

Desarrollo Tecnológico con Open-Source

Desde hace años una de mis pasiones ha sido el desarrollo de electrónica digital, recuerdo que en mis comienzos no había nada más que un lápiz y un papel... lo demás era mucho tiempo de pruebas y ensayos, para finalmente terminar lo que se deseaba lograr. Era una época en que sólo era posible realizar desarrollos con altas inversiones en sistemas y equipos especializados que tenían un altísimo costo, lo que hacía que sólo ciertos privilegiados tuviesen a su disposición este tipo de sistemas y herramientas de trabajo, época en que incluso en las Universidades tenían laboratorios no tan nutridos.

Afortunadamente el desarrollo de la tecnología, la electrónica y la informática han dejado atrás esas épocas y proporciona para cualquier persona una de las mejores herramientas disponibles hoy en día para realizar desarrollo de esta maravillosa área de la tecnología, donde está presente, la electrónica, la ingeniería de software y la manufactura del sistema.

Hoy en día en que nuestro país tiene unos estupendos índices de crecimiento económico, en que sirve de ejemplo a los países de la región y tiene una imagen consolidada, sin embargo también tiene un grave problema para alcanzar ese ansiado desarrollo que es capaz de catapultar a los países al verdadero desarrollo económico y tecnológico, como lo son Holanda, Finlandia, Suecia, por indicar sólo algunos, que comparativamente con nosotros tienen menos recursos naturales y menos población que nuestro país, entonces porqué no estamos a la altura de ellos o no tenemos ni la sombra de su desarrollo?

En países en vía de desarrollo como el nuestro hay esfuerzos que debemos hacer cada uno de nosotros para sustentar un verdadero desarrollo tecnológico y no sólo esperar a los gobiernos y "políticos, tecnócratas" que, muchas veces, hacen tratados y convenios que no van en esa dirección - de hecho, van en la dirección opuesta - dejándose encandilar por las grandes corporaciones que sólo buscan un lucro económico.

Nuestro país, a nivel internacional, no tiene una imagen de país productor de tecnología, sino de productor de materias primas y alimentos: somos excelentes exportadores de cobre (que después recompramos como conductores eléctricos), frutas, salmónes, vino entre otros productos básicos, pero no

hacemos ningún desarrollo tecnológico y nos contentamos de importar tecnología que usamos, bien o mal ya que no se adaptan en muchos casos a nuestras necesidades reales, y terminamos adaptando los procesos productivos a ese desarrollo foráneo, porque no hemos sido capaces de producir nuestra propia tecnología y enfrentar ese problema en forma seria como país.

Tal vez un error de esta visión, en que todo lo que importamos es mejor que lo que nosotros somos capaces de producir, tenga raíces históricas ya que las familias aristocráticas de la colonia mandaban a sus hijos a estudiar a Europa, llegando después a gobernarnos, quedando así muy arraigada la sensación de que lo importado es mejor que lo nuestro.

Ciertamente no podemos pretender tampoco que de la noche a la mañana, mágicamente, nos transformemos en un país productor de tecnología ya que eso requiere de políticos y políticas adecuadas y de un planteamiento concreto en esa dirección, tal como lo han hecho en su momento Japón, Corea o China, que pasaron de países pobres y algunos destruidos por la guerra a potencias mundiales en desarrollo tecnológico. Su metodología inicial fue plagar a cualquier precio de sus "malos productos" a los mercados del tercer mundo, con ello consiguieron recursos económicos que reinvirtieron en educación y especialización y hoy en día producen tecnología de alta calidad, en áreas en donde se requiere un alto grado de especialización y conocimiento.

Nuestro querido país tiene un escenario distinto, no tenemos grandes fabricas, no producimos grandes volúmenes, no tenemos un mercado interno que nos permita crecer, por lo que nos queda sólo especializarnos y aplicar "mucho inteligencia" a nuestros productos y fabricarlos si es el caso en fábricas de producción masiva, pero ya no podremos competir con países productores de tecnología como los indicados.

Ésta es mi visión del estado actual, queda mucho por hacer en nuestro país desde el punto de vista tecnológico y de mentalidad aún, el esfuerzo debe partir por la base, las personas, cada uno de nosotros, solo así lograremos un cambio general, solo así podremos fomentar este desarrollo tecnológico.

Porqué el Open-Source

Últimamente, han estado sonando bastante en el ambiente estas tecnologías como sistemas operativos libres, código abierto, bases de datos abiertas y muchos conceptos que se ven como nuevos, pero la realidad es que llevan muchos años y han alcanzado un alto nivel técnico y gran madurez, lo que los convierte actualmente en una poderosa y atractiva alternativa para el desarrollo.

Dentro de los exponentes que más han ayudado a que los usuarios les comience a interesar este tema ha sido el uso masivo de Linux en los distintos ámbitos del que hacer diario, como las Universidades que fueron los primeros en adoptarlos, las escuelas, algunos organismos públicos y empresas en que primo su gratuidad, la inmunidad a los virus informáticos y su alta calidad.

Cuando partí con ésto ya hace algunos años una de las cosas que más me llamó la atención fue que los desarrolladores, valoran por sobre todo la calidad técnica del código, la colaboración y la generación de comunidades en su entorno, en que lo primordial es compartir el conocimiento en forma desinteresada y para cualquier persona que lo quiera adoptar sin importar su sexo, nivel económico, estado civil, religión o inclinación política, en que lo que debe primar es el respecto entre las personas.

Con ésto, se ha quebrado el oscurantismo del conocimiento, ya que todo este mundo está realmente al alcance de todos, en donde cualquier persona con cierta inquietud intelectual puede lograr realizar desarrollos muy profesionales, lo que convierte a estos sistemas en unas poderosas herramientas para quienes las quieran usar.

La Electrónica

La electrónica digital siempre ha sido una área en la cual se han requerido de herramientas especializadas y cada vez han ido en crecimiento.

Actualmente a diferencia de hace algunos años atrás, los ingenieros de esta área deben saber algo más que de electrónica pura, ya que las especialidades se están cruzando cada vez más, hoy no es suficiente saber sólo de electrónica, ya que el desarrollo de la microelectrónica esta muy relacionado con el software, los microcontroladores hoy en día han alcanzado un alto desarrollo, no es raro encontrar hoy microcontroladores que desarrollan 25MIPS (Millones de Instrucciones Por Segundo), lo cual era absolutamente

irrealizable hace no mucho tiempo atrás, convirtiendo a estas unidades en poderosas herramientas de control, por su alta capacidad de calculo, su rapidez y su simplicidad a la hora de usarlas.

Todo ello ha llevado a los ingenieros electrónicos a seguir especializándose en diferentes áreas, tales como la ingeniería de software, los compiladores, de lenguajes de programación, linkeadores, comunicaciones digitales, redes y muchas otras especialidades afines.

En conjunto con ésto, el desarrollo mundial del Open-Source ha proporcionado una batería de herramientas especializadas para el desarrollo de la electrónica, las que se encuentran en su mayoría dispersas ya que están enfocadas a resolver problemas específicos, pero es perfectamente posible trabajar con ellas, formando un potente sistema de desarrollo, que abarca desde el circuito esquemático, el desarrollo del código binario usando un lenguaje de alto nivel como es el lenguaje 'C', y el assembler en bajo nivel, la programación de los dispositivos con diferentes esquemas de programación.

Mi idea de plantear todo lo anterior, *no ha sido asustar a los posibles desarrolladores*, sino más bien mostrarles que este universo tiene mucho que dar a los inquietos de mente que les interesa experimentar, probar, desarrollar, concretar sus ideas, teniendo por delante un campo importante de desarrollo.

Primer Desarrollo

Este documento sólo pretende entregar los pasos iniciales para que los usuarios de Linux, entusiastas y amantes de la electrónica digital, sepan como iniciarse en este apasionante mundo del desarrollo de sistemas embebidos.

Espero que éste documento le evite a más de alguno andar vagabundeando por la red en busca de información decente y puedan ahorrar un montón de tiempo en su búsqueda; si.... yo lo perdí, ya no hay como recuperarlo.

Como siempre quedo atento a sus comentarios y si hay usuarios que puedan colaborar con este documento, estaré gustoso de recibir sus críticas y observaciones a su respecto.

En la medida de lo posible trataré de incluir otros documentos futuros que puedan ayudar aún más a este apasionante mundo del desarrollo de sistemas embebidos. Así mismo invito a los usuarios que quieran ayudar a ponerse en contacto y poder incluir más

documentación en nuestra lengua.

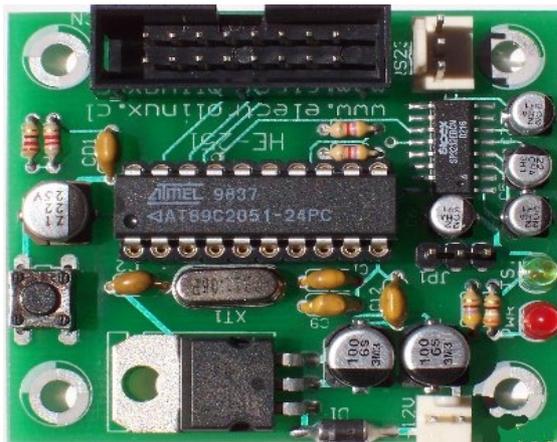
Supondré que tenemos nuestro Debian Etch ya instalado en nuestra máquina, con ambiente X perfectamente configurado y funcionando. No es importante si usa KDE, Gnome, XFce4 u otro sistema de ventanas o ambiente de escritorio, sólo basta que funcione. Pero bueno, basta de palabrería y pongámonos manos a la obra.

Plataforma de Desarrollo

Usaremos ahora en este ejemplo la familia MCS51, por su tremenda popularidad y por su extenso uso por años.

Para ello usaremos un kit, el HE-251 por su bajo costo, que tiene un procesador de esta familia y que es un muy buen microcontrolador para uso de novatos y expertos, por sus características y su versatilidad.

http://www.olimex.cl/product_info.php?products_id=287



Placa de Desarrollo HE-251

Instalando el Software

Para instalar las herramientas de desarrollo de este microcontrolador, realizaremos los siguientes comando en una consola como root.

```
Como root en nuestra maquina:  
#apt-get install sdcc sdcc-doc sdcc-libraries
```

Así de simple, por supuesto si usamos una distribución de Linux distinta, como Fedora, Mandriva, Suse, están los RPM para dichos paquetes.

Para verificar si la instalación ha quedado bien sólo debemos ejecutar como usuario normal con nuestra cuenta:

```
$ sdcc -version
```

Nos entregara en una linea lo siguiente:

```
SDCC :  
mcs51/gbz80/z80/avr/ds390/pic16/pic14/TININ :  
ative/xa51/ds400/hc08 2.6.0 #4309 (Oct 15 :  
2006) (UNIX)
```

Con lo anterior, hemos comprobado lo simple que fue instalar el compilador de lenguaje 'C' para esta arquitectura.

Ahora realizaremos nuestro primer programa que es un simple ejemplo de como realizarlo, y lo más simple posible, el encendido de un LED, que es el equivalente a "hola mundo" cuando programamos.

Código lenguaje C

Para ello escribiremos un programa llamado *led_onoff.c*, el cual esta compuesto por el siguiente código:

```
#include <at89x051.h>  
  
void delay(unsigned char x);  
  
void main(){  
    P3=0xFF; //Definimos como salida  
  
    while(1){  
        P3_7 = 0;  
        delay(200);  
        P3_7 = 1;  
        delay(200);  
    }  
  
void delay( unsigned char x){  
    unsigned char a;  
    for(a=0 ; a < x ; a++ );  
}
```

Con este código, hemos definido como salida el puerto 3 del microcontrolador hemos generado un delay() y un cuerpo principal del código que es main(), este simplemente sube y baja un bit de salida del port P3 (P3.7), cada cierto periodo de tiempo.

Como compilar el código

Ahora lo que nos queda es compilar el programa que hemos escrito en lenguaje 'C'. Para ello realizaremos lo siguiente:

```
$sdcc --model-small --stack-auto led_onoff.c
```

Lógicamente debemos correr este comando en el mismo directorio en donde tenemos el código fuente del programa en 'C'.

Que hemos obtenido de la compilación

El comando anterior genera varios archivos que son el resultado del archivo fuente, los que indico a continuación:

```
led_onoff.asm
led_onoff.ihx
led_onoff.lnk
led_onoff.lst
led_onoff.map
led_onoff.mem
led_onoff.rel
led_onoff.rst
led_onoff.sym
```

Si analizamos el contenido de algunos archivos, nos podemos dar cuenta que hay una rica información del uso de la memoria, de la conversión del código fuente a código de máquina, del assembler, en fin de varios detalles técnicos, que sin duda no a muchos les interese pero esta para ellos a los que si nos interesa.

Si nos damos cuenta en el listado anterior no se ha generado el archivo "*led_onoff.hex*", que es el formato intel del código binario que es lo que grabaremos en la memoria de programa de nuestro controlador.

Convirtiendo de ihx a hex

Requerimos ahora realizar una conversión de formato, ya que la grabación del binario se hace en formato HEX en el microcontrolador, pero si nos damos cuenta en la compilación no nos ha generado un archivo con este formato, por lo que deberemos convertirlo.

Esta conversión en un simple comando que debemos realizar, éste es:

En una consola ejecutamos:

```
$ packihx led_onoff.ihx > led_onoff.hex
```

Esto generará un archivo "*led_onoff.hex*", el cual es el que deberemos usar para programar la memoria de programa del microcontrolador.

Recordemos que lo que deberemos grabar en nuestro microcontrolador es un archivo binario, que contiene el programa en lenguaje de máquina del microcontrolador usado, en este ejemplo el del *MCS-51*.

Programando el Microcontrolador

Hay a lo menos dos formas de programar estos dispositivos. Una, que es la forma clásica, es mediante *Programadores Digitales*, que son productos de un costo más o menos considerable. La segunda forma es usando un programador construido por nosotros mismos, la cual me complace bastante más.

Para ello simplemente programaremos la memoria de programa (flash) del microcontrolador usando un dispositivo por la puerta serial. Lo que detallaremos en un artículo posterior.

Palabras finales

Espero haber sido claro en esta explicación y que hayan entendido porque es importante realizar desarrollos con tecnologías abiertas, el porque hace falta que nuestros países llamados del tercer mundo, hagan desarrollo, porque es importante que los ciudadanos hagan desarrollo de tecnología, porque es importante que los políticos sean personas que entiendan lo que hablan y tome decisiones acertadas.

En artículos posteriores, si a ustedes les ha interesado estos temas, podremos hablar las de otras arquitecturas, de aplicaciones, de otras formas de desarrollar tecnología a bajo costo.

Podremos hablar de las aplicaciones, de robótica, de como controlar pequeños procesos o de como realizar algún tipo de proyecto doméstico.

Pero todo depende de la aceptación de este artículo, y del interés que haya despertado sobre estos temas. Para ello pueden enviar un correo a los editores de esta revista o a mi correo personal, estaré gustoso de responder sus consultar y opiniones.

Hasta siempre..

Atte.

Ricardo Albarracin B.

ralbab@gmail.com