

Hardware Libre

por Raul Espinola (mas que nada fue una recopilación de información, la conclusion es mia, espero les sirva y les guste)

Todos nosotros escuchamos alguna vez hablar sobre el Software Libre, Linux y sistemas libre, pero escuchaste alguna vez sobre Hardware Libre, suena medio raro verdad. Como sabemos el software libre no se refiere especificamente a que el software sea gratuito sino que cumple con ciertas libertades como usar el software, ver el codigo, mejorar el mismo y distribuirlo sin discriminacion.

Temas:

a- Introducción

b- Concepto de Hardware Libre según Antonio Delgado

c- HA (Hardware Abierto) Según www.microbotica.es

d- Historia

e- Definición y Clasificación

1-Hardware estático

1-a Desventajas y Problemas del hardware abierto

1-b Planos en electrónica

1-c Propuesta de definición

2 Hardware reconfigurable

2-a Introduccion

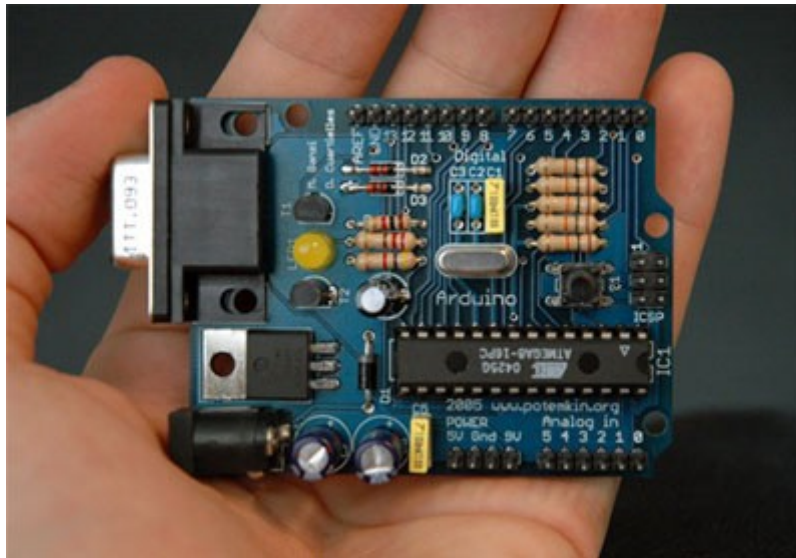
2-b Lenguaje de descripción de hardware

2-c FPGAs

2-d Hardware reconfigurable libre

2-e Un ejemplo de hardware abierto: La tarjeta JPS

f- Conclusion



Introduccion

En la [Informática](#), se denomina [hardware](#) o soporte físico al conjunto de elementos materiales que componen un [computador](#). [Hardware](#) también son los componentes físicos de una [computadora](#) tales como el [disco duro](#), dispositivo de [CD-ROM](#), [disquetera](#), etc. El [hardware](#) se refiere a lo que es tangible, es decir todos los componentes físicos (que se pueden tocar) de la computadora.

El software libre y código abierto ofrece al usuario libertades. Existen licencias que las garantizan y que dan una cobertura legal, como por ejemplo la [GNU GPL](#). El hardware abierto o libre toma estas mismas ideas del software libre y código abierto para aplicarlas en su campo.

Concepto de Hardware Libre según Antonio Delgado

A la sombra del crecimiento del software libre, ha aparecido en los últimos años el llamado hardware libre. Su objetivo es crear diseños de aparatos informáticos de forma abierta, de manera que todas las personas puedan acceder, como mínimo, a los planos de construcción de los dispositivos. Lejos de ser una novedad, esta corriente enlaza directamente con década de los años 70, cuando los primeros aficionados a los ordenadores construían sus propios equipos en los garajes con piezas compradas a diferentes fabricantes y creaban sus propias implementaciones.

Por Antonio Delgado en Pagina Digital

<http://www.paginadigital.com.ar/articulos/2007/2007prim/tecnologia41/hardware-mi-211107.asp>

HA (Hardware Abierto) Según www.microbotica.es

El asentamiento de los Lenguajes de Descripción Hardware (HDL) como VHDL (Very high speed integrated circuit Hardware Description Language) o Verilog HDL, ha impulsado la idea de realizar diseños abiertos de la misma manera que el movimiento Open Source. La reutilización de diseños (cores) o la adaptación de los ya existentes está haciendo mucho más fácil la vida de los ingenieros de hardware, disminuyendo considerablemente los tiempos y costes de diseño.

Con el tiempo y tímidamente va surgiendo una pequeña comunidad que trabaja en conjunto y que intercambia diseños en HDL que van desde procesadores RISC de 32 bits hasta sistemas

bastante más sencillos.

El problema que observamos es que se asociaba el término HA al diseño de soluciones hardware on-chip. Es decir, desde bitstream de configuración para ejecutarse en Lógica Programable (FPGAs) o directamente a nivel ASIC.

Ahora, el proyecto HA es más ambicioso ya que busca agrupar todas las técnicas de diseño y desarrollo hardware. Desde las configuraciones de FPGAs o las máscaras de un ASIC, hasta los esquemas de circuitos o los rutados de PCBs. Es importante destacar que el HA abarca desde la concepción de la idea hasta el punto previo a la fabricación.

Hasta este punto, es natural que todo el trabajo sea realmente un conjunto de bits: esquemáticos con alguna herramienta CAD, código HDL, documentos en procesadores de texto, rutados de pistas, etc... y este conjunto de bits en nada difiere con lo que entendemos por software por lo que es posible adherir la licencia GPL (ya reconocida y ampliamente aceptada) al HA.

Historia

En los años 70s muchos de los futuros CEOs de las grandes empresas del Hardware y Software de la actualidad se encontraban en garages para compartir todos sus diseños electronicos de aparatos informáticos de “forma abierta”, es decir que todos los presentes podian acceder, a los planos, esquemas, PCBs de construcción de los dispositivos.

Luego recién en los años 1990 surge en el sitio Web de Reinoud Lamberts, Open Design Circuits, que fue el primero en proponer la creación de una comunidad del diseño del hardware con el espíritu del [software libre](#). En teoría, [FPGAs](#) permitiría el intercambio de diseños libres electrónicamente, tal como como los programas pueden ser intercambiados.

Pero en la práctica, la comunidad que creció alrededor del sitio nunca agrego diseños libres por la carencia de software libre para el diseño electrónico (que entonces no existía) para lo cual se debatió sobre el uso del [software libre](#) o el [freeware](#) comercial, hasta en el extremo que no se creó ningún diseño real en el sitio en sí mismo.

Pero las discusiones involucraron a una gran cantidad de personas, muchas que estaban implicadas en otras empresas libres del diseño del hardware y esta fue la primera vez que mucha gente había discutido seriamente ¿que era práctico? y ¿que no era práctico? con respecto al hardware. Más bien con la creación de este sitio web, Open Design Circuits sentaron la base para una comunidad entera.

Definición y Clasificación

Compartir diseño hardware es más complicado. No hay una definición exacta. Incluso el propio [Richard Stallman](#) afirma que las ideas del software libre se puede aplicar a los ficheros necesarios para su diseño y especificación (esquemas, [PCB](#), etc), pero no al circuito físico en sí. Por lo cual dependiendo del enfoque. podemos establecer dos clasificaciones. Una, teniendo en cuenta cómo es su naturaleza (estático y reconfigurable) y otra en función a su filosofía.

Clasificación del Hardware

Dada su diferente naturaleza, al hablar de hardware abierto o libre hay que especificar de qué tipo de hardware se está hablando. A continuación se describen cada uno de los diferentes hardware según su naturaleza:

Hardware estético. Entendemos por hardware estético el conjunto de

elementos materiales de un sistema electrónico no reconfigurable. Este

tipo de hardware se caracteriza por ser físicamente único, es decir, o poseemos el circuito o no. Es el más usual hoy en día se caracteriza por estar limitado por su propia existencia física. No podemos copiarlo con facilidad ni distribuirlo "abierto". Sin embargo, lo que sí se puede

hacer es distribuir sus planos de fabricación, o bien a nivel de circuito impreso (PCB) o bien a nivel de fichero de fabricación (GERBER).

Cualquiera de estas dos informaciones es suficiente para permitir a terceros la fabricación del circuito en cuestión, así como su modificación y posible mejora. Una clasificación más detallada puede encontrarse en

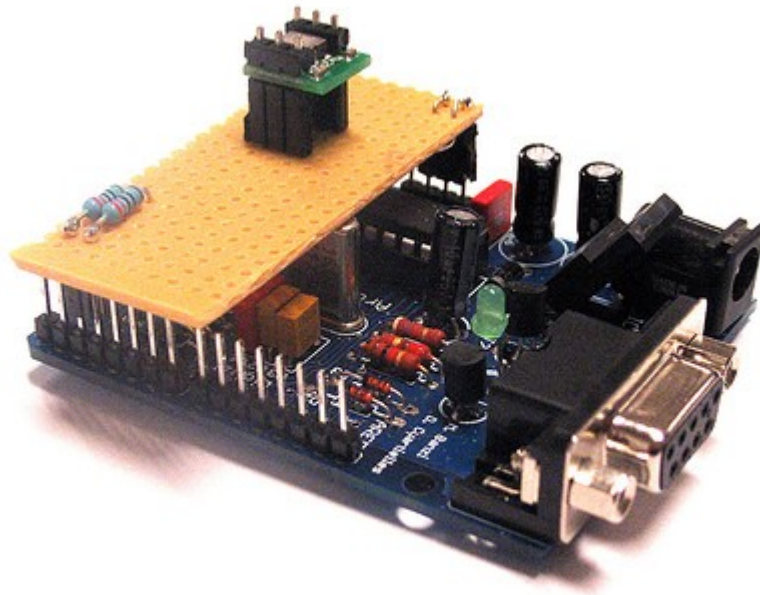
- **Hardware reconfigurable** Este es aquel circuito que se implementa en base a una descripción lógica (HDL) y un sistema base reconfigurable (FPGA). Nuevamente en este caso la descripción (el programa HDL) se puede divulgar con facilidad, mientras que la base no solo no puede intercambiarse (por ser la implementación estática del sistema reconfigurable) sino que no existe en la actualidad ninguna implementación cuyo diseño sea libre.

Así pues, la primera conclusión a la que se puede llegar es que en realidad el hardware libre, por sus propias características de objeto físico, no existe.

Ahora bien, lo que sí pueden existir, tanto en el caso de las implementaciones estáticas como en el de las dinámicas, son diseños de hardware libres (Open Source Hardware). A partir de aquí hardware libre se usará para designar aquellos circuitos, tanto estáticos como dinámicos, cuyo diseño ofrece las mismas libertades que el software libre. En resumen, diremos que un hardware es totalmente libre si:

- El interfaz del hardware ha sido hecho público explícitamente, de forma que dicho hardware puede usarse libremente.
- El diseño del hardware ha sido publicado de forma que otros pueden implementarlo y aprender de él.
- Las herramientas usadas para crearlo son libres, de forma que otros pueden desarrollar y mejorar el diseño.

Cabe Aclarar que existen otra clasificación denominada Filosófica que no entrare en detalles en este momento pero se la puede encontrar en la wikipedia



1-Hardware estático

1-a Desventajas y Problemas del hardware abierto

No se pueden aplicar directamente las cuatro libertades del software libre al hardware, dada su diferente naturaleza. Uno tiene existencia física, el otro no. Aparecen una serie de problemas:

1-Un diseño físico es único. Si yo construyo una placa, es única. Para que otra persona la pueda usar, bien le dejo la mía o bien se tiene que construir una igual. La compartición tal cual la conocemos en el mundo del software NO ES POSIBLE.

2-El Compartir tiene asociado un coste. La persona que quiera utilizar el hardware que yo he diseñado, primero lo tiene que fabricar, para lo cual tendrá que comprobar los componentes necesarios, construir el diseño y verificar que se ha hecho correctamente. Todo esto tiene un coste.

3-Disponibilidad de los componentes. ¿Están disponibles los chips?. Al intentar fabricar un diseño nos podemos encontrar con el problema de la falta de material. En un país puede no haber problema, pero en otro puede que no se encuentran.

Una primera propuesta para definir el hardware libre es la siguiente:

El hardware libre ofrece las mismas cuatro libertades que el software libre, pero aplicadas a los planos del hardware.

Si en el software hablamos de fuentes, aquí hablamos de planos. A partir de ellos podemos fabricar el hardware. El proceso de construcción tiene asociado un coste, que no existe en el caso del software. Sin embargo los planos están disponibles para que cualquiera los pueda usar, modificar y distribuir.

1-b Planos en electrónica

Existen tres tipos de planos, o de ficheros, que describen nuestro diseño:

1-Esquemático: Indica los componentes lógicos y las señales que se conectan entre ellos, pero no nos dice nada de cómo es físicamente la placa.

2-Circuito Impreso (PCB, Printed Circuit Board). Indica el lugar físico en el que situar los componentes, sus dimensiones, encapsulados y qué caminos siguen las pistas para unir sus pines. Nos describe con detalle cómo es físicamente la placa y las dimensiones que tiene.

3-Fichero de fabricación (GERBER). Contiene toda la información necesaria para que se puedan fabricar los PCBs en la industria. Este es un fichero para las máquinas. (Sería algo similar a lo que es el formato Postscript para las impresoras).

Para diseñar la placa usamos los esquemáticos y el PCB. El fichero GERBER es el que obtenemos como resultado y nos permite realizar una fabricación industrial de nuestro PCB. Este fichero puede no haberlo generado el diseñador. A veces es el propio fabricante el que lo genera, a partir de la información que hay en el PCB.

1-c Propuesta de definición

Teniendo en cuenta los tres planos que necesitamos en electrónica, podemos definir el hardware abierto de la siguiente manera:

Un diseño se considera hardware abierto si ofrece las 4 libertades del software libre en el esquemático, PCB y fichero para la fabricación (este último puede no estar disponible).

El fichero de fabricación puede no estar disponible si se trata un prototipo, del que no se ha hecho una tirada industrial. O también puede ocurrir que se haya fabricado industrialmente, pero a partir del PCB. En este caso el autor dispone de unos fotolitos impresos, pero no de un fichero electrónico.

2 Hardware reconfigurable

2-a Introduccion

El hardware reconfigurable es aquél que viene descrito mediante un lenguaje HDL (Hardware Description Language). Su naturaleza es completamente diferente a la del hardware estático. Se desarrolla de una manera muy similar a como se hace con el software. Ahora nuestros diseños son ficheros de texto, que contienen el "código fuente". Se les puede aplicar directamente una licencia libre, como la GPL.

2-b Lenguaje de descripción de hardware

Un **lenguaje de descripción de hardware** (HDL, *Hardware Description Language*) permite documentar las interconexiones y el comportamiento de un circuito electrónico, sin utilizar diagramas esquemáticos.

El flujo de diseño suele ser típico:

- a- Definir la tarea o tareas que tiene que hacer el circuito.
- b- Escribir el programa usando un lenguaje **HDL**. También existen programas de captura de esquemas que pueden hacer esto, pero no son útiles para diseños complicados.
- c- Comprobación de la sintaxis y simulación del programa.
- d- Programación del dispositivo y comprobación del funcionamiento.

Un rasgo común a estos lenguajes suele ser la independencia del hardware y la modularidad o jerarquía, es decir, una vez hecho un diseño éste puede ser usado dentro de otro diseño más complicado y con otro dispositivo compatible.

Uno de los lenguajes utilizados es VHDL y otro que GNU es GHDL

Wiki

http://es.wikipedia.org/wiki/Lenguaje_de_descripción_de_hardware

<http://es.wikipedia.org/wiki/VHDL>

Guia de GHDL

<http://ghdl.free.fr/ghdl/index.html>

2-c FPGAs

Las FPGAs son dispositivos que nos permiten implementar circuitos digitales. Están compuestos por bloques iguales, configurables, llamados CLBs, que se unen dinámicamente según cómo se especifique en la memoria de configuración. De esta forma, cambiando el contenido de la esta memoria, se establecen unas uniones diferentes entre los CLBs, obteniéndose un dispositivo u otro.

El fichero que contiene la configuración se denomina bitstream. La características fundamental de las FPGAs es que son Dispositivos universales. Se pueden ``convertir" en cualquier diseño digital, según el bitstream que se cargue en su memoria de configuración.

<http://es.wikipedia.org/wiki/FPGA>

2-d Hardware reconfigurable libre

El hardware reconfigurable se puede compartir exactamente igual que el software. Las características que tiene son:

- Se pueden ofrecer las 4 mismas libertades del software libre a los ficheros en HDL. Por ejemplo distribuyéndolos bajo licencia GPL.
- Aparecen comunidades hardware que comparten información, como OpenCores y OpenCollector
- Se pueden crear repositorios hardware, lo que permite que muchas personas puedan participar en el desarrollo.
- Pueden aparecer distribuciones que recopilen todo el hardware libre existente.

De igual forma que para ejecutar el software necesitamos un máquina que tenga un procesador, para probar nuestro hardware necesitamos una plataforma con una FPGA en la que ``descargar" los diseños (un hardware estático). La tarjeta JPS es una de ellas, que además es libre (En el sentido que se le ha dado al hardware estático).

<http://www.opencores.org/>

<http://www.opencollector.org/>

<http://www.learobotics.com/personal/juan/publicaciones/art4/html/index.html>

2-e Un ejemplo de hardware abierto: La tarjeta JPS

En la Escuela Politécnica de la Universidad Autónoma de Madrid han desarrollado la tarjeta JPS

una entrenadora para las FPGAs de la familia 4000 y Spartan I de Xilinx, que se está utilizando en el laboratorio de Estructura y Diseño de Circuitos Digitales. Según la clasificación establecida previamente, es del tipo MML. Para su diseño se ha utilizado el programa Eagle, en una máquina corriendo Debian/Sarge.

Al tratarse de hardware libre, están disponibles todos los planos. Algunas de sus ventajas son:

- Cualquiera la puede fabricar. Los alumnos, universidades o empresas que la encuentren interesante la pueden construir.
- Cualquiera la puede modificar. Seguramente no se adaptará a las necesidades de docencia de otras universidades o centro de investigación. Tienen la posibilidad de modificar la placa, en vez de empezar a diseñarla desde cero.
- Cualquier empresa la puede comercializar, y cobrar por ofrecer los servicios de fabricación y verificación del correcto funcionamiento, así como mantenimiento.

Imagen JPS:<http://www.learobotics.com/personal/juan/publicaciones/art4/html/img2.png>

Conclusion

La Idea del Hardware libre no solo es importante, en la aplicación del modelo comunitario y colaborativo para el crecimiento intelectual libre, sobre los sistemas electronicos digitales, sino tambien que debemos luchar contra las patentes de las grandes industrias de dos o tres paises productores de tecnologia (EEUU, Japón, otros), de los cuales proviene el 99% del hardware y de los productos electronicos. Nuestra dependencia como Nación nos hace cada vez mas vulnerables a los deseos de estas naciones, donde al hiperdesarrollo tecnologico, nuestros impulsos consumistas de la ultima tecnologia.y las patentes electronicas igual que el software propietario nos encaden cada vez mas.

En el caso del Hardware seria como pagar licencias de uso por utilizar una RUEDA al creador de esta, o imaginense pagar derechos de autor por crear tus propios muebles o herramientas al que primero las diseño, por como sigue la evolucion de nuestra sociedad donde la integracion tecnologica es cada día mas estrecha, dentro de unos años deberemos pagar licencia, derechos de autor y patentes por el 90% de nuestros artefactos, sistemas electronicos, informaticos e inclusive, tal vez por nuestra vida diaria.

¿Creo yo o es tiempo de Hacer Algo, que me dices TU?

Proyectos de Hardware Libre

De la unión de plataformas de software y hardware abierto, nacen proyectos que suponen una ventaja para usuarios y desarrolladores al disponer estos de mayor libertad para crear e instalar aplicaciones al mismo tiempo que adaptar algunas funciones del hardware.

OpenMoko el Primer Celular Libre

[OpenMoko](#) es una plataforma abierta para desarrolladores para crear un sistema operativo libre basado en Linux que funcione sobre teléfonos móviles. De esta manera, diversos fabricantes de teléfonos móviles podrían incluirlo en los terminales llamados 'smartphone'.

El primer teléfono móvil que funciona con OpenMoko es el [Neo1973](#) fabricado por FIC, cuya versión para desarrolladores está actualmente [a la venta por 300 dólares](#). Se espera que un nuevo dispositivo móvil con mayores funcionalidades sea lanzado próximamente bajo el nombre en clave [HXD8](#).

Android

En este sentido, Google ha presentado recientemente una plataforma de software abierta para desarrolladores de dispositivos móviles llamada [Android](#), junto con la creación de una alianza con 34 empresas del sector denominada 'Open Handset Alliance', para dotar de un sistema operativo abierto a los fabricantes de teléfonos. Esta iniciativa puede suponer que [se pueda ensamblar un teléfono móvil por piezas de diferentes fabricantes](#) y hacerlo funcionar con Android.

<http://www.openhandsetalliance.com>

Arduino

Otra de las iniciativas nacidas bajo el concepto de hardware libre es el [proyecto Arduino](#). Esta plataforma consiste en una placa con diversas entradas y salidas que permite el desarrollo de infinidad de proyectos para el control de objetos interactivos autónomos, al mismo tiempo que sirve de plataforma para estudiantes como iniciación al mundo de la electrónica.

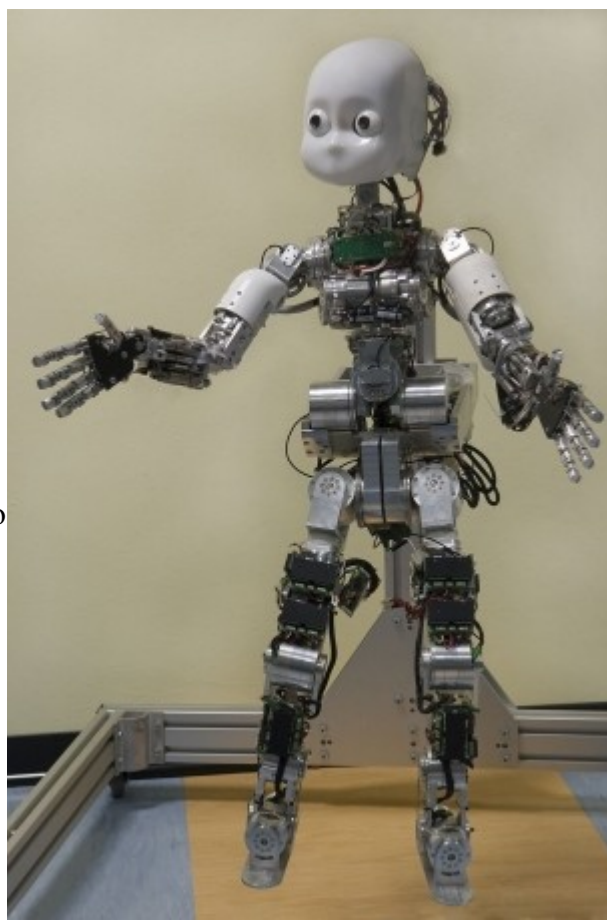
Una de las opciones más utilizadas de la plataforma Arduino es la del diseño y construcción de instrumentos musicales electrónicos en combinación con el software libre de audio [Pure Data](#), que permite crear experiencias sonoras y visuales en tiempo real por parte de artistas digitales.

<http://www.arduino.cc/en/Guide/Introduction>

Robocub:

iCub forma parte del [proyecto RobotCub](#), realizado por un total de 16 grupos de investigación, con financiación de la Unión Europea, que tiene como objetivos la creación de un robot humanoide que sirva de apoyo a otros investigadores, por un lado; y avanzar en el conocimiento que se tiene sobre factores clave de la cognición profundizando en la investigación y el desarrollo de la [plataforma cognitiva de iCub](#), por otro lado.

El software que necesita este robot, tanto las aplicaciones relacionadas como el software embebido y los controladores, se distribuyen bajo la licencia pública general de GNU. El diseño de los componentes mecánicos y electrónicos y una extensa documentación se distribuyen bajo la licencia de documentación libre de GNU.



<http://www.robotcub.org/index.php/robotcub>

Computer systems

- [PC532](#) - a [personal computer](#) design based on the [NS32532](#) microprocessor, released in 1990.
- [ECB AT91](#) - [Single-board computer](#) based on the Atmel AT91RM9200 ARM9 processor (180 MHz).
- [ECB ATmega32/644](#) - [Single-board computer](#) based on the [Atmel](#) ATmega32/644 (20 MHz) with webserver-capability and less than 100mA power consumption
- [Simputer](#) - handheld computer aimed at developing countries

Peripherals

- The [RepRap Project](#): An open source, self-replicating [3D printer](#)

Computer components

CPU

There are several "open source hardware" CPUs, typically implemented as a [soft microprocessor](#).

- [OpenSPARC](#) is an open-source processor project to which [Sun Microsystems](#) have contributed the [UltraSPARC T1](#) and [UltraSPARC T2](#) multicore processor designs.
- [OpenRISC](#) is a group of developers working to produce a very high performance [open source RISC CPU](#).
- [LEON](#) is an open source 32-bit SPARC-like CPU created by the ESA. It's the standard CPU for the European Space Industry.
- [OpenCores](#) is a [foundation](#) that attempts to form a community of designers to support open-source cores (logic designs) for CPUs, peripherals and other devices. OpenCores maintains an open-source on-chip interconnection bus specification called [Wishbone](#).

Several designs include a CPU:

- [Arduino](#), an open-source [physical computing](#) platform based on a simple I/O board and a development environment that implements the open source [Processing](#) / *Wiring* language.
- [Freeduino](#) - open-source electronics prototyping platform derived from Arduino.
- [ASoC](#) (ALSA System on Chip).
- [PLAICE](#) - The PLAICE is an open source hardware and software project developing a powerful in-circuit development tool that combines in one device the features of a FLASH Programmer, Memory Emulator, and High Speed Multi-Channel Logic Analyzer. It runs [uClinux](#).

Graphics cards

- [Open Graphics Project](#) aims to design an open architecture and standard for [graphics cards](#).
- [Project VGA](#) - another free graphics core project, aiming at cheaper hardware

Wireless hardware

- [OpenPattern](#)[\[10\]](#)[\[11\]](#) [\[12\]](#)
- [Sun SPOT](#) is an open source hardware and software platform for [sensor networks](#) and battery powered, wireless, embedded development.

Laptop case

- [VIA OpenBook](#) - a [subnotebook](#) case design released by [VIA Technologies](#)

Audio electronics

- [MIDIbox](#) The MIDIbox project is an open source modular DIY hardware and software platform for MIDI devices like controllers, synthesizers, sequencers
- [Monome 40h](#) - A reconfigurable grid of sixty-four backlit buttons, used via USB. A limited batch of 500 monome 40h has been produced. All the design process, design specifications, firmware and PCB schematics are available online

Other

- [Chumby](#) - "[ambient](#)" information device

Telephony

- [Openmoko](#) - open phone framework (first use case: [FIC Neo FreeRunner](#), Released as of mid 2008)

Machines and tools

- The [Multimachine](#), an open source [machine tool](#) project

Renewable energy

- [Small wind turbines](#): To assist people in the developing countries, and hobbyists alike, several projects have been open-sourced, e.g. the Jua Kali wind turbine, Hugh Piggot's wind turbine, ForceField Wind Turbine, et cetera

Robotics

- [e-puck mobile robot](#), an open-hardware, education oriented, mobile robot.
- [Make Controller Kit](#)

Prosthetics

There is currently an open Prosthetics design forum known as the "[Open Prosthetics Project](#)". The group employs collaborators and volunteers to advance Prosthetics technology while attempting to lower the costs of these necessary devices.

Other projects

- [RONJA](#) - Open source Free Space Optic system, DIY in a garage, 10 Mbit/s full duplex/1.4 km
- [LED Throwies](#) - non-destructive graffiti and light displays.
- [openECGproject](#) - an online community conceived around a simple, but challenging and worthy goal - to develop an open source hardware and software solution for electrocardiography.

Implementaciones de Arduino en Escuelas Tecnicas de Argentina

<http://www.buenosaires.gov.ar/blog/educacion/category/hardware-libre/>

Proyecto microbot Tritt

Iniciatica de MIT para hardware Libre

<http://csg.csail.mit.edu.nyud.net/oshd/index.html>

Grupo de usuarios gEDA

The gEDA project was started because of the lack of [free](#) EDA tools for [POSIX](#) systems with the primary purpose of advancing the state of free hardware or [open source hardware](#). The suite is mainly being developed on the GNU/Linux platform with some development effort going into making sure the tools run on other platforms as well.

<http://www.geda.seul.org/>

Build It. Share It. Profit. Can Open Source Hardware Work?

http://www.wired.com/techbiz/startups/magazine/16-11/ff_openmanufacturing?currentPage=all

Planteos y Dudas:

<http://weblogs.madrimasd.org/softwarelibre/archive/2008/02/11/84211.aspx>

Fuentes:

<http://www.paginadigital.com.ar/articulos/2007/2007prim/tecnologia41/hardware-mi-211107.asp>

<http://www.microbotica.es/web/ha.htm>

<http://www.consumer.es/web/es/tecnologia/hardware/2007/11/20/171514.php>

Muy buen Trabajo de gente de España (Juan Gonzalez 2003-12-31) sobre Hardware Libre les recomiendo su Lectura y varias implementaciones que hicieron. Recomendado.

<http://www.learobotics.com/personal/juan/publicaciones/art4/html/index.html>