

Distribución Linux para electrónicos

Miguel Pareja Aparicio

José Rafael Lajara Vizcaíno

José Pelegrí Sebastián

En el presente artículo nos encontramos con un binomio, que integra las variables: uso de una distribución Linux y aplicaciones para la electrónica. Tras la resolución de la ecuación nos encontramos con la distribución FEL (Fedora Electronic Lab). Dicha distribución presenta la ventaja de ser utilizada para la formación de electrónica sin la necesidad de entrar en detalle en los procesos de instalación de las mismas, además de la posibilidad de disponer de dicha distribución en una memoria portátil para aumentar la portabilidad del trabajo realizado, ya que se dispone de todos los programas necesarios para crear o editar los proyectos relacionados con la electrónica que puedan surgir.

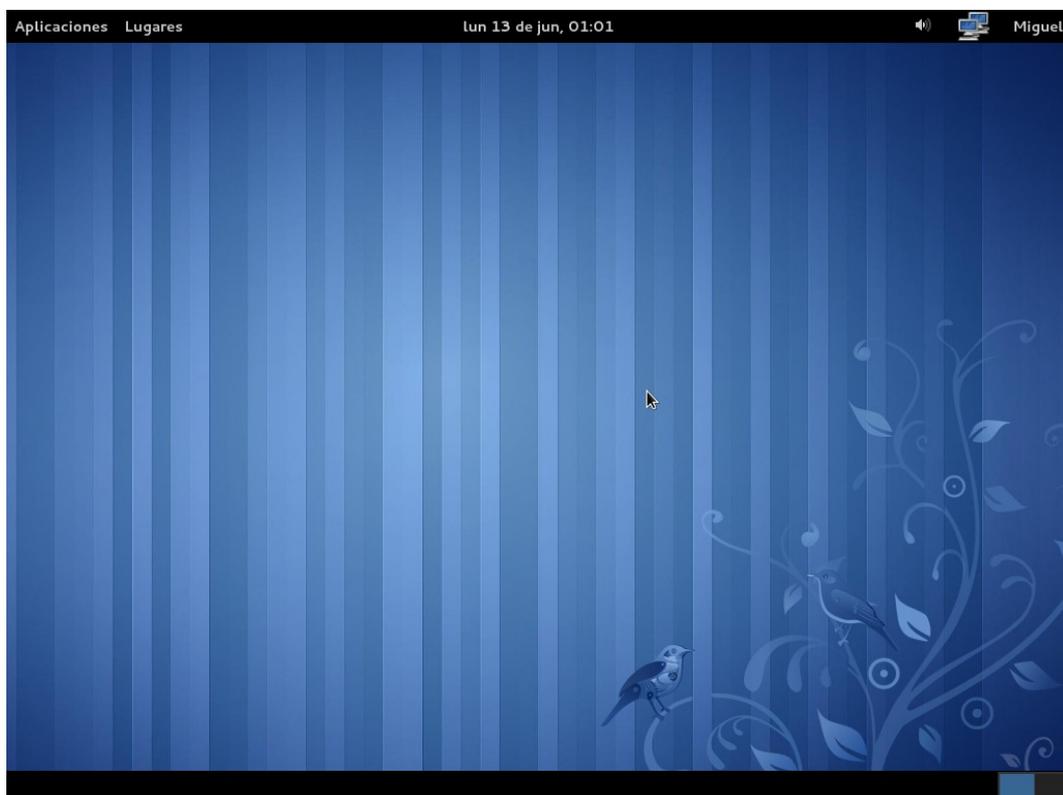


Figura 1.- Escritorio FEL

1.- Introducción

FEL (<http://spins.fedoraproject.org/fel/>) se trata de una distribución basada en Fedora (<http://fedoraproject.org/es/>) que integra una serie de aplicaciones preinstaladas para ingenieros electrónicos. Esto es especialmente interesante para formación, puesto que ofrece directamente el uso de la aplicación sin necesidad de recurrir a las particularidades de instalación de una u otra aplicación, sobre todo para posibles usuarios ajenos al manejo de Linux. Su traducción directa es laboratorio de electrónica de Fedora (a partir de ahora LEB), y como su propio nombre indica es un laboratorio virtual de electrónica, con

aplicaciones relacionadas con el campo: de la simulación de circuitos electrónicos, el diseño de placas de circuito impreso, diseño con microcontroladores y diseño microelectrónico (VLSI).

Se ofrece la posibilidad de disponer de un cd-live para realizar sus pruebas así como la posibilidad de disponer de dicha distribución en un dispositivo de memoria USB, convirtiéndolo en una buena alternativa a otras aplicaciones concretas en la realización de cursos o prácticas. Por ejemplo la oferta de un USB con toda la documentación y la distribución preinstalada, con las aplicaciones a utilizar, no cabe duda que el realizar un curso con aplicaciones libres o gratuitas favorece el precio del mismo, puesto que se evita el gasto adicional de la compra de licencias o el uso de versiones de demostración o limitadas.

En la figura 2, se muestran a modo de ejemplo un conjunto de capturas de las aplicaciones para el diseño electrónico.

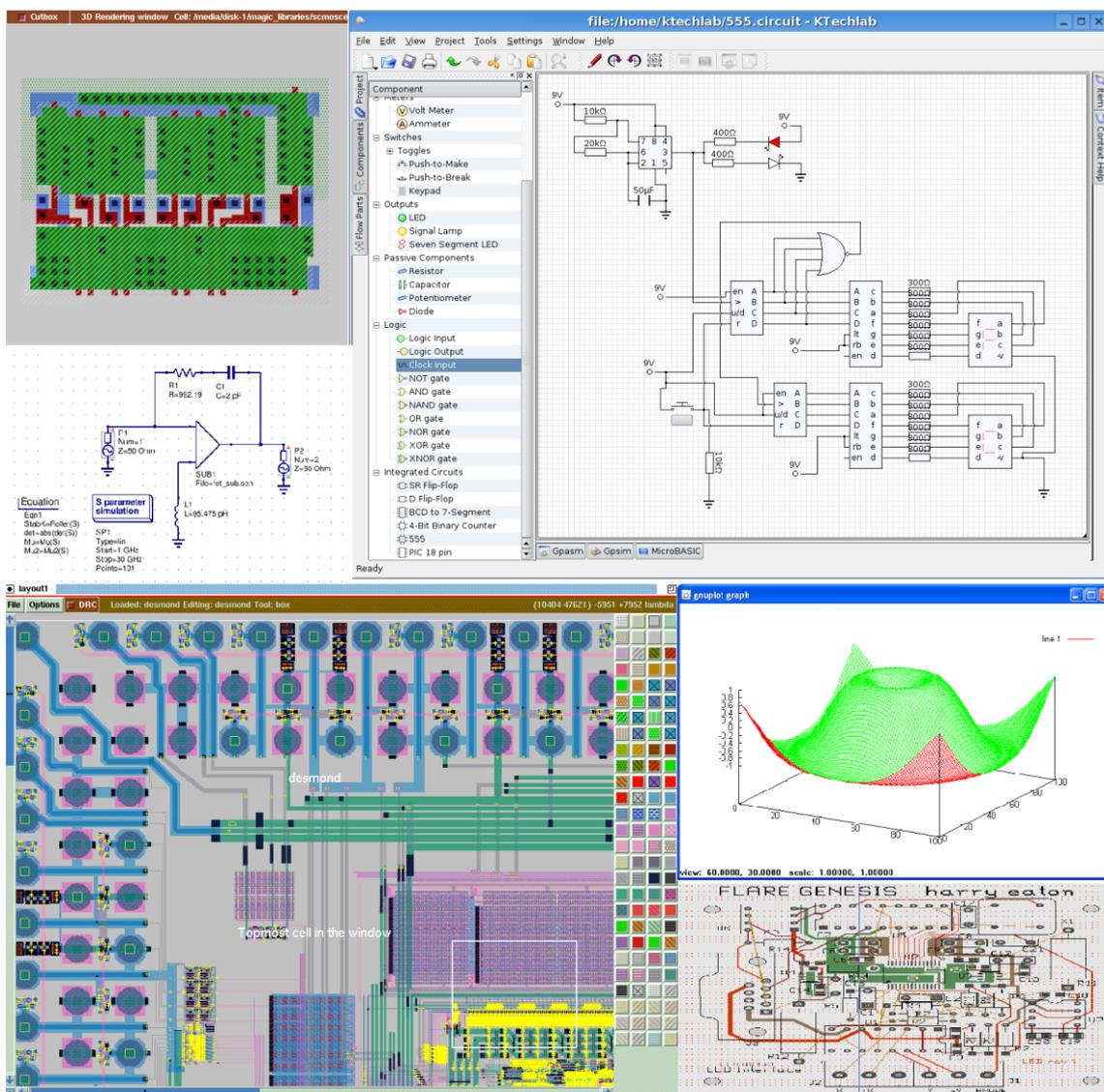


Figura 2.- Aplicaciones LEF

2.- Laboratorio de electrónica Fedora

2.1.- Colección de aplicaciones y acceso

El LEF está destinado principalmente al campo de la ingeniería micro-nano electrónica, porque puede ser utilizado por fabricantes de semiconductores con planta propia o fables para la fabricación de obleas de silicio para el diseño y comercialización de chips. Mediante aplicaciones para VLSI (Very Large Scale Integration) y VHDL: Magic VLSI LAYOUT (<http://opencircuitdesign.com/magic/index.html>), TOPED (<http://www.toped.org.uk>) o ALLIANCE (<http://www-asim.lip6.fr/recherche/alliance>).

Aunque también encontrará aplicaciones para la formación de electrónica (analógica y/o digital), mediante aplicaciones de simulación tales como: gEDA (<http://www.geda.seul.org>) o QUCS (<http://qucs.sourceforge.net>); o diseño de placas de circuito impreso tales como: KICAD (www.lis.inpg.fr/realise_au_lis/kicad/) o PCB LAYOUT (<http://pcb.sourceforge.net>).

Del mismo modo también se puede recurrir a la distribución LEF para la formación de sistemas con microcontroladores, mediante aplicaciones de simulación tales como: KTECHLAB (<http://sourceforge.net/projects/ktechlab/files/>) o GPSIM (<http://sourceforge.net/projects/gpsim/files/>); así como herramientas de desarrollo de microcontroladores PIC, tal como: PIKLAB (<http://piklab.sourceforge.net>). También puede ser de utilidad una aplicación específica para la representación de esquemas CAD (XCIRCUIT -<http://opencircuitdesign.com/xcircuit/>-) o un generador de código ensamblador para retardos (PIKLOOP -<http://pikloops.sourceforge.net>-).

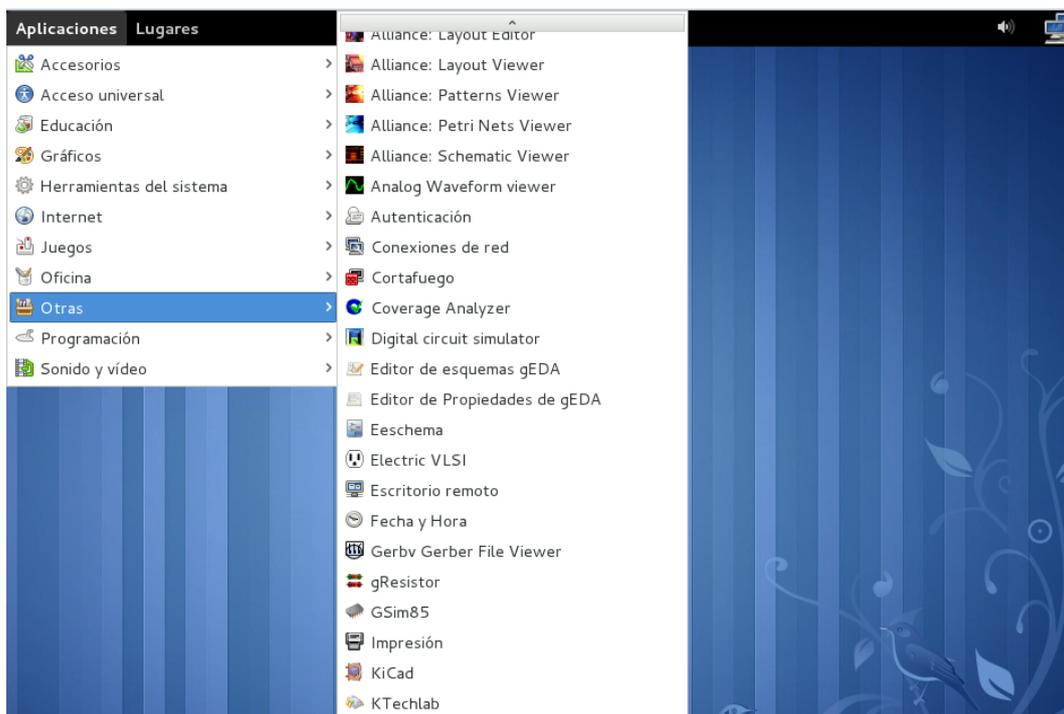


Figura 3.- Aplicaciones para electrónica

Como habrá podido leer, en LEF se dispone de un amplio abanico de aplicaciones relacionadas con el desarrollo de circuitos electrónicos. Se han comentado las más representativas, no sólo para ingenieros electrónicos sino que también para técnicos medios o superiores relacionados con el campo de la electrónica, así como para aficionados con

carreras relacionadas con el campo de la electrónica. Para el acceso de dichas aplicaciones se dispone del menú “Otras” (figura 3).

Dispone también de una aplicación de cálculo numérico, similar a Matlab, denominado Octave (OCTAVE -<http://octave.org>-), a la que se puede acceder a través del menú “Educación” (figura 4).

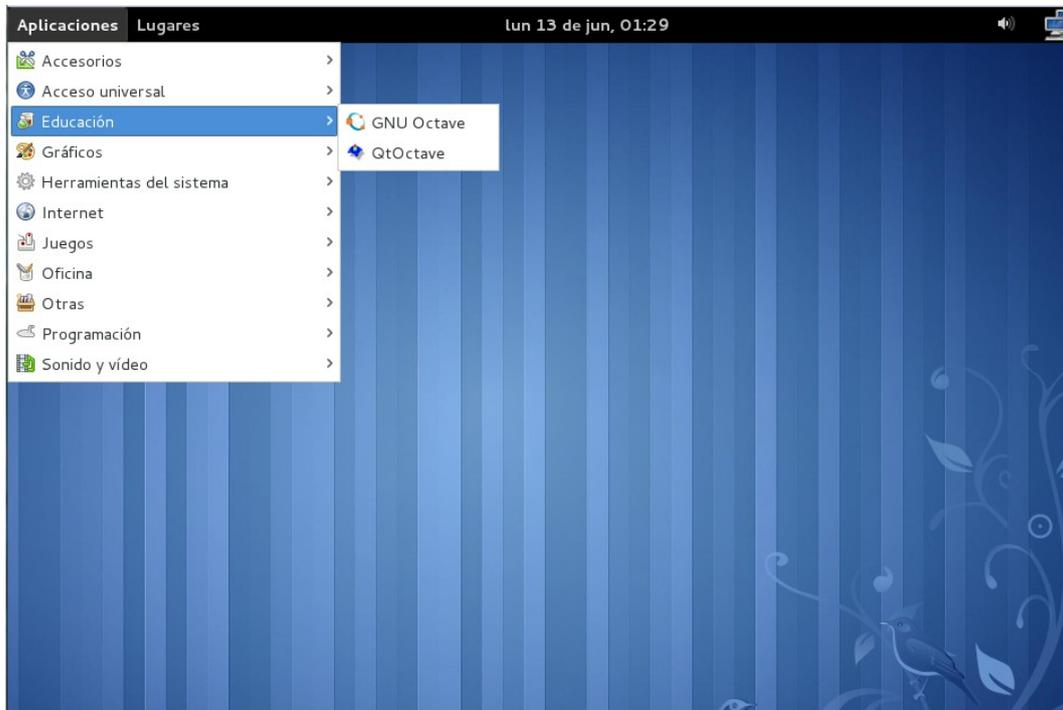


Figura 4.- Acceso Octave

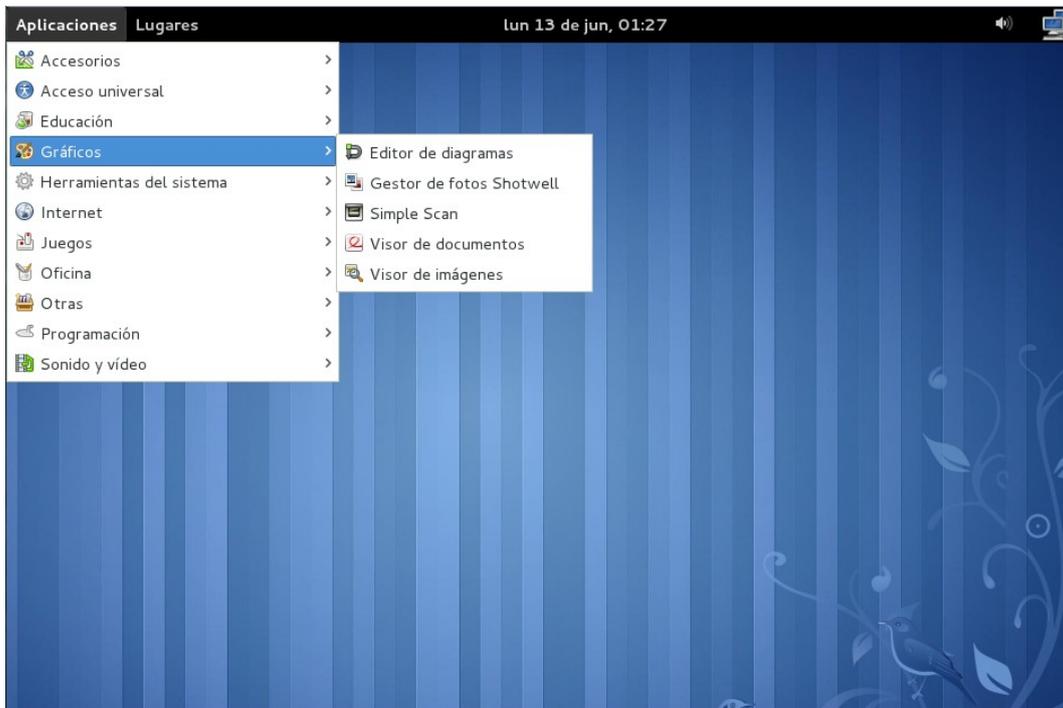


Figura 5.- Acceso Dia (Editor de diagramas)

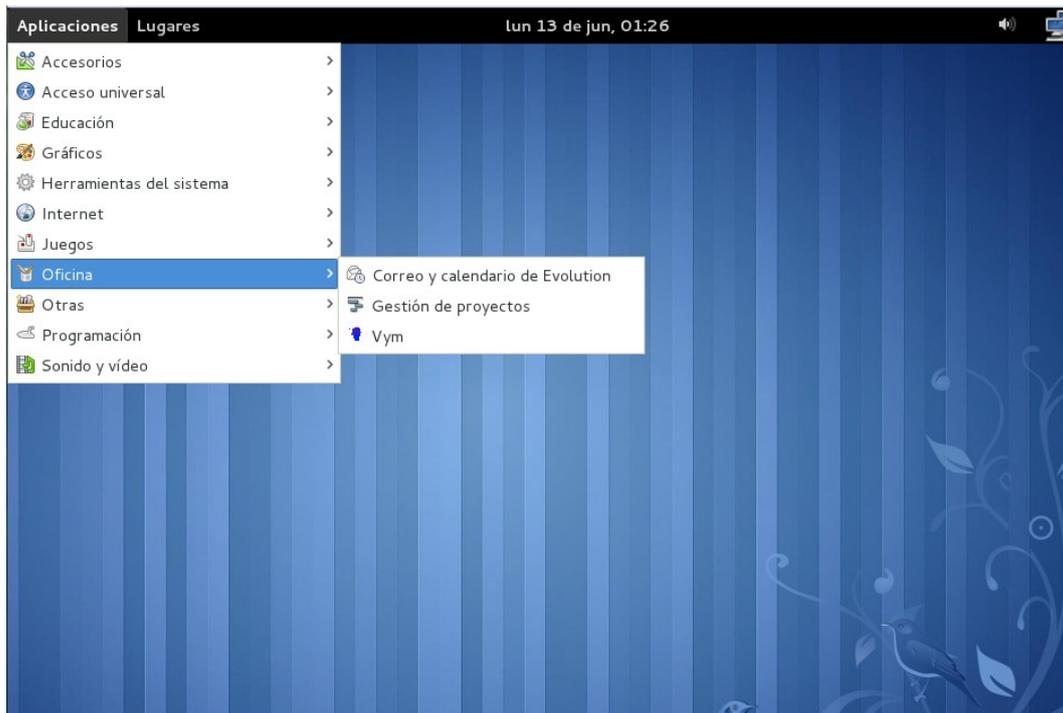


Figura 6.- Acceso Planner (Gestor de proyectos)

También se encuentran otras aplicaciones preinstaladas que pueden ser de utilidad para el técnico en electrónica como son Dia (<http://projects.gnome.org/dia/>), para la representación de diagramas de flujo, o Planner (<http://live.gnome.org/Planner>), para la realización de diagramas Gantt para la gestión de proyectos mediante tareas o actividades.

Llegado a este punto, ya se puede intuir la facilidad que puede presentar en la etapa de formación, como es el acceso a un conjunto de aplicaciones relacionadas con el campo de la electrónica utilizando software libre, lógicamente a usuarios con conocimientos básicos sobre informática (primeras etapas de la formación) el proceso de instalación o manejo de Linux puede presentar cierta incertidumbre y el uso de una distribución como la mostrada en el presente texto puede despejar las posibles dudas.

2.2.- Ejemplos de aplicación

A continuación se presentan algunos ejemplos de utilización de las aplicaciones que pueden encontrar en la distribución LEF.

2.2.1.- Placas de circuito impreso: Kicad

Se trata de un software de código libre (GPL) cuya principal tarea es la de crear circuitos impresos a partir de un esquema electrónico, mediante el uso de las diferentes herramientas disponibles. Además de disponer de amplia documentación en varios idiomas, que puede ser vista y modificada utilizando la aplicación Writer (OpenOffice).

Se considera como un paquete de software o herramienta EDA, ya que integra una herramienta CAD (Eeschema para la edición de esquemas) y una herramienta CAM (PcbNew para la edición de placas de circuito impreso); además de otras herramientas que ayudan en la creación de diseños electrónicos (Gerbview para la visualización de archivos

GERBER, Cypcb para la selección de las huellas -footprint- de los componentes electrónicos desde el esquema, y Kicad como el gestor de proyectos que une todas las herramientas). Destaca la opción de visualización de la placa de circuito impreso en tres dimensiones que proporcionan un diseño final muy profesional, para ser incluido en la posterior documentación y que puede ser de utilidad a un ingeniero mecánico que tenga que acoplar nuestra electrónica a un equipo o carcasa.

Esta aplicación puede ser un sustituto a distribuciones comerciales como el paquete electrónico Orcad, con la salvedad de que no dispone de la opción de simulación. Aunque presenta la posibilidad de que otras aplicaciones generen el código netlist tras la simulación y a posteriori realizar la placa de circuito impreso con Kicad.

En la figura 7 se muestra un pequeño diseño de un amplificador de audio utilizando el circuito integrado LM386, utilizando KICAD, desde el esquema hasta la representación en 3D, pasando por el diseño del circuito impreso.

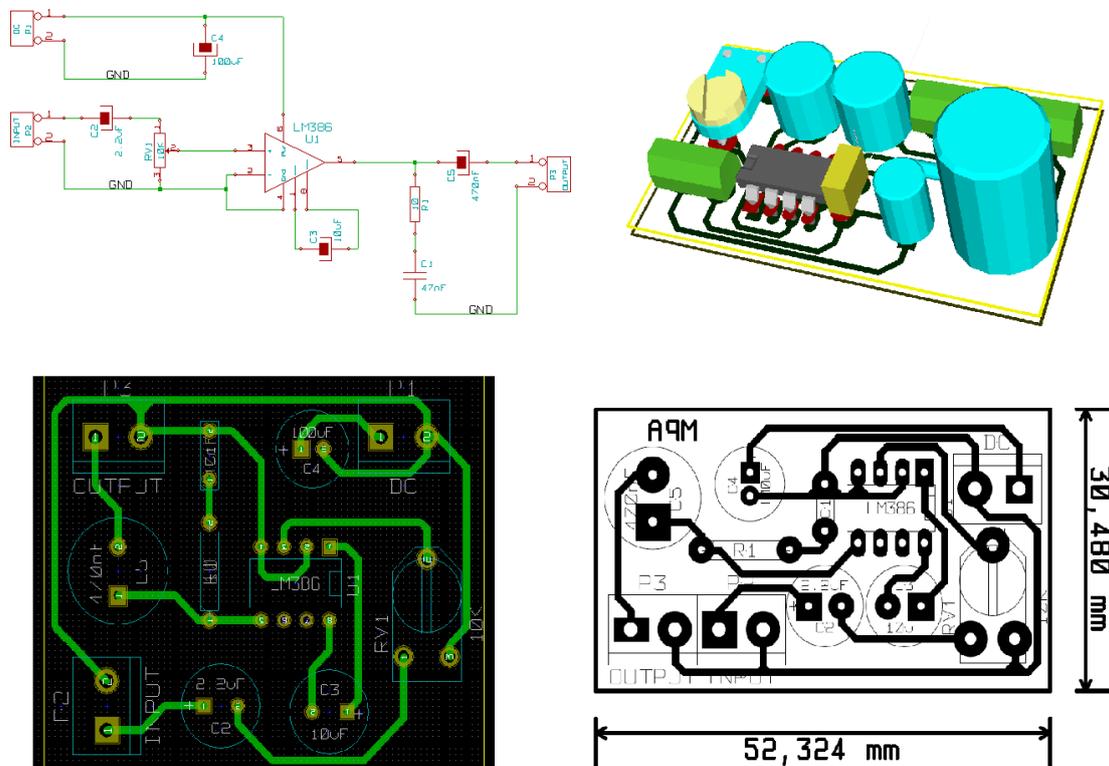


Figura 7.- Ejemplo PCB con Kicad

2.2.2.- Microcontroladores PIC: PIKLAB

Piklab es un entorno integrado de desarrollo para diseños basados en microcontroladores PIC y dsPIC. Con Piklab se puede realizar todo el proceso de creación del diseño con microcontrolador: edición del texto, en varios lenguajes: ensamblador, C o Basic. Pero antes de utilizar uno u otro hay que instalar las librerías necesarias para la compilación y el ensamblado (por ejemplo para la programación en ensamblador necesita tener instalado gputils).

Piklab es la versión libre de MPLAB para Linux, que se trata de un software gratuito ofrecido por la empresa Microchip (www.microchip.com), aunque se puede recurrir a Wine (software para ejecutar aplicaciones Windows, <http://www.winehq.org/>) para utilizar MPLAB en Linux.

En la figura 8 se muestra una muestra de utilización de un circuito simple de alarma utilizada en la formación de microcontroladores PIC.

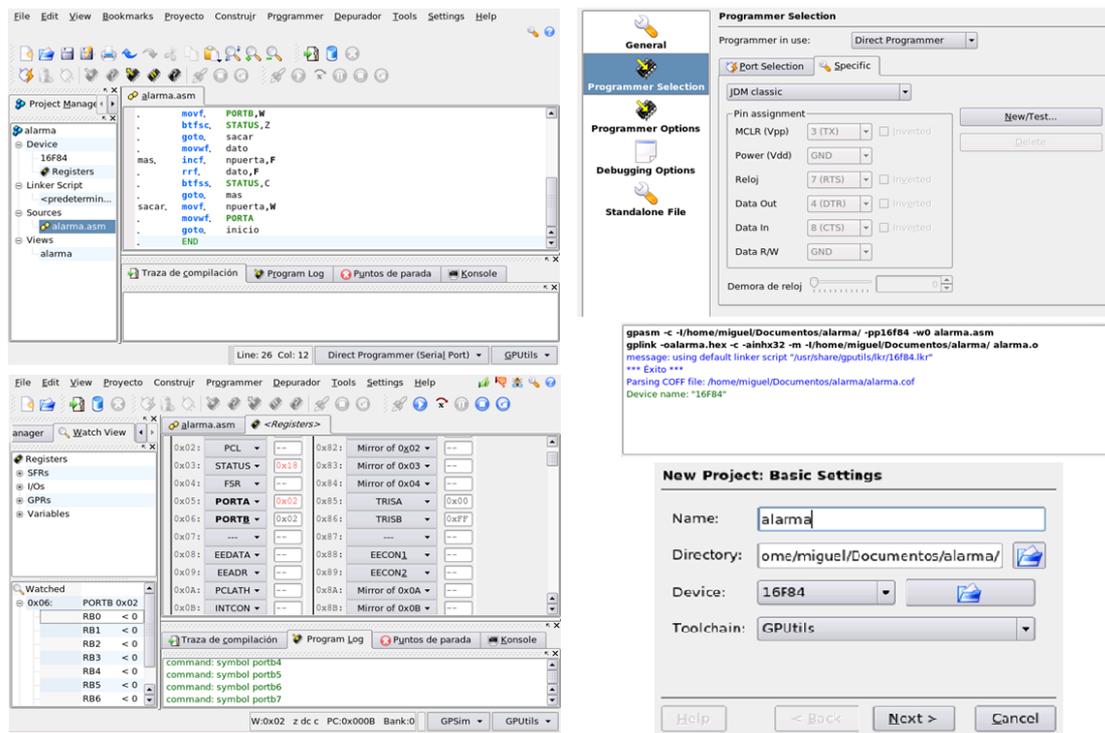


Figura 8.- Ejemplo diseño PIKLAB

2.2.3.- Simulación: QUCS

Qucs (Quite Universal Circuit Simulator) es un simulador integrado de circuitos (según su página web, todavía en desarrollo pero totalmente funcional), utiliza una interfaz gráfica (GUI) para introducir los componentes que forman el circuito y realiza la simulación, pudiendo ser presentado en una sola página. Permite la simulación: en continua (DC), alterna (AC), análisis paramétrico, análisis de balance de armónicos, análisis de ruido, etc.

Se trata de una aplicación multi-plataforma; que dispone de versiones compatibles entre distintos sistemas operativos. Desde su página web se puede acceder a gran cantidad de información (<http://qucs.sourceforge.net/docs.html>).

En comparación con otras aplicaciones, ésta presenta una interfaz sencilla y simple de manejar y puede sustituir a Orcad para simulaciones, sin la opción de creación de placas de circuito impreso.

En la figura 9 se muestra un ejemplo de diseño de un convertidor Buck, en dónde se muestran los componentes y se modela el circuito de control mediante un generador de señal cuadrada para modificar el ciclo de trabajo mediante la ecuación y poder ver los

efectos de la tensión de salida respecto a la tensión de entrada. Se puede ver como en la misma página se muestran las gráficas de simulación de la señal de salida y control, así como el circuito y los parámetros para la simulación (en este caso se trata de un análisis en el tiempo).

Buck converter: convertidor dc/dc reductor

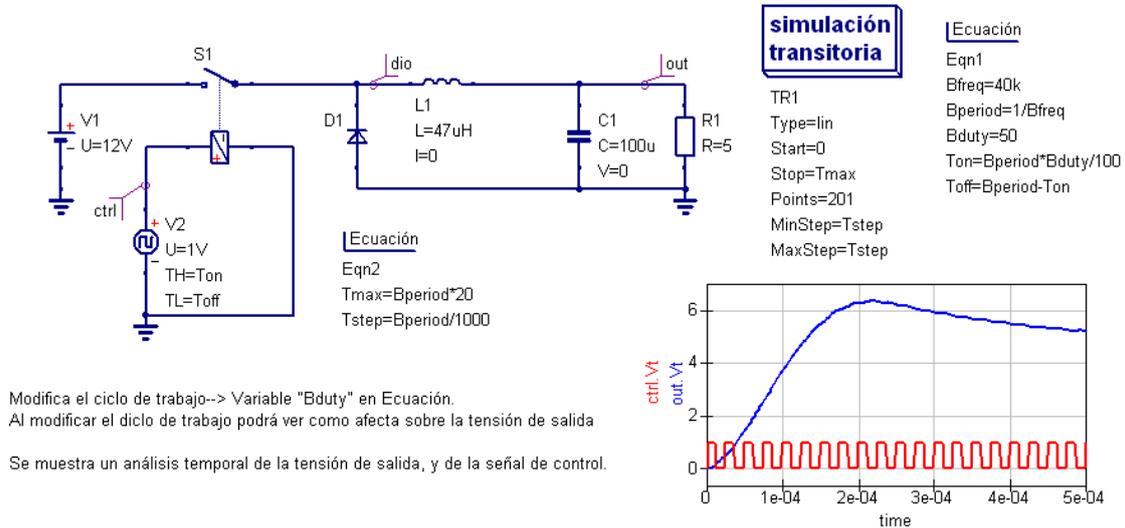


Figura 9.- Ejemplo QUCS

2.2.4.- Diagrama de flujo: DIA

Dia está basado en el software de Microsoft Office Visio, permite dibujar diferentes tipos de diagramas, dispone de librerías con símbolos para la realización de: diagramas, organigramas, representación de redes, entre otros. Con ficheros de extensión: *.eps, *.svg, *.xfig, *.wmf y *.png, con la posibilidad de imprimir o distribuir en varias páginas. En la figura 10 se muestra un diagrama de flujo que representa el cambio de hora mediante dos pulsadores.

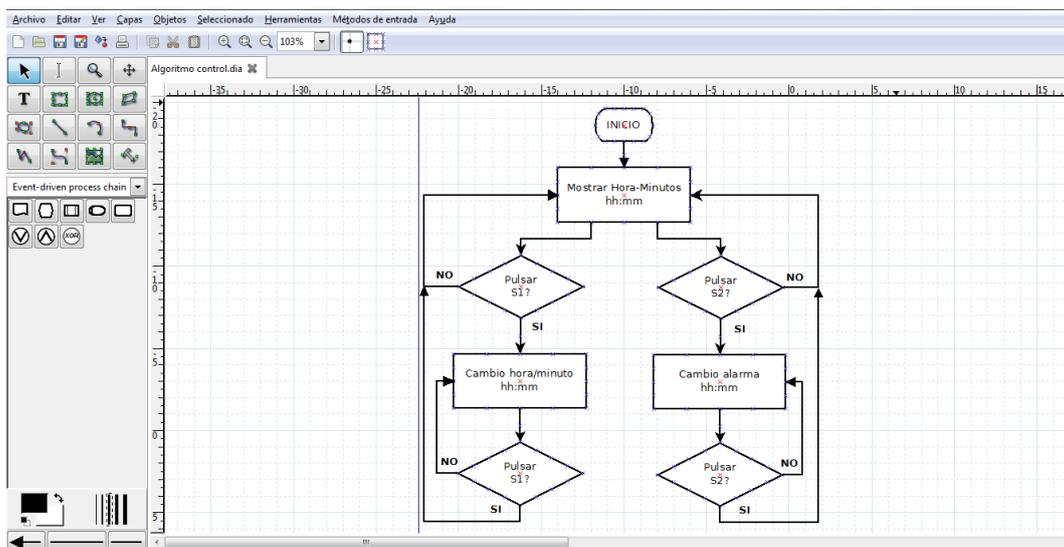


Figura 10.- Flujograma con Dia

3.- Arranque desde USB

Una alternativa a la utilización de un cd-live, es instalar la distribución en una memoria USB y que permite la ejecución al arranque del equipo informático. Tiene la ventaja respecto a la utilización de un cd-rom es que se puede reutilizar e incluso reservar una parte de la memoria para almacenar datos y guardar los cambios realizados.

Para instalar el sistema operativo tiene que disponer del paquete livecd-tools (<http://www.ohloh.net/p/livecd-tools>), y tiene que ejecutar el script livecd-iso-to-disk con la siguiente estructura:

```
/usr/bin/livecd-iso-to-disk /path/to/live.iso /dev/sdb1
```

En donde se debe sustituir /dev/sdb1 por la partición donde quiere guardar la imagen, es decir la memoria USB. Antes debes montar la unidad flash (Pendrive). Comentar que no se trata de un proceso destructivo puesto que cualquier dato que tenga en la memoria USB permanecerá, es decir se instala en el espacio libre.

También puede recurrir a la aplicación LiveUSB Creator (<https://fedorahosted.org/liveusb-creator/>), que permite instalar el sistema operativo a una unidad flash USB. Permite reservar un espacio en la unidad flash, permitiendo guardar archivos y modificaciones en el sistema operativo Live. Incluso no será necesario la descarga previa del sistema operativo, el cual, tras seleccionarlo se descargará automáticamente.

Una vez descargada la aplicación, se ejecuta completando:

- Use existing Live CD: se incluye la dirección en donde se encuentra la imagen.
- Download Fedora: también se puede elegir el descargar una distribución de Fedora, en este caso se descargará antes de ser instalado.
- Target Device: se selecciona el dispositivo en donde se instalará el sistema operativo. Por ejemplo la memoria USB, antes el sistema operativo debe haberla reconocido.
- Persistent Overlay: se puede reservar un espacio de memoria para datos (por ejemplo: documentos, transparencia, ficheros para las aplicaciones).

A continuación hay que hacer clic en Create Live USB, y se realiza el proceso automáticamente. Una vez finaliza el proceso no muestra ninguna ventana de aviso, simplemente verá que la barra de proceso indica el 100% y el botón de Create Live USB vuelve a estar activo (figura 11).



Figura 11.- Entorno de Creative Live USB

Respecto a la memoria USB, el único requisito que indican es que tenga un tamaño de 1Gb o superior. Contra más memoria se disponga, mayor cantidad de memoria se podrá reservar para los datos.

Finalmente para ser ejecutado se debe modificar la secuencia de arranque desde el Setup de la Bios, seleccionando la opción USB hard drive. En algunos equipos se puede acceder a un menú de arranque pulsando F12 al poner en marcha el equipo, debe estar activa la opción F12 Boot Menu en el Setup de la Bios.

En el caso de no poder realizar el arranque directamente sobre la memoria USB, se puede recurrir a una distribución de linux que permite el arranque desde el USB utilizando un cd-rom. Se puede recurrir a la imagen SLAX Boot CD v5.1.8 (<http://www.slax.org/download.php>).

Otra opción es recurrir a un gestor de arranque, un ejemplo puede ser el uso de GAG 4.9 (<http://gag.sourceforge.net/es-index.html>), que permite ser instalado en el equipo o en un disquete. Lógicamente si se trata de un equipo público (por ejemplo el disponible en un aula) no se permitirá instalar el gestor de arranque en el equipo a menos que tengamos acceso a él habitualmente, no sería el caso de tratarse de una charla o conferencia.

4.- Conclusión

LEF permite la utilización directa de aplicaciones sin necesidad de interesarse o complicarse con la instalación o utilización de Linux, pasando a un marco secundario y centrándose en la utilización de las aplicaciones específicas, ya sea: en la formación, trabajos académicos, proyectos para asignaturas, proyectos o tesis finales, o finalmente en el desarrollo profesional.

Otra ventaja es el coste. Por ejemplo, si se está realizando un curso sobre el diseño de circuitos impresos, llega un momento en que se debe explicar el proceso mediante el uso de aplicaciones informáticas, posiblemente propietarias, muy caras o con versiones recortadas; así pues, si se explica con la aplicación Kicad se tiene la ventaja de que si se utiliza software libre y gratuito. Además el alumno pueda acceder a sus propios diseños en su domicilio

teniendo la misma distribución que la utilizada en el curso. Con ello se evitan algunos problemas, tales como: incompatibilidades entre versiones de las aplicaciones, el trabajar con el mismo procesador de texto en la elaboración de notas, misma versión utilizada en clase y en el domicilio (facilita el aprendizaje, puesto que los menús son los mismos), etc.

Del mismo modo la utilización de una distribución de Linux con todo el software necesario instalado, puede ser un primer paso para que personas no usuarias de Linux, pero que trabajan o estudian en la campo de la electrónica, puedan comenzar a iniciar sus andaduras en el mundo del software libre, puesto que se puede comenzar a trabajar sin la necesidad de ser instalado de forma definitiva en el equipo informático. Y esta idea no parece tan descabellada por la fácil disponibilidad (además de precios muy razonables) de dispositivos portátiles como son las memorias USB. La portabilidad es elevada porque no ocupan un espacio mayor que un llavero, en el cual se dispone de las aplicaciones informáticas y herramientas ofimáticas necesarias para el seguimiento de un diseño o proyecto electrónico. También puede ser de utilidad en momentos en que se está trabajando con un proyecto o trabajo con determinada aplicación informática (por ejemplo un diseño con microprocesador utilizando: Pkclab, gpsim, gputils y ktechlab), con lo que se evita tener que instalar las aplicaciones necesarias en cada uno de los equipos que se utilicen.

Otra ventaja puede radicar en la formación a distancia u online (también en las prácticas de la formación presencial) el requisito de utilización o requisitos de software propietario o de pago, evitaría el recurrir a versiones de demostración (con sus limitaciones) y o shareware (con limitaciones temporales), pueden repercutir en un coste menor del curso y sin la necesidad de que los alumnos se tengan que desplazar al centro de formación para utilizar dichas aplicaciones.

Un inconveniente respecto a la utilización de la distribución LEF, son las limitaciones respecto a la documentación disponible. Además en los centros de formación se suele tender a la utilización de software principalmente de pago, debido a que se dispone de mayor información accesible (son muy utilizados), de forma sencilla se puede acceder a ejemplos rápidamente y a documentación adicional (utilizados en otros centros y disponen de mucha bibliografía). En consecuencia, por parte de los actuales ingenieros el estar formados en este tipo de aplicaciones puede repercutir que posteriormente cuando accedan a la docencia (el grupo reducido de ingenieros que después formarán a los futuros ingenieros), utilicen dichas aplicaciones. Otra desventaja a tener en cuenta es que en el mundo laboral no se suelen utilizar dichas aplicaciones, puesto que dependiendo del tipo de diseño se requiere cierto nivel de seguridad o soporte. Por ejemplo un software que está en continuo desarrollo puede tener limitaciones que otras opciones de pago no tienen y pueden complicar el proceso de diseño, en consecuencia hay un aumento de tiempo que repercute a un aumento del coste.

Por último también merece la pena comentar que el uso de herramientas de software libre requiere un cambio de mentalidad, ya que se tiende a pensar que dichas herramientas son solo para informáticos que saben mucho de programación, o que para utilizar software libre hay que ser informáticos. Lógicamente a la hora de colaborar en revisión o actualización o corrección de fallos, hay que tener nociones de programación. Pero del mismo modo la

difusión o la elaboración de documentación sobre su utilización, como manuales o guías, también es colaborar en su difusión. Puesto que no hay que olvidar que cuando se adquiere un software éste va acompañado de un manual de instalación, configuración y utilización. En consecuencia se ha presentado una distribución que permite desde el primer momento a empezar a trabajar bajo Linux, con unas aplicaciones específicas del diseño electrónico, con unos conocimientos básicos de informática. Después la utilización de una aplicación requerirá de la consulta de documentación externa para su manejo, pero esto es lo mismo que con cualquier otra aplicación sea de pago, libre o gratuita.

Animar a partir de estas líneas a probar LEF y que conozca sus posibilidades, en cualquier campo relacionado con el diseño electrónico, tanto a estudiantes como docentes, puesto que utilizando la distribución LEF se puede empezar a utilizar unas aplicaciones sin entrar en conceptos de instalación del sistema operativo.

5.- Referencias

- [1] Breit K. y Judith H. “Manual Dia”. <http://projects.gnome.org/dia/>, 2000.
- [2] Bandel D. y Napier R. “Edición especial Linux 6ª edición” Prentice Hall, 2003.
- [3] Goorah C. “Fedora Electronic Lab in Research & Development environment” LinuxTag, 2009.
- [4] Martinez J.D. “Software libre para simulación electrónica”. Boletín electrónico de la Rama estudiantes de la UNED, edición XII. IEEE UNED Student Branch, 2009.
- [5] Pareja M. “Diseño electrónico con aplicaciones libres o gratuitas para el técnico en electrónica”, ECU 2008.
- [6] Pareja M. “Diseño y desarrollo de circuitos impresos con Kicad”, Rc Libros, 2010.
- [7] Varios autores “Workbook of Qucs”. <http://qucs.sourceforge.net/docs.html>, 2007.