 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
 simpl
 ~~SMPL~~ SMPL-b
 snd
 ~~SND~~ Snd
 solv
 ~~SOLV~~ SOLV
 ~~SOLV~~ SOLV-b
 solve
 ~~SOLVE~~ Solve
 solvn
 ~~SOLVN~~ SolvN-b
 sonic
 ~~SONIC~~ sonic
 sp
 ~~SP~~ sp-b
 sqr
 ~~SQR~~ SQR





src
 ~~SRC~~ SRC
 ~~SRC~~ SRC-b
 ~~SRC~~ Src-b
 srta
 ~~SRTA~~ SRTA
 ~~SRTA~~ SrtA-b
 srt d
 ~~SRTD~~ SRTD
 ~~SRTD~~ SrtD-b
 ssa
 ~~SSA~~ SSa-b
 ssab
 ~~SSAB~~ SSab-b
 ssb
 ~~SSB~~ SSb-b
 sse
 ~~SSE~~ SSE-b
 stat
 ~~STAT~~ STAT-b
 ~~STAT~~ Stat-b
 std
 ~~STD~~ STD
 step
 ~~STEP~~ Step-b
 stick
 ~~STICK~~ STICK-b
 sto
 ~~STO~~ STO-b
 ~~STO~~ Sto-b
 stop
 ~~STOP~~ STOP
 ~~STOP~~ Stop-b
 str
 ~~STR~~ STR
 ~~STR~~ STR-b
 ~~STR~~ Str-b
 strp
 ~~STRP~~ STRP-b
 strt
 ~~STRT~~ STRT
 ~~STRT~~ Strt-b

stup	STUP STUP-b	TANH tanh1-b	type	TYPE TYPE-b
styl	STYL STYL-b	tcd	unit	UNIT UNIT-b
sum	SUM Sum-b	test	upr	UPR Upr-b
svas	SVAS SVAS-b	text	usb	USB USB
swap	SWAP SWAP	TEXT TEXT	var	VAR var
sx	SX sx-b	TEXT Text	VAR VAR-b	Var Var-b
sx1	SX1 sx1-b	then	vct	VCT VCT-b
sx2	SX2 sx2-b	THEN Then-b	velo	VELO VELO-b
sy	SY sy-b	time	ver	VER VER-b
sybl	SYBL SYBL	TIME TIME-b	vert	VERT Vert
syd	SYD SYD	tlow	VERT Vert-b	
t	T T	tlow	vlum	VLUM VLUM-b
	T t-b	tmpr	vnlk	VNLK VNLK-b
	T t-b2	TMPR TMPR-b	vrnr	VRNR VRNR-b
	T Tera-b	to	vwin	VWIN VWIN-b
	T tsnd-b	TO To-b	VWIN VWin-b	
	T.B Ttheta-b	tool	wake	WAKE WAKE-b
tabl	TABL TABL	TOOL TOOL-b	web	WEB WEB
	TABL TABL-b	top	WEB Web-b	
	Tabl Tabl-b	TOP TOP	wend	WEND WEnd-b
tang	TANG Tang	TOP TOPleft	whle	WHLE Whle-b
	TANG Tang-b	TOP TOPtop	wiz	WIZ WIZ-b
tanh	TANH tanh-b	tpd	x	
tanh1	TANH tanh1-b	TPD tpd		
		tran		
		TRAN TRAN		
		TRAN TRAN-b		
		trig		
		TRIG TRIG		
		trn		
		TRN Trn-b		
		tup		
		TUP tUp-b		
		tvm		
		TVM TVM-b		

fact! factorialx-b	xpower3-b	Yleq-b
sigmax-b	x4	Ylt-b
Sx-b	X4	y1
txequal	x4	y1-b
txgeq	xpowers4-b	y2
txgt	xcal	Sy2-b
txleq	XCAL	y2-b
txlt	xfct	y3
x	Xfct-b	y3-b
X-b	xinv	ycal
x-b	xInv-b	YCAL
X-b2	xor	yes
X-b3	Xor-b	YES
xbar-b	xrw	yfct
xequal	XRw	Yfct-b
xequal-b	XRwplus	yicpt
xgeq-b	xt	YICPT
xgt-b	Xt-b	yld
xhat-b	xy	YLD
xleq-b	Sxy-b	YLD-b
xlt-b	xy-b	yt
x1	y	Yt-b
x1-b	sigmay-b	z
xbar1-b	Sy-b	Z
x1inv	tYequal	Z-b
x1Inv-b	tYgeq	z-b
x2	tYgt	zero
Sx2-b	tYleq	ZERO
X2	tYlt	zlow
x2	Y	zLow-b
x2-b	Y-b	zoom
xbar2-b	Y-b2	ZOOM
xpowers2-b	ybar-b	ZOOM-b
x2inv	Yequal	zup
x2Inv-b	Yequal-b	zUp-b
x3	Ygeq-b	
X3	Ygt-b	
x3	yhat-b	
x3-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
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}
13
14 \RequirePackage{graph35-pixelart}
15 \RequirePackage{graph35-keys}

```

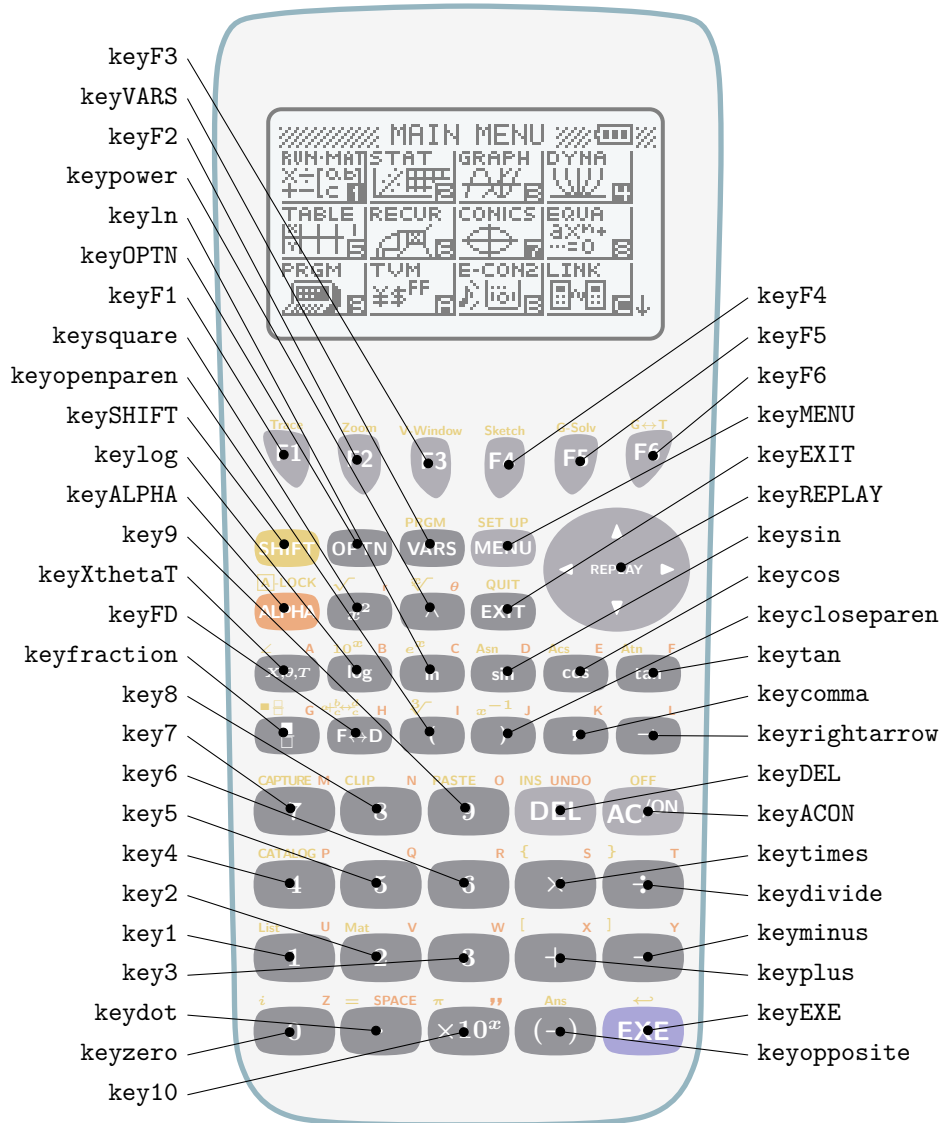


FIGURE 6 – Mots-clefs des touches

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 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 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. 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 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@tikzshiftalpha{,
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@tikzshiftalpha{,
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=0pt] (\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=graph35SHIFTEXT,
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=0pt] (\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
368     \ifdefstring{\graph@type}{text}{
369       \node[color=graph35KEYTEXT] {\tiny\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/conten
370     }}

```

```

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 ALPHA.

```

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 n
385         \draw[very thick, color=graph35ALPHABORDER, fill=graph35ALPHA] plot [smooth cycle] coordina
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/K
392     \ifdefstring{\graph@type}{text}{
393         \node[color=graph35KEYTEXT] {\scriptsize\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/
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}\pgfkeysvalue
401     position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
402     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalue
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 MENU.

```


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}\pgfkeysvalueof{/GRAPH35/KEY/shift}},
471     position=\pgfkeysvalueof{/GRAPH35/KEY/shift position},
472     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalueof{/GRAPH35/KEY/shift}},
473     color=graph35SHIFTEXT,
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}\pgfkeysvalueof{/GRAPH35/KEY/alpha}},
481     position=\pgfkeysvalueof{/GRAPH35/KEY/alpha position},
482     node={\pgfkeysvalueof{/GRAPH35/KEY/prefix}\pgfkeysvalueof{/GRAPH35/KEY/name}\pgfkeysvalueof{/GRAPH35/KEY/alpha}},
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 node/.style={draw=graph35ACONBORDER, fill=graph35ACON}]
495     \draw[very thick, color=graph35ACONBORDER, fill=graph35ACON] plot [smooth cycle] coordinates {
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/KEY/content}) at (0,0) {\pgfkeysvalueof{/GRAPH35/KEY/content}};
503   \end{scope}
504   \ifdefstring{\graph@type}{text}{
505     \node[color=graph35KEYTEXT] {\small\bfseries\sffamily{\pgfkeysvalueof{/GRAPH35/KEY/content}}}%
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}\pgfkeysvalueof{/GRAPH35/KEY/shift}},

```

```

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   }}{
538     \node[color=graph35KEYTEXT] {\small$\boldsymbol{\pgfkeysvalueof{/GRAPH35/KEY/content}}$};
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 graph35+E .	10
2	Ancres des touches	10
3	Ancres de la touche REPLAY	11
4	Ancres de l'écran	11
5	Ancres du boîtier	12
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	S	T
<code>\battery</code> <u>84</u>	<code>\setgraphcolor</code>	<code>\tikzbattery</code> 86, <u>89</u> , 753
C	.. <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>
F	336, 340, 355,	<code>\tikzkey</code> .. 195, <u>198</u> ,
<code>\function</code> <u>96</u>	359, 379, 383,	781, 783–802,
K	409, 413, 449,	804–823, 825–830
<code>\key</code> <u>192</u>	453, 489, 493,	<code>\tikzmenu</code> 74,
M	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	