

KiCad

19 de octubre de 2016

Índice general

1. Introducción **1**

1.1. KiCad 1

1.2. Ficheros y directorios en KiCad 2

2. Instalación y configuración **4**

2.1. Opciones de visualización 4

2.2. Instalación de la configuración por defecto 4

2.3. Inicialización de algunas opciones y utilidades 4

2.4. Configuración de las rutas 5

2.5. Inicialización del editor de textos 6

2.6. Inicialización del visor de PDF 6

2.7. KiCad: principios de uso 6

3. Uso del gestor del proyecto en KiCad **8**

3.1. Ventana principal 8

3.2. Panel de utilidades 9

3.3. Vista de árbol del proyecto 9

3.4. Barra de herramientas superior 10

4. Usando plantillas **11**

4.1. Definicion 11

4.2. Plantillas 11

4.2.1. Metadatos 11

4.2.2. Ficheros Requeridos: 11

4.2.3. Ficheros Opcionales: 12

4.2.4. Ejemplo: 12

4.2.5. Operacion 13

4.2.6. Ubicación de las plantillas: 15

Manual de referencia

Copyright

Este documento está protegido por Copyright © 2010 a 2015 de sus autores, mencionados a continuación. Puede distribuirlo y/o modificarlo bajo los términos de la GNU General Public License (http://www.gnu.org/licenses/gpl.html), versión 3 o posterior, o la Creative Commons Attribution License (http : //creativecommons.org/licenses/by/3.0/), versión 3.0 o posterior.

Todas las marcas mencionadas en esta guía pertenecen a sus legítimos propietarios.

Contribuidores

Jean-Pierre Charras, Fabrizio Tappero.

Traducción

Antonio Morales <antonio1010.mr@gmail.com>, 2015

Realimentación

Por favor dirija cualquier reporte de fallo, sugerencia o nuevas versiones a:

- Acerca de la documentación de KiCad: https://github.com/KiCad/kicad-doc/issues
- Acerca del software KiCad: https://bugs.launchpad.net/kicad
- Acerca del software KiCad i18n: https://github.com/KiCad/kicad-i18n/issues

Fecha de publicación y versión del software

21 de Mayo de 2015.

Capítulo 1

Introducción

1.1. KiCad

KiCad es una herramienta software open-source para la creación de diagramas electrónicos y diseño de placas de circuito impreso. Bajo su singular fachada, KiCad incorpora un elegante conjunto con las siguientes herramientas software:

- **KiCad** : gestor del proyecto
- **Eeschema** : Editor de esquemas y componentes
- **CvPcb** : Selector de huellas (ejecutado siempre desde Eeschema)
- **Pcbnew** : editor de placa de circuito y editor de huellas
- **GerbView** : visor de ficheros Gerber

Se incluyen 3 herramientas

- **Bitmap2Component**: fabricante de componentes desde logotipos. Crea un símbolo de esquema o una huella a partir de una imagen de mapa de bits.
- **PcbCalculator**: Una calculadora útil para calcular componentes para los reguladores, ancho de las pistas en función a la corriente, lineas de transmisión…
- **PlEditor**: Editor del diseño de página.

Estas herramientas están generalmente se ejecuta desde el director del proyecto, pero pueden ser también funcionan como herramientas independientes.

En este momento, KiCad se considera suficientemente maduro y puede utilizarse para el desarrollo y mantenimiento exitoso de PCB's complejas.

KiCad no presenta limitación alguna en cuanto al tamaño de la placa y puede gestionar hasta 32 capas de cobre, 14 capas técnicas y 4 capas auxiliares.

KiCad puede crear todos los archivos necesarios para la construcción de placas de circuito impreso:

- Archivos Gerber o fotolitos
- archivos de taladrado
- archivos de ubicación de componentes
- y mucho mas.

Al ser de código abierto (licencia GPL), KiCad representa la herramienta ideal para proyectos orientados a la creación de equipos electrónicos con estilo open-source.

KiCad está disponible para Linux, Windows y Apple OS X (aún experimental, pero funcionando bien)

1.2. Ficheros y directorios en KiCad

KiCad crea y utiliza archivos (y directorios) con las siguientes extensiones de archivos específicas para la edición de esquemas y placas.

Fichero del Gestor del proyecto:

*.pro	Archivo que contiene algunos parámetros para el proyecto actual, incluyendo la lista de bibliotecas de componentes.

Archivos del Editor de Esquemas

*.sch	Archivos de esquemas, que no contienen los componentes en si mismos.
*.lib	Archivos e bibliotecas de símbolos, que contienen las descripciones de los componentes: forma gráfica, pines, campos.
*.dcm	documentación de las bibliotecas de símbolos, contiene algunas descripciones de componentes: comentarios, palabras clave, referencia a las hojas de características.
*_cache.lib	archivo de caché para las bibliotecas de símbolos, que contiene una copia de los símbolos de los componentes utilizados en el esquema del proyecto proyecto.

Ficheros y directorios del Editor de placas:

*.kicad_pcb	Archivo de la placa que contiene toda la información del diseño de la página.
*.pretty	**Directorio de las bibliotecas** de huella. La carpeta es la biblioteca en sí.
*.kicad_mod	Archivos de huella, que contiene una descripción una huella de cada uno.
*.brd	archivo de la placa en el formato antiguo. Puede leerse, pero no ser escrito por el editor de la placa.
*.mod	biblioteca Huella en el formato antiguo. Puede ser leído por la huella o el editor de la placa, pero no ser escrito.
fp-lib-table	lista de bibliotecas de huella (*tabla de bibliotecas de huellas*: Lista de bibliotecas de componentes (en varios formatos) que son cargadas por la placa, el editor de huellas o por CvPcb

Ficheros comunes

*.kicad_wks	Archivos de descripción del diseño de página, para la gente que quiere un diseño de hoja personalizado.
*.net	Archivo de Netlist creado por el esquema, y leído por el editor de la placa. Este archivo esta asociado al archivo .cmp, para los usuarios que prefieren archivos separados para la asociación de componentes/huella.

Ficheros especiales

*.cmp	Almacenan la asociación entre los componentes usados en el esquema y sus huellas. Puede ser creado por Pcbnew, e importado por Eeschema. El objetivo es una importación hacia atrás desde Pcbnew a Eeschema, para los usuarios que cambian las huellas dentro Pcbnew (por ejemplo utilizando el comando *Exchange Footprints*) y quieren importar estos cambios en el esquema.

Otros ficheros

Son generados por KiCad para fabricación o documentación.

*.gbr	Ficheros Gerber, para fabricación
*.drl	Ficheros de taladrado (Formato Excellon), para fabricación.
*.pos	Ficheros de posición files (formato ascii), para maquinas de inserción automáticas.
*.rpt	Ficheros de informes (formato ascii), para documentación.
*.ps	Fichero de impresión (postscript), para documentación.
*.pdf	Fichero de impresión (formato pdf), para documentación.
*.svg	Fichero de impresión (formato svg), para documentación.
*.dxf	Fichero de impresión (formato dxf), para documentación.
*.plt	Fichero de impresión (formato HPGL), para documentación.

Capítulo 2

Instalación y configuración

2.1. Opciones de visualización

Pcbnew necesita OpenGL v2.1 o superior.

2.2. Instalación de la configuración por defecto

Por defecto se suministra un archivo de configuración llamado *kicad.pro* en kicad/template. Sirve como plantilla para cualquier nuevo proyecto.

Si existe otro archivo de configuración por defecto *fp-lib-table*, será usado sólo una vez para crear una lista de bibliotecas de huellas. (o de lo contrario, esta lista se creará a partir de cero)

El archivo por defecto *kicad.pro* puede modificarse libremente si fuera necesario, principalmente para establecer la lista de archivos de biblioteca cargados por Eeschema.

Además unos cuantos parámetros (el tamaño del texto por defecto, grosor de línea por defecto, principalmente para Pcbnew) se almacenan aquí.

Verifique que tiene acceso de escritura a kicad/template/kicad.pro

Ejecute KiCad y cargue el proyecto *kicad.pro*

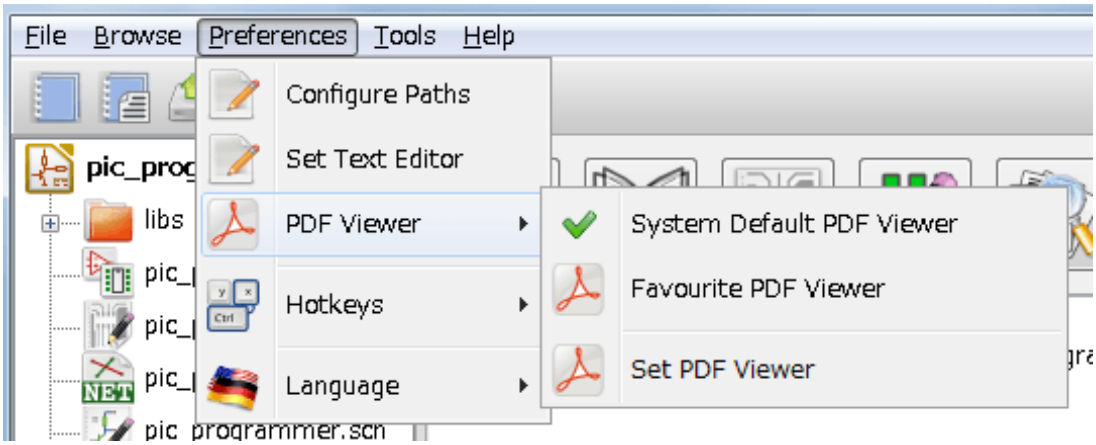
Ejecute Eeschema a través KiCad. Modifique y actualice la configuración de Eeschema, y sobre todo la lista de bibliotecas que desea utilizar cada vez que cree nuevos proyectos.

Ejecute Pcbnew a través KiCad. Modifique y actualice la configuración de Pcbnew, especialmente la lista de bibliotecas de huellas. Pcbnew creará o actualizará un archivo de lista de la biblioteca llamada **tabla de bibliotecas de huellas**. Hay 2 archivos de lista de la biblioteca (con nombre fp-lib-table). El primero (que se encuentra en el directorio principal del usuario) es global para todos los proyectos. La segunda, si es que existe (que se encuentra en el directorio del proyecto) es específica para el proyecto.

2.3. Inicialización de algunas opciones y utilidades

Cuando se usa KiCad, son útiles un editor de texto y un visor de PDF. Ajustarlos es una buena idea.

Estos ajustes están accesibles desde el menú de preferencias



3 opciones son especialmente importantes:

- Configurar las rutas
- Visor de PDF
- Ajustar el editor de texto

2.4. Configuración de las rutas

En KiCad, se puede definir algunas rutas usando *variable del entorno*. Unas pocas variables del entorno se definen internamente por KiCad, y pueden ser utilizados para definir rutas de acceso (para bibliotecas, modelos en 3D, etc).

Esto es útil cuando las rutas absolutas no son conocidos o están sujetas a cambios. Este es el caso de las bibliotecas “oficiales” construidas para KiCad:

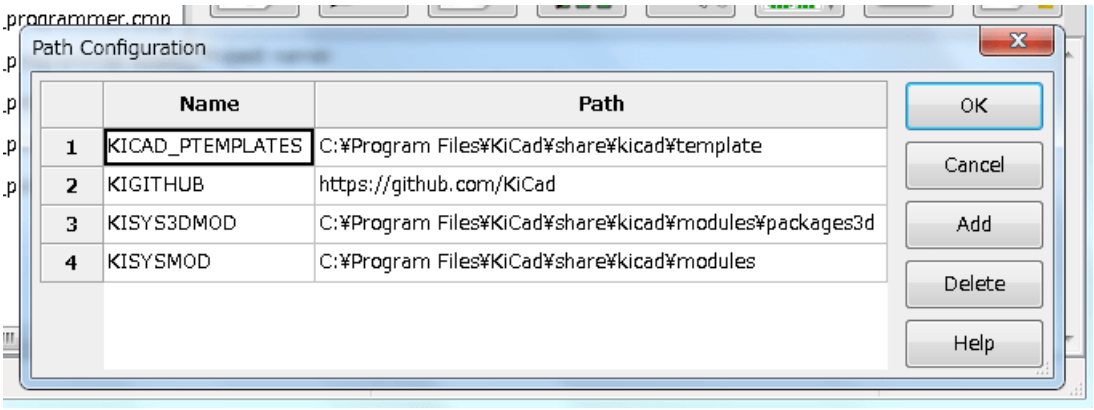
- por las rutas a estas bibliotecas, cuando se instalan en el disco
- para las rutas a los modelos 3D utilizados en las definiciones de la huellas.

Por ejemplo, la ruta completa de la biblioteca de huellas *connect.pretty* se define de esta manera cuando se utiliza la variable de entorno KISYSMOD para definir la ruta completa: ${KISYSMOD}/connect.pretty

Obviamente, se puede utilizar como definición su ruta completa, si esta ruta completa es bien conocida, y nunca cambia.

Esta opción le permite definir algunas rutes en una variable de entorno, y añadir sus propias variables del entorno, para definir otras rutas personales, si fuera necesario.

KIGITHUB	utilizado con frecuencia en los ejemplos de tablas de bibliotecas de huellas. Si utiliza esta variable, debe ser definida.
KISYS3DMOD	ruta por defecto para los archivos de modelos 3D de componentes. Debe ser definida, ya que las rutas absolutas normalmente no se utilizan.
KISYSMOD	ruta por defecto de las carpetas de la bibliotecas de huellas. Debe ser definida si no se utilizan rutas absolutas en los nombres de las bibliotecas de huellas.



Tenga en cuenta también la variable del entorno

- **KIPRJMOD**

es **siempre** definida internamente por KiCad, y es la **ruta absoluta proyecto actual**.

Por ejemplo, ***${KIPRJMOD}/connect.pretty*** es siempre el directorio ***connect.pretty*** (la libreria de huellas pretty) que se encuentra ***dentro del directorio del proyecto actual***

- **Si modifica la configuración de las rutas, por favor, cierre y vuelva a ejecutar KiCad para evitar cualquier problema en el uso de las rutas.**

2.5. Inicialización del editor de textos

Antes de usar un editor de texto para ver/editar archivos en el proyecto actual, tiene que elegir el editor de texto que desea utilizar. El menú:

Preferences/SetText Editor

le permite ajustar que editor de texto usar.

2.6. Inicialización del visor de PDF

Puede utilizar el visor de PDF predeterminado o elegir su propio visor de PDF

Seleccione *Preferencias/Visor PDF/Visor PDF Favorito* para elegir su visor de PDF preferido

Su propio (no por defecto) visor se elige mediante el menú *Preferencias/Visor PDF/Ajustar Visor PDF*

En Linux, el visor PDF predeterminado es conocido por ser a veces extraño, así que usar *Visor de PDF Favorito* después de seleccionar un visor de PDF adecuado es casi obligatorio.

2.7. KiCad: principios de uso

Con el fin de gestionar un proyecto KiCad: archivos de esquemas, archivos de la tarjeta de circuito impreso, bibliotecas, archivos complementarios de fabricación para foto-trazado, taladrado y archivos de colocación automática de componentes, se recomienda crear un proyecto de la siguiente manera:

- **Crear un directorio de trabajo para el proyecto** (usando KiCad u otros medios).
- **En este directorio, utilice KiCad para crear un archivo de proyecto** (archivo con extensión .pro) a través de los iconos de opción ”Crear un nuevo proyecto” o ”Crear un nuevo proyecto desde plantilla”.



aviso

Se recomienda usar un directorio único para cada proyecto de KiCad. No combine múltiples proyectos en un solo directorio.

KiCad crea un archivo con extensión .pro que mantiene una serie de parámetros para la gestión del proyecto (como la lista de las bibliotecas utilizadas en el esquema). Los nombres por defecto de los archivos principales, tanto archivo del esquema principal como el archivo de placa de circuito impreso, se derivan del nombre del proyecto. Por lo tanto, si un proyecto llamado *ejemplo.pro* fue creado en un directorio llamado *ejemplo*, se crearán los archivos predeterminados:

ejemplo.pro	archivo de gestión del proyecto.
ejemplo.sch	archivo del esquema principal.
ejemplo.kicad_pcb	archivo de placa de circuito impreso.
ejemplo.net	archivo de netlist.
ejemplo.xxx varios archivos creados por los otros programas de utilidades.	ejemplo-cache.lib

Capítulo 3

Uso del gestor del proyecto en KiCad

El gestor del proyecto en KiCad (kicad o archivo kicad.exe) es una herramienta desde donde puede ejecutar fácilmente las otras herramientas (editores, visor de Gerber y herramientas de utilidades) al crear un diseño.

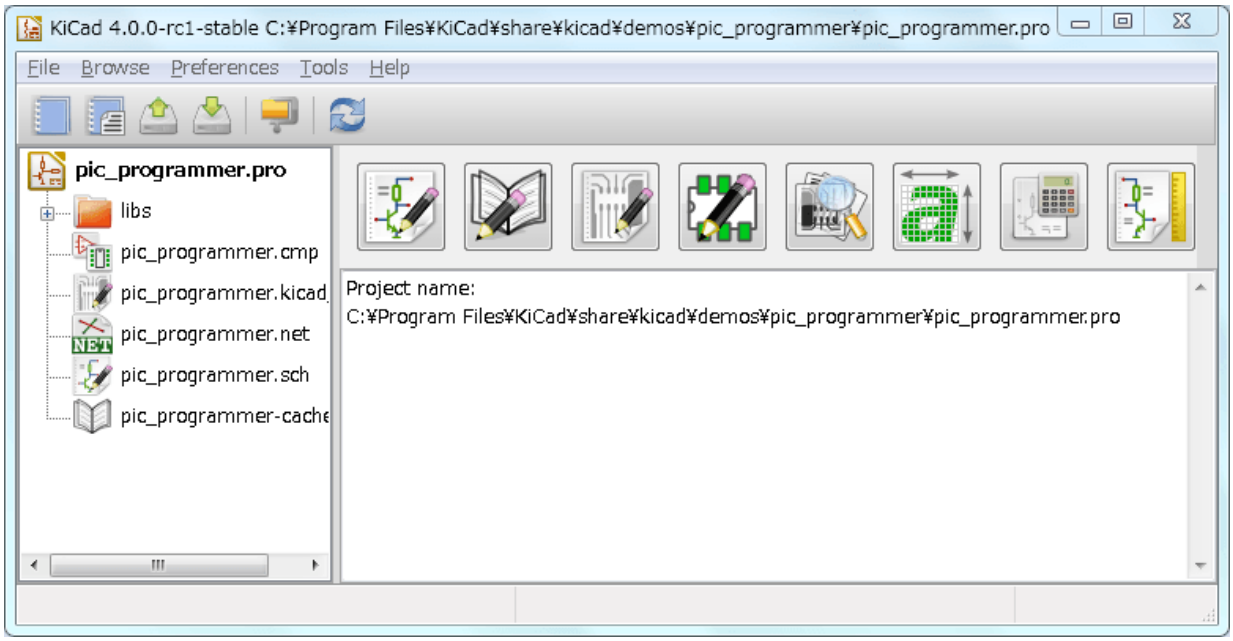
Ejecutar los otros programas desde el gestor del proyecto de KiCad tiene algunas ventajas:

- control cruzado entre el editor de esquemas y el editor de placas de circuito
- control cruzado entre el editor de esquemas y el selector de huellas (CvPcb).

Pero solo se pueden editar los ficheros del proyecto actual.

(Cuando estas herramientas se ejecutan en modo *autónomo*, puede abrir cualquier fichero de cualquier proyecto pero el control cruzado entre herramientas puede dar resultados inesperados)

3.1. Ventana principal

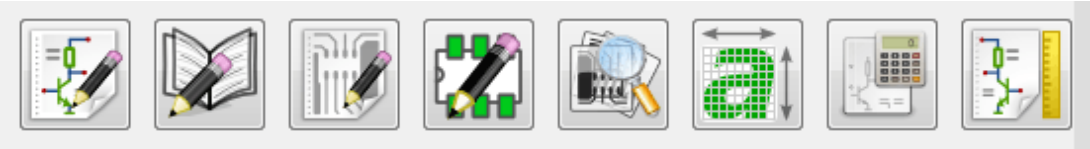


La ventana principal KiCad se compone de una vista tipo árbol del proyecto, un panel que contiene botones que sirven para ejecutar las distintas herramientas del software, y una ventana de mensaje. El menú y la barra de herramientas se pueden utilizar para crear, leer y guardar archivos de proyecto.

3.2. Panel de utilidades

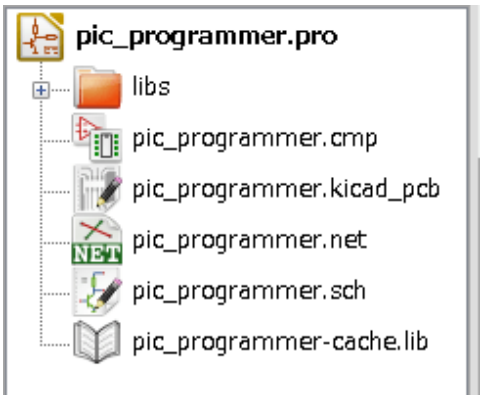
KiCad le permite ejecutar todas la herramientas software autónomas que están incluidas en el.

El panel de herramientas esta compuesto por 8 botones que corresponden a los siguientes comandos (de 1 a 8, de izquierda a derecha)



1	**Eeschema**	El editor de esquemas.
2	**LibEdit**	El editor de símbolos de componentes y gestor de bibliotecas de símbolos.
3	**Pcbnew**	El editor del diseño de la placa de circuito.
4	**FootprintEditor**	El editor de huellas y gestor de bibliotecas de huellas.
5	**Gerbview**	Un visor de archivo GERBER. También puede visualizar archivos de taladrado.
6	**Bitmap2component**	Una herramienta para crear huellas o simbolos a partir de imagenes de bitmap para crear logos.
7	**Pcb Calculator**	Una herramienta para calcular anchos de pistas y otras cosas.
8	**Pl Editor**	Un editor para el diseño de la pagina. para crear/personalizar el cuadro de referencia.

3.3. Vista de árbol del proyecto



- Haciendo doble clic sobre el icono de Eeschema se ejecutará el editor de esquemas que en este caso abrirá el fichero pic_programmer.sch.
- Haciendo doble clic sobre el icono de Pcbnew se ejecutará el editor del diseño, en este caso abrirá el archivo pic_programmer.kicad_pcb.
- Haciendo clic derecho sobre cualquiera de los archivos en el árbol del proyecto le permite ver las acciones genéricas de manipulación del archivo.

3.4. Barra de herramientas superior



La barra de herramientas superior le permite realizar ciertas acciones básicas sobre los archivos (de izquierda a derecha)

	Crea un archivo de proyecto. Si se encuentra la plantilla kicad.pro dentro de kicad/ template, se copia dentro del directorio de trabajo.
	Crea un proyecto partiendo de una plantilla.
	Abre un proyecto existente.
	Actualiza y guarda el árbol del proyecto actual.
	Crea un archivo zip con el proyecto actual completo. Esto incluye ficheros de esquemas, bibliotecas, pcb, etc.
	Reconstruye y redibuja el árbol del proyecto, aveces es necesario tras un cambio en el árbol

Capítulo 4

Usando plantillas

4.1. Definicion

Una plantilla es un directorio de archivo, que incluye un directorio de metadatos.

El nombre del sistema template (SYSNAME) es el directorio bajo el que se almacenan los ficheros de plantillas. El directorio metadata (METADIR) contiene ficheros predefinidos que proveen información sobre las plantillas.

Todos los fichero y directorios en una plantilla son copiados a la ruta del nuevo proyecto cuando se crea un proyecto usando una plantilla, a excepción de METADIR.

Todos los ficheros y directorios que comienzan por SYSNAME remplazarán SYSNAME por el nuevo nombre del proyecto, excluyendo la extensión de archivo.

4.2. Plantillas

Estas facilitan una rápida creación de proyectos que tienen atributos comunes tal como formas de la placa del circuito predefinida, posiciones de conectores, símbolos, reglas de diseño, etc.

4.2.1. Metadatos

El METADIR de una plantilla debe contener ciertos archivos requeridos, y opcionalmente otros ficheros.

4.2.2. Ficheros Requeridos:

meta/info.html

contiene información sobre la plantilla en formato html que es usada por el usuario para comprobar como será el proyecto aplicando la plantilla. La etiqueta <title> determina el nombre actual de la plantilla que sera mostrado al usuario en la selección de plantillas.

Usando html como formato de este documento implica que las imágenes pueden estar en linea sin necesidad de inventar un nuevo esquema.

Solo puede usar lenguaje HTML básico para formatear este documento.

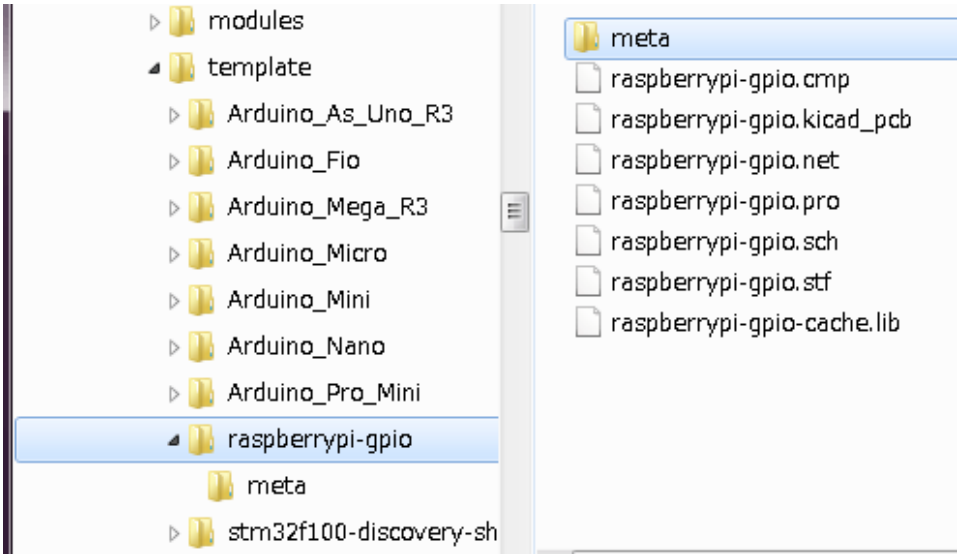
4.2.3. Ficheros Opcionales:

meta/icon.png

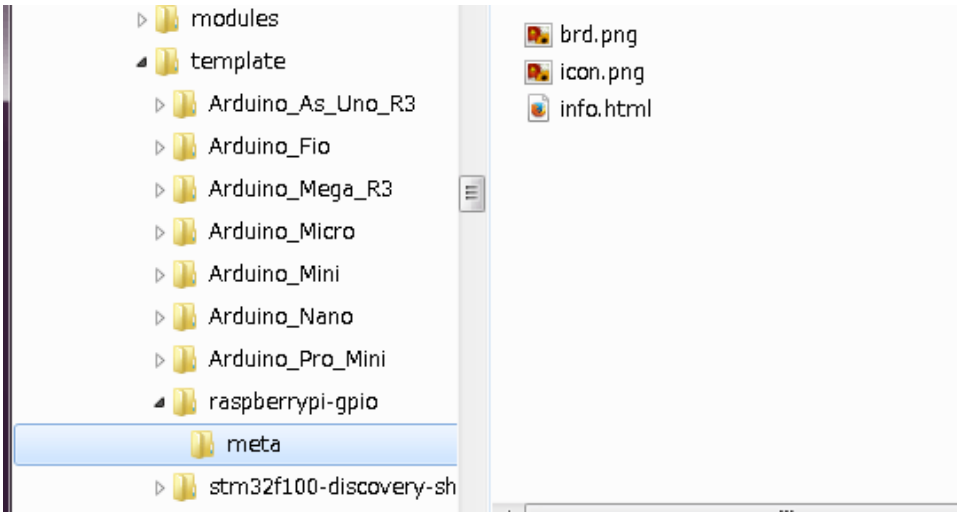
Un icono en formato PNG de 64 x 64 pixels que es usado como icono seleccionable dentro de la ventana de selección de plantillas.

4.2.4. Ejemplo:

Aquí se muestra una plantilla de una placa para raspberrypi-gpio:



Y la información de metadata:



brd.png es un fichero opcional.

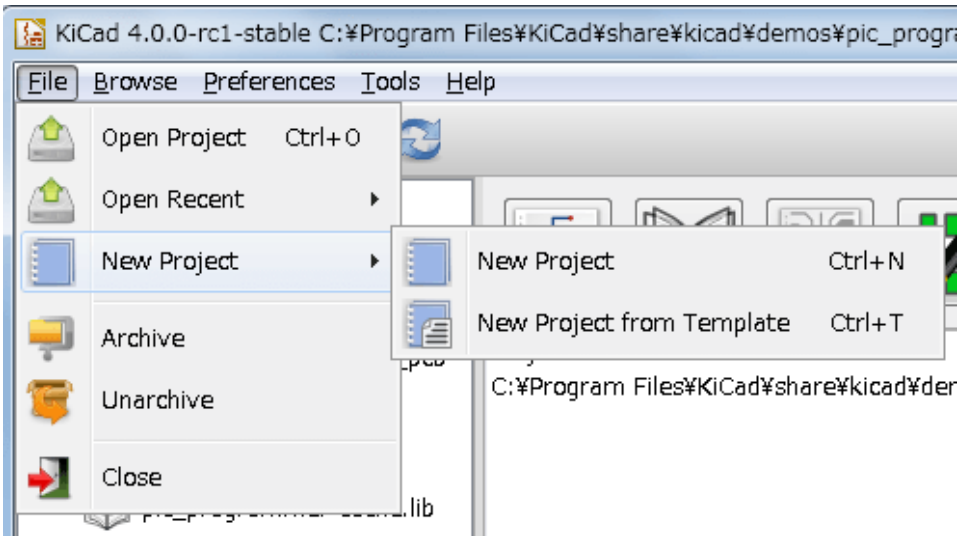
Aquí se muestra un fichero info.html de muestra:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
```

```
<META HTTP-EQUIV="CONTENT-TYPE" CONTENT="text/html;
charset=windows-1252">
<TITLE>Raspberry Pi - Expansion Board</TITLE>
<META NAME="GENERATOR" CONTENT="LibreOffice 3.6 (Windows)">
<META NAME="CREATED" CONTENT="0;0">
<META NAME="CHANGED" CONTENT="20121015;19015295">
</HEAD>
<BODY LANG="fr-FR" DIR="LTR">
<P>This project template is the basis of an expansion board for the
<A HREF="http://www.raspberrypi.org/" TARGET="blank">Raspberry Pi $25
ARM board.</A> <BR><BR>This base project includes a PCB edge defined
as the same size as the Raspberry-Pi PCB with the connectors placed
correctly to align the two boards. All IO present on the Raspberry-Pi
board is connected to the project through the 0.1&amp;quot; expansion
headers. <BR><BR>The board outline looks like the following:
</P>
<P><IMG SRC="brd.png" NAME="brd" ALIGN=BOTTOM WIDTH=680 HEIGHT=378
BORDER=0><BR><BR><BR><BR>
</P>
<P>(c)2012 Brian Sidebotham<BR>(c)2012 KiCad Developers</P>
</BODY>
</HTML>
```

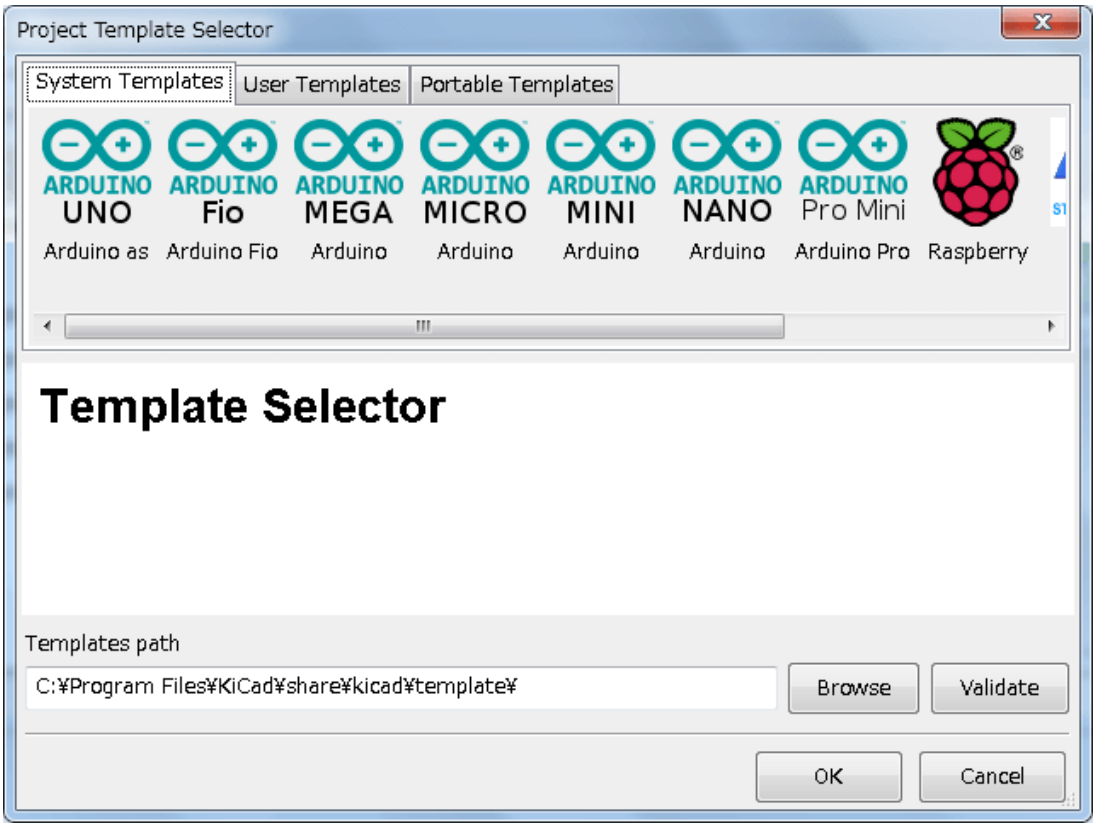
4.2.5. Operacion

El menú Archivo de KiCad muestro 2 opciones:

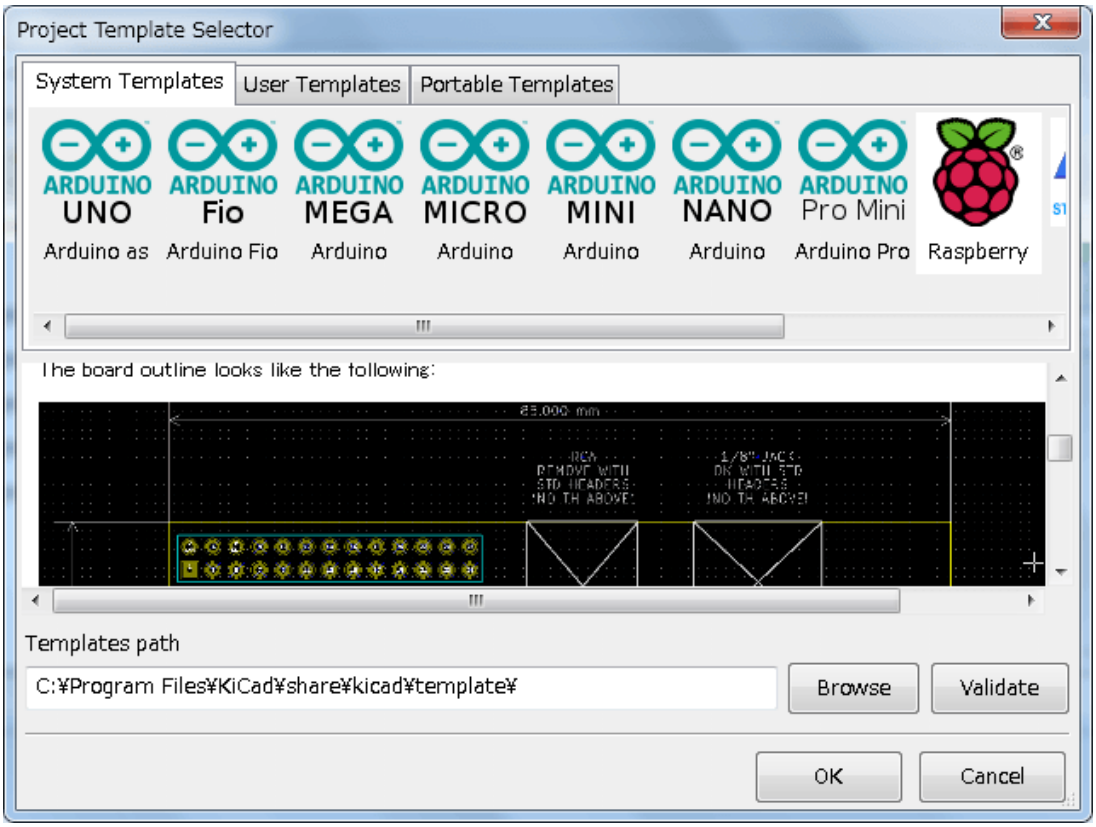


- **Nuevo proyecto** Crea un proyecto en blanco mediante la copia de template/kicad.pro al directorio de trabajo actual.
- **Proyecto desde plantilla** Abre la ventana de selección de plantilla. La ventana de selección de plantillas tiene una lista de iconos, y una ventana de visualización. Un clic sobre el icono de una plantilla en la parte superior cargará el archivo de metadatos info.html y lo mostrará en la ventana de visualización. Un clic en el botón OK inicia creación del nuevo proyecto. La plantilla será copiada

a la nueva ubicación del proyecto (excluyendo MetaDir como se ha mencionado anteriormente) y los archivos que coinciden con las reglas de sustitución de cadenas serán renombrados para reflejar el nombre del nuevo proyecto.



Después de seleccionar una plantilla:



4.2.6. Ubicación de las plantillas:

La lista de las plantillas disponibles se recopila de las siguientes fuentes:

- Para plantillas del sistema: <kicad bin dir>/../share/template/
- Para plantillas de usuario:
  - en Unix: ~/kicad/templates/
  - en Windows: C:\Documents and Settings\username\My Documents\kicad\templates
  - en Mac: ~/Documents/kicad/templates/
- Cuando se define la variable del sistema KICAD_PTEMPATES, existe una tercera pagina: Plantillas portables, cuya lista de plantillas se encuentra en la ruta KICAD_PTEMPLATES.