

Création d'empreintes pour le programme de placement open-source « PCB »

<i>Author</i>	<i>e-mail</i>	<i>Date</i>	<i>Changes made</i>
Stephen Meier		2003, 2004	Wrote initial document.
Stuart D. Brorson	sdb@cloud9.net	27/12/04	Formatting changes, added more tables & info.
Stuart D. Brorson	sdb@cloud9.net	29/01/05	Included dimensioned pad drawing, incorporated changes suggested by Dan McMahill, updated ElementArc, other improvements.

Copyright © 2003 Meier Rippin L.L.C.
Copyright © 2004 MRA Tek LLC
Copyright © 2004, 2005 Stuart D. Brorson
Traduit par Iznogood de la Iznogood-factory.org

Introduction

PCB est un programme open-source utilisé pour la conception physique de circuits imprimés (i.e. placement de composants sur les cartes). PCB est une application GPL librement téléchargeable qui fonctionne sous Linux, BSD et autres Unices supportant X11. PCB fait partie de la suite gEDA (<http://www.geda.seul.org/>). La page d'accueil de PCB est <http://pcb.sf.net/>. Si vous n'êtes pas familiers de la conception avec PCB ou de la suite gEDA, vous devriez prendre le temps de vous familiariser avec ces sujets avant de continuer à lire ce document.

Lors de la conception d'un circuit imprimé, une des choses les plus importantes à définir sont les motifs d'emplacements – ou empreintes (footprints) – pour chaque composant. Comme tout programme de placement, PCB nécessite une empreinte pour chaque composant. L'empreinte indique à PCB comment dessiner les pastilles (pads) du composant ou les trous pour les broches, les contours de l'emplacement du composant, son nom et autres propriétés associées avec un composant individuel. Les empreintes utilisées dans PCB sont habituellement stockées dans un fichier externe et sont lues par PCB lorsque le circuit est créé ou mis à jour. Un des avantages majeurs de l'utilisation de PCB est que les fichiers d'empreintes sont en ASCII brut et sont bien structurés. L'objectif de ce document est de détailler la manière dont les empreintes PCB (motifs d'emplacements) sont définies dans les fichiers d'empreintes et expliquer comment les utiliser efficacement pendant le placement en utilisant PCB.

Une fonctionnalité inhabituelle de PCB est qu'il supporte deux bibliothèques d'empreintes complètement séparées avec deux mécanismes d'empreintes différents. C'est parce que PCB est un vieux programme qui a été développé par différentes personnes pour plusieurs plates-formes différentes pendant plus de deux décades. Avec son historique, PCB a dû traiter diverses limitations imposées par la vitesse, la taille de la mémoire, etc. de la plate-forme cible.

Le premier système d'empreinte est appelé « oldlib » ou la « M4 library » (NdT: bibliothèque M4). Ce système est historique (quelqu'un dit archaïque) ; elle dépend du langage de macro GNU M4 pour générer des empreintes à la volée. Les empreintes de la bibliothèque M4 sont préfixées par un tilde (« ~ »), par exemple, « ~geda ». **Ce document ne couvre pas l'utilisation de la bibliothèque M4 !** De toute manière, nous vous recommandons d'éviter l'utilisation de la bibliothèque M4 – son fonctionnement interne est obscur et plusieurs des empreintes ne sont pas contrôlées et peuvent être défectueuses (i.e. erreur dans la taille des trous de broche). Néanmoins, la bibliothèque M4 est grande, elle a ses supporters et forme une partie essentielle de PCB. Elle ne disparaîtra donc pas de sitôt.

Le second système d'empreintes pour PCB est appelé « newlib ». Les empreintes de la Newlib sont définies en utilisant des fichiers textes ASCII qui définissent chaque primitive graphique formant une empreinte entière. **Ce document se focalise uniquement sur la définition et l'utilisation des empreintes newlib.** Vous pouvez utiliser les empreintes newlib qui sont distribuées avec PCB ou vous pouvez créer les vôtres et les placer dans un répertoire dédié.

Il existe plusieurs manières de créer une empreinte newlib en utilisant PCB. Par exemple, vous pouvez créer une empreinte graphiquement avec PCB en la dessinant puis en la sauvegardant. Cette procédure est documentée dans la documentation PCB principale ; nous ne couvrons pas cette procédure ici. Ce document se concentrera sur la manière dont sont définies les empreintes dans le fichier ASCII. En utilisant cette information, vous pouvez soit créer une empreinte de zéro en utilisant un éditeur de texte

ou copier une empreinte existante et éditer ses paramètres pour qu'ils correspondent exactement à l'empreinte que vous souhaitez créer. Les utilisateurs chevronés de PCB préfèrent habituellement la dernière méthode pour la création d'empreintes car il est facile de débiter avec un symbole connu et éditer les fichiers d'empreintes vous donnent plus de contrôle sur l'empreinte.

Comprendre comment initialiser correctement des empreintes dans un fichier est important. Une empreinte peut affecter d'une manière critique la fabricabilité d'une carte sur laquelle elle est implantée. Si les pastilles d'empreinte sont au mauvais endroit ou si le masque de soudure n'est pas correctement défini, il peut être impossible d'attacher le composant à ses pastilles (pads). Si le masque de soudure ne couvre pas les pistes à côté des pastilles, les pistes peuvent être soudées aux pastilles. Les circuits utilisant des empreintes qui ont les pastilles au bon emplacement mais de la mauvaise taille peuvent avoir une fabricabilité réduite ou diminuer leur durée de vie. C'est pourquoi la définition correcte des empreintes est une partie critique de la création d'une carte fonctionnelle. Veuillez lire le document standard IPC-SM-782A « Surface Mount Design and Land Pattern Standard » pour une information plus complète des impératifs et des impacts des motifs des composants de montage en surface.

Comment sont utilisées les empreintes dans votre fichier .pcb

Dans le fichier PCB (habituellement appelé quelque chose comme foo.pcb), une empreinte individuelle est appelée un « Element ». Si vous examinez un fichier .pcb, vous verrez plusieurs déclarations d'Element dans le fichier – vous devriez avoir une déclaration pour chaque composant que vous avez placé dans votre circuit.

La première ligne de l'entrée Element contient des informations sur l'empreinte elle-même. Dans les éléments sont placés les composants graphiques atomiques composant les empreintes. Ces derniers incluent les pastilles à souder, les trous pour les broches, les lignes dessinées sur la couche imprimée (silkscreen) et les autres éléments qui composent une empreinte. Par exemple, ce qui suit est un motif PCB simple pour une résistance 0603.

```
Element(0x00 "Surface Mount Chip Resistor 0603" "R0" "" 0 0
-31 -82 2 100 0x00)
(
    Pad(-2 0 2 0 39 30 50 "pad 1" "1" 0x00000100)
    Pad(65 0 69 0 39 30 50 "pad 2" "2" 0x00000100)
    ElementLine(-21 -35 87 -35 5)
    ElementLine( 87 -35 87 35 5)
    ElementLine( 87 35 -21 35 5)
    ElementLine(-21 35 -21 -35 5)
)
```

Entre les parenthèses, après « Element », nous avons l'information appartenant à l'empreinte complète, telle que sa position, son refdes et ainsi de suite. Après cette première information, nous avons deux éléments graphiques « Pad ». Cet exemple de résistance en 0603 nécessite deux pastilles ; d'autres composants peuvent en nécessiter des centaines, menant à des centaines de lignes de « Pad » pour un tel composant. L'information de la ligne « Pad » spécifie la taille de la pastille pour la soudure, tous les dégagements autour de la pastille, chaque nom et numéro de pastille et autres attributs. L'élément graphique « ElementLine » produit une ligne dessinée sur la couche graphique. Dans cet exemple, il existe quatre éléments « ElementLine » dessinés sur la couche imprimée, correspondant à une boîte autour de l'empreinte 0603. L'information dans la ligne « ElementLine » spécifie la largeur et la position des lignes qui sont dessinées sur la couche imprimée.

De plus amples détails sur la définition de ces éléments graphiques sont fournis dans les sections suivantes de ce document.

Particularités de PCB

À cause de son long historique de développement, PCB possède certaines particularités qu'il est intéressant de connaître. Cette section tente d'en lister certaines nécessaires pour créer ou éditer des fichiers d'empreintes.

- **Système de coordonnées.** Il est particulièrement important de rappeler que PCB utilise le système de coordonnées graphiques standard des ordinateurs et pas un système de coordonnées Cartésiennes. Cela signifie que X augmente vers la droite (comme d'habitude) mais Y augmente en descendant. Veuillez garder ceci en mémoire lorsque vous définissez les emplacements graphiques des éléments d'empreintes.
- **Unités.** À l'origine, PCB utilisait les mils (1/1000ième d'un pouce (inch)) comme unité de mesure de base. Néanmoins, les récentes mise à jour du programme ont amélioré sa résolution à 1/100 mils (1e-5 in). Les deux unités sont librement utilisables dans les fichiers d'empreintes. Par définition, les unités entre parenthèses « () » sont des **mils**. Les unités entre crochets « [] » sont en **1/100 mils**. **Faites attention à cette dichotomie et contrôlez toujours la manière dont sont spécifiées vos unités!**
- **Bibliothèques d'empreintes.** La bibliothèque d'empreintes originale de PCB a été écrite en utilisant le langage de macro M4. L'opinion habituelle à propos de la bibliothèque d'empreinte M4 tient dans le fait qu'elle fut une idée astucieuse avec l'Amiga considérée comme une machine à la pointe mais que c'est maintenant une épave difficile à maintenir. Heureusement, un système de bibliothèque d'empreintes graphique moderne – newlib – a été développé pour PCB. La majorité des nouvelles empreintes fournies au projet PCB utilisent la syntaxe d'empreinte newlib. **La bibliothèque newlib est la syntaxe de la bibliothèque d'empreinte décrite dans ce document.**
- **Implémentation du masque de soudure de PCB.** PCB permet seulement à PADS de déterminer la taille et la forme du masque de soudure. C'est pourquoi la création de la Gang shadow masks windows (voir Glossaire) peut seulement se produire en initialisant la taille de PAD et en plaçant correctement les composants individuels suffisamment proche les uns des autres de telle manière que la shadow mask windows se fusionne.
- **Keepouts (chevauchement).** Actuellement, PCB ne traite pas le concept de chevauchement. Vous devez donc chercher les chevauchement manuellement. Un anti-chevauchement de fortune peut être créé en entourant votre empreinte avec une boîte dessinée sur la couche graphique. Elle doit s'étendre un peu au-delà du contour du composant dans toutes les directions. Alors, pendant la vérification manuelle de la disposition des composants, il faut contrôler qu'aucune boîte ne touche à l'autre. Notez qu'aucun DRC n'apparaîtra en utilisant cette méthode, vous devez donc faire attention lors du placement et de l'inspection des composants.

Développer une nouvelle empreinte pour PCB – work flow

Lors de la création d'une nouvelle empreinte, vous suivrez typiquement le cheminement qui suit:

1. Déterminer la marque (centre) de l'empreinte. C'est souvent soit le centre du composant, soit sa broche 1.
2. Déterminer la rotation du composant autour de la marque. Dans PCB, cela peut être 0, 90, 180 ou 270 degrés.
3. Déterminez le placement sur la grille (Grid) et sa relation avec le centre. C'est particulièrement important si vous avez prévu d'assembler votre circuit en utilisant une machine à insertion automatique car cette dernière voudra placer les composants à une position particulière de la grille. Consultez votre sous-traitant pour obtenir plus d'informations sur leurs impératifs. Si vous faites l'assemblage de votre carte à la main, vous n'avez pas besoin de vous en inquiéter.
4. Déterminer la méthode de soudure devant être utilisée. Les processus de soudure à la vague ou en refusion ont des nécessités sur la dimension des pastilles ainsi que sur les jours du masque de soudure (Solder Mask). En particulier, assurez-vous de comprendre comment votre masque doit être appliqué. De plus amples informations à ce propos peuvent être obtenues avec les documents IPC ou avec votre sous-traitant.
5. Déterminer l'emplacement et la taille des pastilles. Ils sont habituellement fournis avec les éléments fournis par les fabricants de composants. Alternativement, vous pouvez consulter les documents IPC qui spécifient les empreintes recommandées pour plusieurs composants communs. Gardez en mémoire que l'emplacement et la taille de la pastille dépendent – à un certain point – des tolérances de votre sous-traitant. Encore une fois, utilisez des valeurs larges pour vos dimensions de pastilles ou discutez-en avec votre sous-traitant sur leurs recommandations quant à leurs règles de placement.
6. Déterminer la méthode d'application de masque de soudure et leurs tolérances.
7. Déterminer l'épaisseur du masque de soudure. Ceci dépend du type de processus de soudure que vous avez l'intention d'utiliser, des tolérances de votre sous-traitant et d'autres facteurs. Il est toujours mieux d'utiliser des valeurs larges ou de consulter votre sous-traitant et de faire l'assemblage à la maison en premier.
8. Ouvrir un fichier d'empreinte dans un répertoire du chemin de recherche de PCB. Vous voudrez habituellement nommer le fichier avec un nom suggérant l'empreinte(s) qu'il contient. Une ou plusieurs empreintes (Elements) peuvent vivre dans un fichier simple. Aucune convention de dénomination n'est obligatoire avec PCB. C'est pourquoi aucun suffixe de nom de fichier n'est nécessaire, bien que vous pourriez vouloir votre propre convention de dénomination. Les exemples peuvent être « Res_0805_large.fpt » ou « TQFP-44.pcbfootprint ».
9. Créer la macro Element dans le fichier empreinte.
10. Dans le corps d'Element, ajoutez une ligne Pad pour chaque pastille de composant.
11. Dans le corps d'Element, ajoutez une ligne Pin pour chaque broche traversante de composant. Vous pouvez aussi utiliser une ligne Pin afin de définir un trou pour le montage de composant.
12. Dans le corps d'Element, ajoutez des lignes ElementLine pour créer le motif du contour. ElementLine produira une ligne sur la couche imprimée. Le motif du contour n'a pas besoin d'englober le placement des broches mais cela peut être adapté pour un placement correct.

Anatomie d'une empreinte

Pour comprendre les paramètres utilisés dans la définition d'une empreinte, considérons l'empreinte utilisée par une résistance SMD (NdT: Surface Mount Device). Une empreinte est montrée ci-dessous avec l'illustration 1.

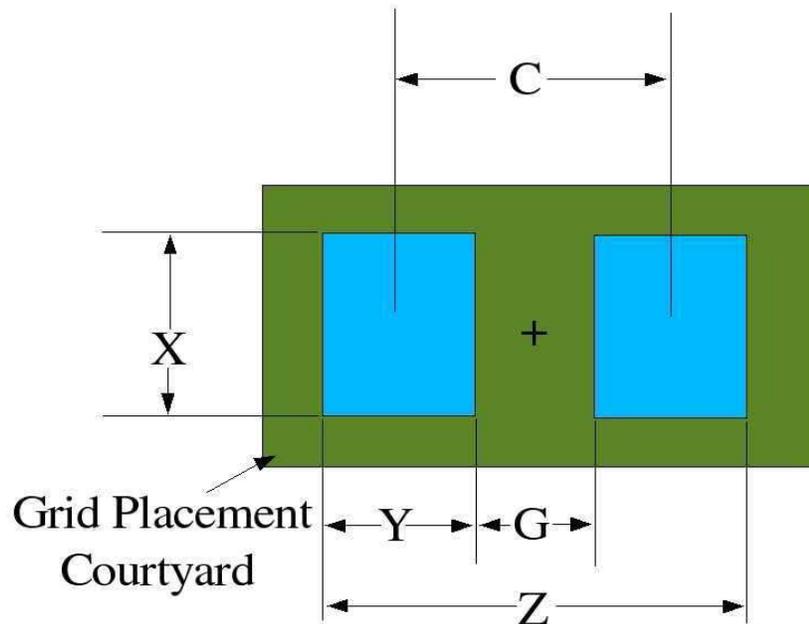


Illustration 1 Footprint (land pattern) of SMT resistor.

Les données pour différentes tailles de résistances sont présentées dans la table.

Type	C	X	Y	Z	G	Grid
'0402	51.2	27.5	35.4	86.6	15.7	39.4x118.1
'0603	66.9	39.4	43.3	110.2	23.6	157.5x118.1
'0805	74.8	59.1	51.2	126.0	23.6	157.5x315.0
1206	110.2	70.9	63.0	173.2	47.2	157.5x393.7
1210	110.2	106.3	63.0	173.2	47.2	118.1x393.7
2010	173.2	106.3	70.9	244.1	102.4	118.1x551.2
2512	220.5	126.0	70.9	291.3	149.6	315.0x629.9

Les dimensions C, X, Y, Z, G et Grid sont toutes en mils. Les données sont dérivées de la table dans la page 73 Of IPC-SM-782A « Surface Mount Design and Land Pattern Standard ».

Chaque Pad de PCB touche plusieurs couches. Si la Pad est sur la face composant, elle impacte la couche côté composants, le masque côté composant (l'épaisseur du masque de soudure côté composant) et le placement du composant. Si la couche composant possède un polygone alors ce dernier est effacé de la pastille de la valeur entrée dans la macro Pad.

Chaque instance de macro nécessite ces paramètres sélectionnés pour les techniques de fabrication utilisées pour placer et souder les composants sur la carte. Le document standard (IPC-SM-782A) les couvre en détail. Le champ de cet article couvrira l'utilisation de ces documents standard pour générer

les motifs adaptés à PCB.

Bonnes et mauvaises techniques de création d'empreintes

- Bonnes – Assurez-vous que vos pastilles de soudure sont suffisamment larges pour les composants SMT (SMD). Les pastilles devraient fournir suffisamment d'espace pour le développement d'un coude de soudure entre votre composant et la pastille elle-même. Les pastilles recommandées sont habituellement adaptées. Néanmoins, il peut quelques fois être utile d'augmenter la taille de la pastille de quelques mils dans chaque direction si vous avez de la place sur votre carte. Néanmoins, n'allez pas trop loin dans la finesse d'écartement ; si les pastilles sont trop rapprochées, vous courez le risque de créer des ponts de soudure entre des pastilles adjacentes !
- Bonnes – Contrôlez deux fois toutes vos empreintes (et aussi votre circuit). Un nombre important d'erreurs sur les circuits provient d'erreurs d'empreintes faute d'attention suffisante. Les éléments à contrôler :
 - Vos trous sont-ils suffisamment larges pour les broches ? Il n'y a pas de problèmes à agrandir légèrement les trous de quelques mils pour s'assurer que tout aille bien.
 - Vos pastilles sont-elles suffisamment larges ? L'espacement des pastilles est-il initialisé correctement ?
 - PCB utilise les mils comme unité de mesure pour les empreintes. Quelque fois, les fabricants utilisent des unités métriques en définissant les empreintes. C'est particulièrement vrai pour les connecteurs. Assurez-vous que vous avez converti les unités métriques en mils dans vos empreintes.
 - Êtes-vous sûr d'avoir les bonnes empreintes pour le boîtier que vous avez spécifié ? (J'ai personnellement eu plusieurs cas où une empreinte SO-16 avait été placée pour un composant MSOP-16. Un de ces cas était de ma faute ! Les erreurs comme ceci coûtent de l'argent.)
 - Mécaniques. Assurez-vous que les composants sont adaptés et que vous ne les avez pas placés trop près les uns des autres. De même si votre carte doit être dans un espace restreint ou qu'elle doit satisfaire des restrictions de hauteur, assurez-vous que vous avez correctement incorporé ces contraintes dans votre conception. Comme PCB ne gère pas les concepts de chevauchement ou les restrictions de hauteur, vous devez vérifier ces contraintes manuellement.
- Bonnes – Lorsque vous avez fini avec votre circuit, vérifiez les fichiers Gerber en utilisant un visualiseur Gerber. Cette étape importante vous aidera à trouver des erreurs qui ne serait pas montrées avec PCB. Plusieurs visualiseurs Gerber libres existent sur le net ; une recherche Google rapide en identifiera plusieurs pour vous. Sur Windows, j'utilise « GCPrevue ». Sur Linux, « gerbv » est un visualiseur Gerber décent.
- Bonnes – Effectuer un placement d'essai en utilisant vos composants. Une fois que vous avez créé votre PCB, transférez-le sur papier avec une imprimante PostScript en utilisant une échelle 1:1 et placez tous vos composants sur leur empreinte. C'est une bonne manière de trouver des erreurs d'empreintes (et d'autres sur le circuit).
- Bonnes – Utilisez le masque de soudure au-dessus de la zone de cuivre pour éviter la migration de

soudure. Les tolérances du masque de soudure : les impressions peuvent être utilisées pour produire des masques avec un espacement de 15 mils. Celles provenant de photos fournissent un espacement de 3 mils.

- Bonnes – Inspectez votre circuit et vérifiez que toutes les régions de plan de masse sont connectées à leurs réseaux respectifs. Les thermiques sont placés manuellement dans PCB, il est donc facile de les oublier. C'est particulièrement important si votre carte possède des couches intermédiaires, car il n'est pas facile de retravailler une couche interne. Vous pourriez aussi vouloir vérifier que les couches à plan de masse possèdent des vides (anti-pastilles) autour des vias ou des broches non-connectées.
- Bonnes – Inspectez votre circuit pour vérifier que toutes les annotations sont réalisées sur la couche imprimée. Le contrôleur DRC de PCB n'identifiera pas les erreurs dues aux textes sur les parties métal. Vérifiez aussi que votre texte n'est pas trop proche des pastilles métalliques – si votre fabricant a des problèmes de sérigraphie, l'encre peut aller sur vos broches et vous ne pourrez pas les souder.
- Mauvaises – Le masque de soudure ne doit pas couvrir les repères ou les zones de repères comme il peut y avoir une oxydation et interférer avec le positionnement automatique de ces repères.
- Mauvaises – La contamination par le masque de soudure des pastilles de composants peut créer des erreurs. Un masque de soudure insuffisant laissant des parties métal peut générer des connexions non désirées.

Element

La balise « Element » contient une empreinte complète pour un composant particulier. L'en-tête de l'Element contient des informations pertinentes pour la globalité de l'empreinte. Dans le corps de la macro Element sont placés les composantes graphiques individuelles de l'empreinte. Par exemple, chaque pastille ou broche du composant nécessite un trou. Généralement, le contour graphique est aussi fourni. Le corps est le code entre les parenthèses.

Format

```
Element (element_flags, description, pcb-name, value,
        mark_x, mark_y, text_x, text_y,
        text_direction, text_scale, text_flags)
(
    individual graphical components, such as Pad, Pin,
    or ElementLine.
)

Element [element_flags, description, pcb-name, value,
        mark_x, mark_y, text_x, text_y,
        text_direction, text_scale, text_flags]
(
    individual graphical components, such as Pad, Pin,
    or ElementLine.
)
```

Notez que soit les mils, soit les 1/100 de mil sont permis pour la balise Element. C'est signalé par l'utilisation des parenthèses « () » ou des crochets « [] ».

Description détaillée

Objet	Valeur permise	Explication	Commentaire
element_flags	unsigned hex value		
description	string	Texte de description de l'empreinte	Saisi par l'utilisateur.
pcb-name	string	Refdes utilisé sur ce PCB particulier	Ce champ est rempli par PCB lui-même. Laissez-le blanc lorsque vous définissez le fichier d'empreinte.
value	string	Valeur du composant sur ce PCB particulier	Ce champ est rempli par PCB lui-même. Laissez-le blanc lorsque vous définissez le fichier d'empreinte.

Objet	Valeur permise	Explication	Commentaire
mark_x	Decimal integer (mils or 1/100 mils)	C'est l'emplacement X de la marque de l'empreinte. Il indique à PCB où placer l'empreinte lors de la première lecture du circuit. Lorsque vous placez le composant, PCB initialisera cette valeur.	Habituellement initialisé à 10 mil donc la position initiale du composant est sur la zone de travail de la carte.
mark_y	Decimal integer (mils or 1/100 mils)	C'est l'emplacement Y de la marque de l'empreinte. Il indique à PCB où placer l'empreinte lors de la première lecture du circuit. Lorsque vous placez le composant, PCB initialisera cette valeur.	Habituellement initialisé à 10 mil donc la position initiale du composant est sur la zone de travail de la carte.
text_x	Decimal integer (mils or 1/100 mils)	Coordonnée X de la position initiale de Refdes w.r.t. mark location. Plus tard, PCB initialisera cette valeur lorsque vous déplacerez refdes.	On doit expérimenter de manière à trouver le positionnement optimal initial pour le texte.
text_y	Decimal integer (mils or 1/100 mils)	Coordonnée Y de la position initiale de Refdes w.r.t. mark location. Plus tard, PCB initialisera cette valeur lorsque vous déplacerez refdes.	On doit expérimenter de manière à trouver le positionnement optimal initial pour le texte.
text_direction	decimal integer	0 = horizontal 1 = CCW 90 deg 2 = 180 deg 3 = CW 90 deg	
text_scale	decimal integer		Habituellement initialisé à 100.
text_flags	unsigned hex value		

Exemple

```

Element(0x00 "Surface Mount Chip Resistor 0603" "" "" 0 0
-31 -82 0 100 0x00)
(
    Pad(-2 0 2 0 39 30 50 "pad 1" "1" 0x00000100)
    Pad(65 0 69 0 39 30 50 "pad 2" "2" 0x00000100)
    ElementLine(-21 -35 87 -35 5)
    ElementLine( 87 -35 87 35 5)
    ElementLine( 87 35 -21 35 5)
    ElementLine(-21 35 -21 -35 5)
)

```

Cet exemple définit une résistance 0603 SMT ayant deux pastilles de soudure et quatre ElementLines sur la couche graphique pour définir le motif. Notez que les dimensions dans cet exemple sont en **mils** car ils sont entre parenthèses.

Pad

L'élément Pad est inclus dans le corps de l'empreinte (Element). Il décrit une métallisation rectangulaire simple servant de motif pour le composant SMT.

Format

```
Pad (x1 y1 x2 y2 thickness clearance mask name pad_number flags)
```

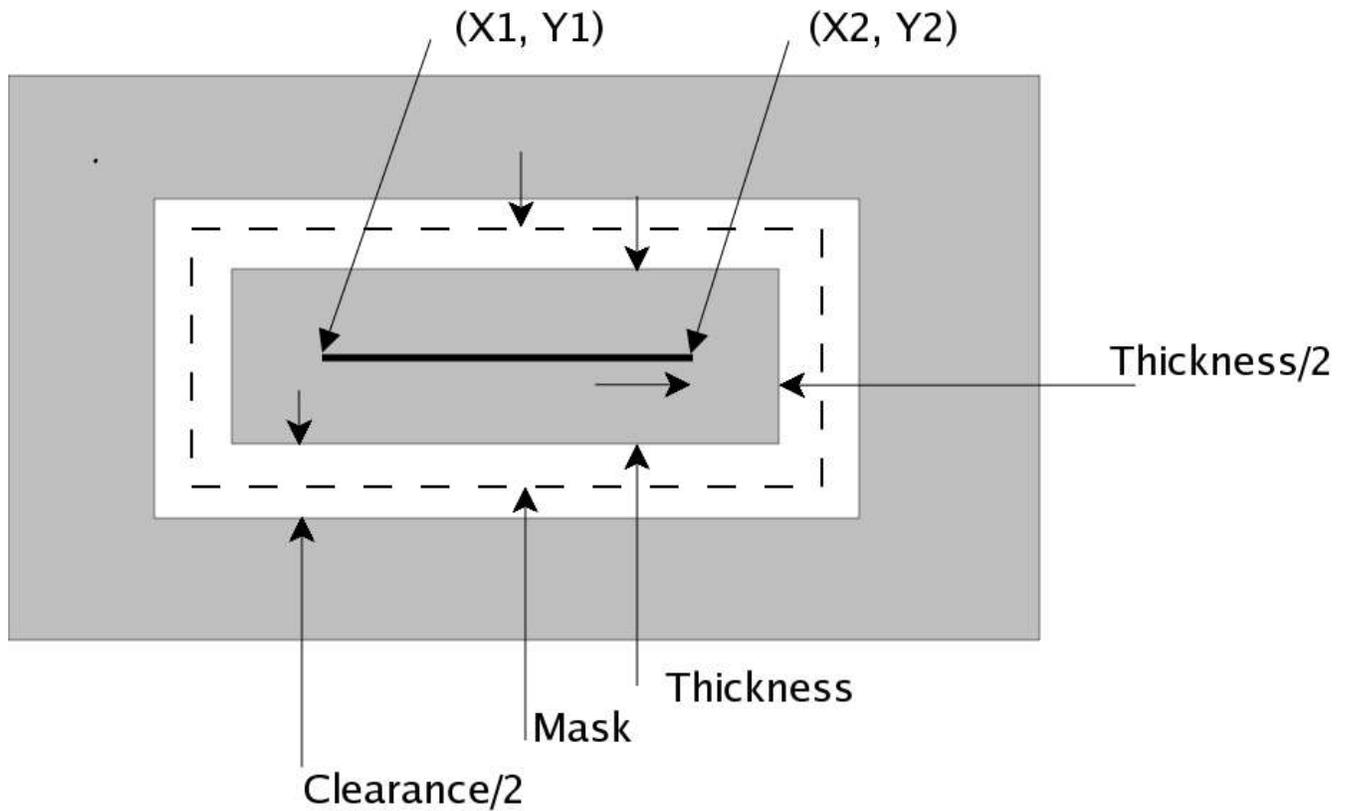
```
Pad [x1 y1 x2 y2 thickness clearance mask name pad_number flags]
```

Description détaillée

Objet	Valeur permise	Explication	Commentaire
x1	Decimal integer (mils or 1/100 mils)	Coord X du premier point de segment de ligne. Voir schéma.	
y1	Decimal integer (mils or 1/100 mils)	Coord Y du premier point de segment de ligne. Voir schéma.	
x2	Decimal integer (mils or 1/100 mils)	Coord X du second point de segment de ligne. Voir schéma.	
y2	Decimal integer (mils or 1/100 mils)	Coord Y du second point de segment de ligne. Voir schéma.	
thickness	Decimal integer (mils or 1/100 mils)	Largeur de métal entourant le segment de ligne. Voir schéma.	
clearance	Decimal integer (mils or 1/100 mils)	Séparation des pastilles des autres conducteurs pour les autres couches. Voir schéma.	C'est une séparation – pas une épaisseur. Notez aussi le facteur ½ en définition. Voir schéma.
mask	Decimal integer (mils or 1/100 mils)	Épaisseur du masque de soudure. Voir schéma.	L'épaisseur du masque de soudure est la zone autour de la pastille où le masque de soudure n'est pas appliqué.
name	string	Nom de la pastille. Chaîne d'identification arbitraire.	Peut être i.e. pad_1, plus, ou toute autre chaîne.
pad_number	string	Numéro de Pad. Utilisé pour l'utilisation des « rats », qui doit donc être cohérent avec la définition dans la netlist.	
flags	hex value	Défini dans la section « Flag » ci-dessous.	

Comme montré dans la figure ci-dessous, les dimensions de pastille sont définies par un segment de

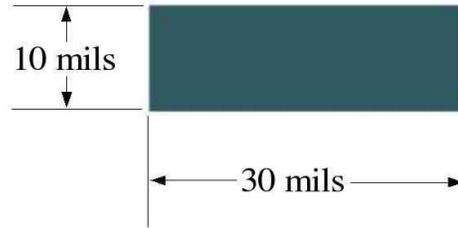
ligne avec des bords $(x1, y1)$ et $(x2, y2)$. Tous les autres paramètres sont définis en relation avec ce segment de ligne.



Notes

- Dans la version de PCB de développement 1.99o les points entrés $(x1,y1)$ et $(x2, y2)$ sont réordonnés de telle manière que $x1$ soit plus petit que $x1$ et $x2$. De la même manière, $y1$ devient la plus petite de $y1$ et $y2$.
- Les pastilles avec une épaisseur à zéro ne seront pas dessinées.

Exemple 1

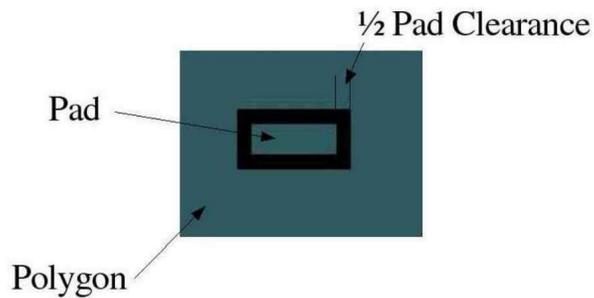


Pad (-10 0 10 0 10 0 0 "pad test" "1" 0x00000100)

Illustration 2Pad example 1

Cette pastille a été créée avec une ligne de 20 mil qui est orientée le long de l'axe x. La pastille complète fait 10 mils de plus long dû au paramètre d'épaisseur (thickness). Ce paramètre donne aussi une épaisseur de pastille de 10 mils le long de l'axe y. De manière à donner à une pastille une longueur particulière, vous avez besoin de soustraire le paramètre d'épaisseur depuis les points de début et de fin.

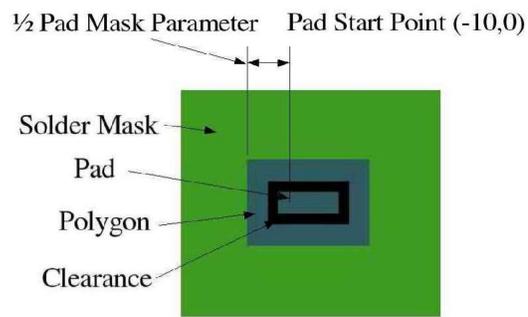
Exemple 2



Pad(-10 0 10 0 10 10 0 "pad test" "1" 0x00000100)

Clearance est la zone qui est vidée de tout polygone dans lequel est placé la pastille.

Exemple 3



Pad(-10 0 10 0 10 10 40 "pad test" "1" 0x00000100)

Il est important de noter que comme la métallisation de la pastille elle-même, l'épaisseur du masque de soudure est situé en fonction du segment de ligne sur lequel est placé la pastille.

Pin

L'élément Pin est contenu dans le corps de l'empreinte (Element). Il définit un trou simple avec un entourage métallique. La macro Pin est habituellement utilisée afin créer une empreinte pour un composant traversant. Elle peut aussi être utilisée pour créer un trou utilisé afin monter les composants sur la carte ou pour monter la carte elle-même.

Format

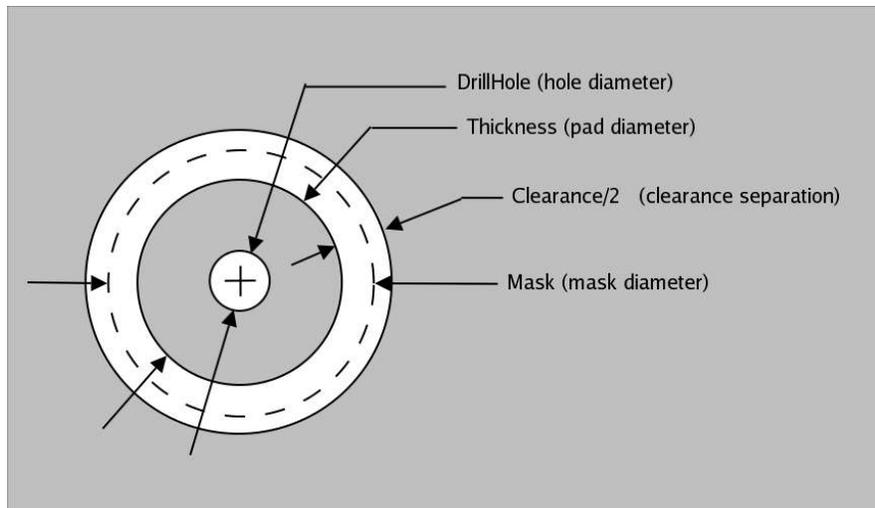
```
Pin(x y Thickness Clearance Mask DrillHole Name Number  
Flags)
```

```
Pin[x y Thickness Clearance Mask DrillHole Name Number  
Flags]
```

Description détaillée

Objet	Valeur permise	Explication	Commentaire
x	Decimal integer (mils or 1/100 mils)	Coordonnée x de la broche.	
y	Decimal integer (mils or 1/100 mils)	Coordonnée y de la broche.	
Thickness	Decimal integer (mils or 1/100 mils)	Diamètre de la métallisation entourante.	
Clearance	Decimal integer (mils or 1/100 mils)	Séparation du métal des autres conducteurs sur toutes les couches.	C'est une séparation – pas un diamètre. Notez aussi le facteur ½ dans la définition. Voir schéma.
Mask	Decimal integer (mils or 1/100 mils)	Diamètre de l'épaisseur du masque de soudure.	
DrillHole	Decimal integer (mils or 1/100 mils)	Diamètre des trous de perçage.	
Name	string	Nom du Pin. C'est un nom arbitraire pour la broche.	
Number	decimal integer	Numéro de Pin. Cette valeur est utilisée par PCB pour attacher les réseaux.	
Flags	hex value	Défini dans la section « Flag » ci-dessous.	

Les diverses dimensions définissant une broche sont montrées dans l'illustration ci-dessous.



Exemple

```
Pin[5400 -11200 8000 2000 9000 4300 "Pin_6" "6" 0x02004001]
```

Notez que les dimensions de cet exemple sont en **1/100 mil** car elles sont entre crochets.

ElementLine

La macro ElementLine dessine les segments de ligne sur la couche imprimée en association avec la couche sur laquelle le composant est placé (composant ou soudure).

Format

```
ElementLine(x1 y1 x2 y2 Thickness)
```

```
ElementLine[x1 y1 x2 y2 Thickness]
```

Description détaillée

Objet	Valeur permise	Explication	Commentaire
x1	decimal integer (mils or 1/100 mils)	Coordonnée x du point de démarrage du segment.	
y1	decimal integer (mils or 1/100 mils)	Coordonnée y du point de démarrage du segment.	
x2	decimal integer (mils or 1/100 mils)	Coordonnée x du point de fin du segment.	
y2	decimal integer (mils or 1/100 mils)	Coordonnée y du point de fin du segment.	
Thickness	decimal integer (mils or 1/100 mils)	Épaisseur du segment de ligne sur la couche imprimée.	

Exemple

```
ElementLine [-16000 -39100 59200 -39100 1000]
```

Notez que les dimensions utilisées dans cet exemple sont en **1/100 mil** car elles sont entre crochets.

ElementArc

Un ElementArc est habituellement utilisé pour dessiner un cercle ou un ovale sur la couche imprimée. Il peut aussi être utilisé pour dessiner un arc de cercle (i.e. un cercle incomplet) sur la couche imprimée. Si le composant est placé sur la couche haute de la carte, le cercle ou l'ovale est placé sur le côté haut (composant) de la carte. si le composant est placé sur la couche basse de la carte, le cercle ou l'ovale est dessiné sur l'arrière (soudure) de la carte.

Format

```
ElementArc(x y Width Height StartAngle Delta Thickness)
```

```
ElementArc[x y Width Height StartAngle Delta Thickness]
```

Description détaillée

<i>Item</i>	<i>Allowed value</i>	<i>Explanation</i>	<i>Comment</i>
x	Decimal integer (mils or 1/100 mils)	Position centrale X du cercle ou de l'ovale.	
y	Decimal integer (mils or 1/100 mils)	Position centrale Y du cercle ou de l'ovale.	
Width	Decimal integer (mils or 1/100 mils)	C'est la largeur horizontale du cercle ou de l'ovale.	Pour le cercle, utilisez Width = Height. Pour l'ovale, Width sera différent de Height.
Height	Decimal integer (mils or 1/100 mils)	Hauteur verticale du cercle ou de l'ovale.	
StartAngle	Decimal integer between 0 and 360 degrees.	Angle de démarrage de l'arc – mesuré en degrés dans le sens des aiguilles depuis l'axe X négatif (i.e. degrés dans le sens des aiguilles depuis le rayon horizontal pointant vers la gauche.)	
Delta	Decimal integer between 0 and 360 degrees.	Angle balayé par l'arc en degrés. La direction du balayage est dans le sens des aiguilles (CW).	360 pour les cercles complets ou les ovales. Les cercles incomplets – i.e. arcs – sont aussi possibles.
Thickness		Épaisseur d'un segment de ligne sur la couche imprimée.	

Exemple

```
ElementArc (350 -410 50 50 180 90 10)
```

Notez que les dimensions de cet exemple sont en **mil** car ils sont entre parenthèses.

Drapeaux (Flags) importants

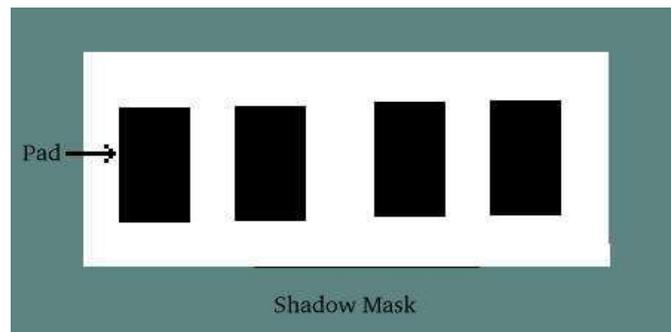
Certaines fonctionnalités des broches et des pastilles composant une empreinte sont considérées comme des bits individuels dans le champ « Flag ». Les bits flag sont filtrés par un « OU » avec une valeur hex seule qui est incorporée dans la déclaration de pastille ou de broche. Cette section liste quelques drapeaux qui sont importants pour la création d'empreintes. La liste complète des drapeaux peut être trouvée dans le code source de `~pcb/src/const.h`.

Description détaillée

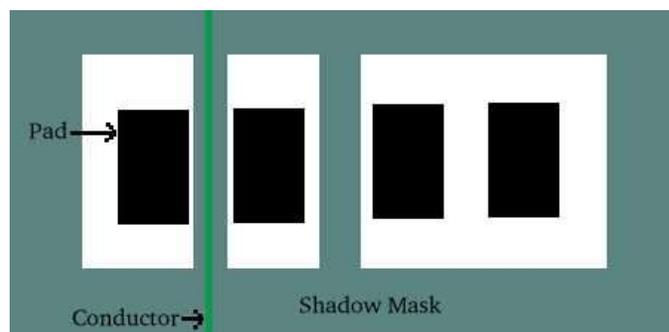
<i>Mnémonique</i>	<i>Valeur Hex</i>	<i>Explication</i>	<i>Commentaire</i>
NOFLAG	0x0000	Valeur NULL	
PINFLAG	0x0001	C'est une broche	
VIAFLAG	0x0002	C'est un via	
HOLEFLAG	0x0008	Cette broche ou via est seulement un trou.	
DISPLAYNAMEFLAG	0x0020	Affiche les noms des broches/pastilles.	
SQUAREFLAG	0x0100	Pin est carré, pas rond.	
USETHERMALFLAG	0x0400	Dessine broche et via avec les thermiques.	
OCTAGONFLAG	0x0800	Dessine broche ou via comme des octogones au lieu de ronds.	

Glossaire

- **Masque de soudure (Solder Mask)** – C'est une couche appliquée à la surface du PCB qui empêche la zone couverte d'être soudée. Habituellement, seules les broches de composant et les trous de broches sont laissées exposées. Les pistes laissées exposées peuvent être soudées par inadvertance.
- **Fenêtre de masque de soudure (Gang Solder Mask Window)** – Une fenêtre suffisamment large pour couvrir plus d'une pastille. Les pistes ne faisant pas parti du réseau, peuvent être soudées lorsqu'elles passent à côté d'une pastille.



- **Fenêtre de masque de soudure scindé (Pocket Solder Mask Window)** – Une fenêtre qui couvre une seule pastille. Cela nécessite des tolérances plus faibles dans la création d'un masque de soudure. Cela peut être nécessaire de manière à faire passer des pistes entre les pastilles.



- **Pistes (Trace)** – Fil conducteur qui route un signal entre deux ou plusieurs composants.