

Introducción al Desarrollo de Electrónica Digital en FreeBSD

Sr. Ricardo Albarracín B.
Ingeniero de Desarrollo de Sistemas Digitales
Electrolinux
rab@electrolinux.cl

Serena 25 de Abril del 2009
Flisol2009

Temas que se tratarán

Temas que se tratarán

Apagar los CELULARES por favor

Temas que se tratarán

Apagar los CELULARES por favor

- Porque usar FreeBSD?
- Algunas herramientas de desarrollo electrónico.
- Que seleccionaremos para hacer nuestros desarrollos.
- Compiladores de **lenguaje C** para Microcontroladores [μ C].
- Veremos algún ejemplo de esquemáticos y pcb.
- Áreas de aplicaciones
- Estructura general de un programa en **lenguaje C**.
- Compilacion y carga en un Microcontrolador [μ C].
- Porque es importante en la educación?
- Consultas de los asistentes.

Introducción: ¿porqué usarlo?

- FreeBSD tiene muchas herramientas de desarrollo para Electrónica como cualquier otro ambiente Unix.

Introducción: ¿porqué usarlo?

- FreeBSD tiene muchas herramientas de desarrollo para Electrónica como cualquier otro ambiente Unix.
- Existe una masa crítica de usuarios mucho mayor en FreeBSD, lo que permite disponer de ayuda fácilmente.

Introducción: ¿porqué usarlo?

- FreeBSD tiene muchas herramientas de desarrollo para Electrónica como cualquier otro ambiente Unix.
- Existe una masa crítica de usuarios mucho mayor en FreeBSD, lo que permite disponer de ayuda fácilmente.
- Estas herramientas *siempre* son compiladas en *tu propia máquina*, lo que deja más *rápido* y *estable* al sistema.

Introducción: ¿porqué usarlo?

- FreeBSD tiene muchas herramientas de desarrollo para Electrónica como cualquier otro ambiente Unix.
- Existe una masa crítica de usuarios mucho mayor en FreeBSD, lo que permite disponer de ayuda fácilmente.
- Estas herramientas *siempre* son compiladas en *tu propia máquina*, lo que deja más *rápido* y *estable* al sistema.
- Si bien la compilación requiere de *mayor tiempo*, es el precio que debemos pagar por una *mayor estabilidad del sistema*.

Introducción: ¿porqué usarlo?

- FreeBSD tiene muchas herramientas de desarrollo para Electrónica como cualquier otro ambiente Unix.
- Existe una masa crítica de usuarios mucho mayor en FreeBSD, lo que permite disponer de ayuda fácilmente.
- Estas herramientas *siempre* son compiladas en *tu propia máquina*, lo que deja más *rápido* y *estable* al sistema.
- Si bien la compilación requiere de *mayor tiempo*, es el precio que debemos pagar por una *mayor estabilidad del sistema*.
- FreeBSD es un sistema que esta suficientemente maduro para el **desktop**.

Reseña de las herramientas

- 1 *Herramientas gráficas libres* para diseño de esquemáticos y PCB:

Reseña de las herramientas

- 1 *Herramientas gráficas libres* para diseño de esquemáticos y PCB:
 - 1 **Schematics:** gEDA, Kicad, Oregano (Schematic capture, Circuit simulador) y otras libres.

Reseña de las herramientas

- 1 *Herramientas gráficas libres* para diseño de esquemáticos y PCB:
 - 1 **Schematics:** gEDA, Kicad, Oregano (Schematic capture, Circuit simulador) y otras libres.
 - 2 **PCB:** PCB, Kicad al manos, otras en desarrollo.

Reseña de las herramientas

- 1 *Herramientas gráficas libres* para diseño de esquemáticos y PCB:
 - 1 **Schematics:** gEDA, Kicad, Oregano (Schematic capture, Circuit simulador) y otras libres.
 - 2 **PCB:** PCB, Kicad al manos, otras en desarrollo.
 - 3 **Comerciales:** para SCH/PCB Eagle (la que uso), se debe configurar emulacion Linux por ser un paquete binario para Linux.

Reseña de las herramientas

- 1 *Herramientas gráficas libres* para diseño de esquemáticos y PCB:
 - 1 **Schematics:** gEDA, Kicad, Oregano (Schematic capture, Circuit simulador) y otras libres.
 - 2 **PCB:** PCB, Kicad al manos, otras en desarrollo.
 - 3 **Comerciales:** para SCH/PCB Eagle (la que uso), se debe configurar emulacion Linux por ser un paquete binario para Linux.
- 2 Para los **Compiladores de 'C'** para generar *código embebido* a diferentes arquitecturas de $\mu C/\mu P$, hay también alternativas libres y de una alta calidad.

Reseña de las herramientas

- 1 *Herramientas gráficas libres* para diseño de esquemáticos y PCB:
 - 1 **Schematics:** gEDA, Kicad, Oregano (Schematic capture, Circuit simulador) y otras libres.
 - 2 **PCB:** PCB, Kicad al manos, otras en desarrollo.
 - 3 **Comerciales:** para SCH/PCB Eagle (la que uso), se debe configurar emulacion Linux por ser un paquete binario para Linux.
- 2 Para los **Compiladores de 'C'** para generar *código embebido* a diferentes arquitecturas de $\mu C/\mu P$, hay también alternativas libres y de una alta calidad.
 - 1 Compilador **'SDCC'** para las familias mcs51, gbz80, z80, avr, ds390, pic16, pic14, TININative, xa51, ds400, hc08. Una excelente alternativa libre GPL.

Reseña de las herramientas

- 1 *Herramientas gráficas libres* para diseño de esquemáticos y PCB:
 - 1 **Schematics:** gEDA, Kicad, Oregano (Schematic capture, Circuit simulador) y otras libres.
 - 2 **PCB:** PCB, Kicad al manos, otras en desarrollo.
 - 3 **Comerciales:** para SCH/PCB Eagle (la que uso), se debe configurar emulacion Linux por ser un paquete binario para Linux.
- 2 Para los **Compiladores de 'C'** para generar *código embebido* a diferentes arquitecturas de $\mu C/\mu P$, hay también alternativas libres y de una alta calidad.
 - 1 Compilador '**SDCC**' para las familias mcs51, gbz80, z80, avr, ds390, pic16, pic14, TININative, xa51, ds400, hc08. Una excelente alternativa libre GPL.
 - 2 Compilador '**AVR-GCC**' para muchas arquitecturas dentro de la gamma AVR-RISC, con resultados muy satisfactorios, por su calidad de código y pequeño tamaño del **binario generado**.

Herramientas Que son y como instalarlas

Recomiendo seleccionar **sólo una** y aprender a usarla bien.

Herramientas Que son y como instalarlas

Recomiendo seleccionar **sólo una** y aprender a usarla bien.

- 1 **gEDA**: El proyecto gEDA esta orientado en la creacion de una suite de Herramientas para la Automatización del Diseño Electrónico. Esta herramienta se utiliza para el diseño de circuitos electrónicos y eléctricos, simulación, prototipos y producción.
 - Ports: `/usr/ports/cad/geda`, para instalar `#make install clean`

Herramientas Que son y como instalarlas

Recomiendo seleccionar **sólo una** y aprender a usarla bien.

- 1 **gEDA**: El proyecto gEDA esta orientado en la creacion de una suite de Herramientas para la Automatización del Diseño Electrónico. Esta herramienta se utiliza para el diseño de circuitos electrónicos y eléctricos, simulación, prototipos y producción.
 - Ports: `/usr/ports/cad/geda`, para instalar `#make install clean`
- 2 **Kicad**: Constituye un gestor de proyectos que permite trabajar al mismo tiempo con distintas aplicaciones: Eeschema, Pcbnew, Cvpcb y Gerbview.
 - Ports: `/usr/ports/cad/kicad`, para instalar `#make install clean`

Herramientas Que son y como instalarlas

Recomiendo seleccionar **sólo una** y aprender a usarla bien.

- 1 **gEDA**: El proyecto gEDA esta orientado en la creacion de una suite de Herramientas para la Automatización del Diseño Electrónico. Esta herramienta se utiliza para el diseño de circuitos electrónicos y eléctricos, simulación, prototipos y producción.
 - Ports: **`/usr/ports/cad/geda`**, para instalar *`#make install clean`*
- 2 **Kicad**: Constituye un gestor de proyectos que permite trabajar al mismo tiempo con distintas aplicaciones: Eeschema, Pcbnew, Cvpcb y Gerbview.
 - Ports: **`/usr/ports/cad/kicad`**, para instalar *`#make install clean`*
- 3 **Oregano**: Constituye otra herramienta para generar esquemáticos y simulación de circuitos digitales y análogos.
 - Ports: **`/usr/ports/cad/oregano`**, para instalar *`#make install clean`*

La que uso por historia y compatibilidad (item4)

Muy buena herramienta de desarrollo <http://www.cadsoft.de/>

Eagle tiene la ventaja que es una herramienta de **diseño electrónico de uso profesional**, lleva varios años en el mercado, **no es libre**, pero hay una **versión freeware en los ports**, con las mismas características que la versión profesional pero con limitaciones de tamaño (sólo una hoja en el esquemático y no más de 10x8cm en PCB).

La que uso por historia y compatibilidad (item4)

Muy buena herramienta de desarrollo <http://www.cadsoft.de/>

Eagle tiene la ventaja que es una herramienta de **diseño electrónico de uso profesional**, lleva varios años en el mercado, **no es libre**, pero hay una **versión freeware en los ports**, con las mismas características que la versión profesional pero con limitaciones de tamaño (sólo una hoja en el esquemático y no más de 10x8cm en PCB).

Amplias bibliotecas de componentes electrónicos de uso público.

La que uso por historia y compatibilidad (item4)

Muy buena herramienta de desarrollo <http://www.cadsoft.de/>

Eagle tiene la ventaja que es una herramienta de **diseño electrónico de uso profesional**, lleva varios años en el mercado, **no es libre**, pero hay una **versión freeware en los ports**, con las mismas características que la versión profesional pero con limitaciones de tamaño (sólo una hoja en el esquemático y no más de 10x8cm en PCB).

Amplias bibliotecas de componentes electrónicos de uso público.

Tener claro que heremos

Lo importante es saber bien cuales son las necesidades para el diseño, ya que **Eagle** en su versión libre, satisface prácticamente todas las necesidades para desarrollos pequeños, como hobbies y robótica, proyectos de baja escala, en donde no se requieran mayores tamaños a los indicados.

Ejemplos de uso (item5)

Demostración del uso de Eagle 4.16r2

Se verá un breve uso del programa:

- Veremos un esquemático desarrollado en Eagle 4.16r2
- El PCB generado a contar de su esquemático.
- Resultado final... la fabricación.

Ejemplos de uso (item5)

Demostración del uso de Eagle 4.16r2

Se verá un breve uso del programa:

- Veremos un esquemático desarrollado en Eagle 4.16r2
- El PCB generado a contar de su esquemático.
- Resultado final... la fabricación.

Que tenemos hasta ahora?

- Un sistema Operativo muy confiable en donde desarrollar.
- Muestra de la herramienta de diseño (esquemático/pcb).
- Y hemos fabricado nuestro hardware.

Demostración (item6)

Uso y aplicaciones

- Usos en Robótica, Hobbys, inteligencia artificial y colaborativa, aeromodelismo, Control, Domótica, Instrumentación, etc.
- Desarrollo de equipos comerciales, terminales de captura de datos, electrodomésticos.
- Usos para *desarrollos especiales o a pedido*.

Acá hago la demostración del programa.

El código embebido

Generalidades del Código C para Microcontroladores

Los programas en lenguaje C para los Microcontroladores de 8/16 bits son distintos a los realizados en una máquina con un S.O. ya que este se encarga de muchos problemas de bajo nivel, estos son: la programación de los registros internos, programación de periféricos, acceso a memoria, ya que no hay un administrador de manejo de memoria como en las arquitecturas de 32bits.

Hay que tener en cuenta que los recursos siempre son *limitados* y una buena programación debe velar por ellos, es decir se debe hacer un código eficiente, de tal forma de *sacarle el jugo a los escasos recursos*.

Aún con estas limitaciones, hay resultados muy satisfactorios en muchas áreas, con bajos costos finales y con herramientas de calidad.

El proceso de Compilación y carga (item8)

Compilación

- Veremos el archivo de Makefile para compilar.
- Proceso de Compilación.
- Carga del código binario generado a la *Memoria de Programa del μC* .
- Corremos el programa y hacemos pruebas de comunicaciones con el μC .

Estructura Clásica de un Programa para μC

```
void main( void ){
    init_cpu();
    serial_init();

    serial_str("\r\nReady\r\n$ ");

    while(1){
        pecado_mortal();
        .....
        foo();
    }
}
```

Porque es importante en la educación?

- Hemos visto un completo sistema de desarrollo disponible usando *Software libre*.

Porque es importante en la educación?

- Hemos visto un completo sistema de desarrollo disponible usando *Software libre*.
- *Método* aplicable a muchas escuelas y colegios en donde se imparte/enseña *electrónica e informática*.

Porque es importante en la educación?

- Hemos visto un completo sistema de desarrollo disponible usando *Software libre*.
- *Método* aplicable a muchas escuelas y colegios en donde se imparte/enseña *electrónica e informática*.
- Aplicables a *docencia y a laboratorios de investigación* en Universidades e Institutos profesionales.

Porque es importante en la educación?

- Hemos visto un completo sistema de desarrollo disponible usando *Software libre*.
- *Método* aplicable a muchas escuelas y colegios en donde se imparte/enseña *electrónica e informática*.
- Aplicables a *docencia y a laboratorios de investigación* en Universidades e Institutos profesionales.
- Incentiva el desarrollo tecnológico en la sociedad, ya que se encuentran disponibles para todos.

Porque es importante en la educación?

- Hemos visto un completo sistema de desarrollo disponible usando *Software libre*.
- *Método* aplicable a muchas escuelas y colegios en donde se imparte/enseña *electrónica e informática*.
- Aplicables a *docencia y a laboratorios de investigación* en Universidades e Institutos profesionales.
- Incentiva el desarrollo tecnológico en la sociedad, ya que se encuentran disponibles para todos.
- La inversión en infraestructura es baja, comparadas con otras plataformas.

Porque es importante en la educación?

- Hemos visto un completo sistema de desarrollo disponible usando *Software libre*.
- *Método* aplicable a muchas escuelas y colegios en donde se imparte/enseña *electrónica e informática*.
- Aplicables a *docencia y a laboratorios de investigación* en Universidades e Institutos profesionales.
- Incentiva el desarrollo tecnológico en la sociedad, ya que se encuentran disponibles para todos.
- La inversión en infraestructura es baja, comparadas con otras plataformas.
- Incentiva la cooperación, colaboración y la libertad del conocimiento.

Porque es importante en la educación?

- Hemos visto un completo sistema de desarrollo disponible usando *Software libre*.
- *Método* aplicable a muchas escuelas y colegios en donde se imparte/enseña *electrónica e informática*.
- Aplicables a *docencia y a laboratorios de investigación* en Universidades e Institutos profesionales.
- Incentiva el desarrollo tecnológico en la sociedad, ya que se encuentran disponibles para todos.
- La inversión en infraestructura es baja, comparadas con otras plataformas.
- Incentiva la cooperación, colaboración y la libertad del conocimiento.
- Es una **real alternativa** para hacer un verdadero **Desarrollo de Electrónica**.