



Transferencia tecnológica y de conocimientos en el diseño de sistemas embebidos

Carlos Iván Camargo Bareño

Universidad Nacional de Colombia
Facultad de Ingeniería, Departamento de Ingeniería Eléctrica y Electrónica
Bogotá, Colombia
2011

Transferencia tecnológica y de conocimientos en el diseño de sistemas embebidos

Carlos Iván Camargo Bareño

Tesis presentada como requisito parcial para optar al título de:
Doctor en Ingeniería Eléctrica

Director:
Ph.D. Luis Fernando Niño Vásquez

Línea de Investigación:
Transferencia Tecnológica
Universidad Nacional de Colombia
Facultad de Ingeniería, Departamento de Ingeniería Eléctrica y Electrónica
Bogotá, Colombia
2011

A mis padres, a paito por su confianza ciega y por estar siempre disponibles en todo momento

A mi amada esposa, por su infinita paciencia, comprensión y ternura, y por último a mi amado y esperado camarguito por darme una razón más para hacer de este país un mejor sitio para vivir.

Agradecimientos

Deseo expresar mis más sinceros agradecimientos a Luis Fernando Niño, quien sin su ayuda y confianza no habría podido terminar este trabajo. Asimismo, quiero agradecer a mis amigos del Departamento de Ingeniería Eléctrica y Electrónica (John Jairo, Iván, Sebastian, Carlos Sanchez, Pablo, John) por su constante apoyo; a Andrés Calderón compañero leal en este proceso de aprendizaje, mil gracias por su ayuda desinteresada; y a mis estudiantes quienes me han enseñado que el componente humano es lo más importante de todo proceso de enseñanza.

Resumen

El presente trabajo pretende ser el punto de partida de un proceso de transferencia tecnológica y de conocimientos exitoso en el área de diseño digital de sistemas embebidos que contribuya a dar solución al problema de atraso y dependencia tecnológica de la industria electrónica nacional. Para lograr esto, se aplicó una metodología informal y pública para transferencia tecnológica con el fin de: detectar las necesidades del país; adquirir, asimilar y aplicar los conocimientos necesarios para producir sistemas digitales que utilizan tecnologías y metodologías de diseño modernas. Utilizando la capacitación como canal para la transferencia se creó un plan de estudios para la línea de electrónica digital que ayuda a formar en los profesionales las habilidades necesarias para concebir, diseñar, implementar y operar sistemas digitales. Para ayudar en el proceso de enseñanza/aprendizaje y con el fin de facilitar la difusión en la industria, se diseñaron una serie de plataformas hardware que pueden ser utilizadas como diseños de referencia en la fabricación de nuevos productos, la información necesaria para entender, reproducir, modificar y programarlas se encuentra disponible a quien esté interesado. Finalmente, tomando como referencia el movimiento de software libre y código abierto se difundirán los resultados de este trabajo a todos los sectores de la sociedad.

Palabras clave: Transferencia tecnológica, sistemas embebidos, diseño digital, educación en ingeniería .

Abstract

This work aims to be the starting point of a successful technology and knowledge transfer process in the area of embedded systems design helping to solve the problem of backwardness and technological dependence of the domestic electronics industry. To achieve this, we applied an informal and public method for technology transfer in order to: identify the needs of the country, acquire, assimilate and apply the knowledge necessary to produce systems using modern design methodologies. Using training as a channel for the transfer we created a curriculum for digital electronics line that helps form the professional skills necessary to conceive, design, implement and operate digital systems. To assist in the teaching / learning and to facilitate the spread in the industry, we designed some hardware platforms that can be used as reference designs in the manufacture of new products, the information needed to understand, reproduce , modify and program is available to anyone. Finally, with reference to the free software movement and open source disseminate the results of this work to all sectors of society.

Keywords: technology transfer, embedded systems, digital design, engineering education

Contenido

Agradecimientos	VII
Resumen	IX
Siglas	XVII
1. Introducción	2
1.0.1. Sistemas Embebidos	3
1.0.2. ¿Por qué los Sistemas Embebidos?	4
1.1. Estado de la Industria Electrónica en Colombia	4
1.2. El Conocimiento como Bien Público	7
1.3. Propuesta de Metodología para la Transferencia Tecnológica y de Conocimientos . .	9
2. Transferencia Tecnológica	12
2.1. Introducción	12
2.2. Tecnología	12
2.3. Concepto de Transferencia Tecnológica	14
2.3.1. Tipos de Transferencia Tecnológica	14
2.3.2. Canales para la Transferencia de Tecnología	15
2.4. Obstáculos para una Transferencia Exitosa	17
2.5. Recomendaciones para una Transferencia Exitosa	18
2.6. Metodología Propuesta	21
2.6.1. Etapas de la Metodología	21
2.6.2. Factores Formales e Informales	24
3. Implementación de la Metodología	27
3.1. Introducción	27
3.1.1. Qué es un Sistema Embebido	27
3.1.2. Arquitectura	27
3.1.3. Aplicaciones	28
3.1.4. Metodología de Diseño	29
3.2. Implementación de la Metodología	31
3.2.1. Selección de la Tecnología	31
3.2.2. Adquisición de Tecnología	35
3.2.3. Adaptación	38
3.2.4. Absorción y Asimilación	47
3.2.5. Aplicación	55
3.2.6. Difusión y Desarrollo	58

4. Plan de Estudios Para la Enseñanza/Aprendizaje de Sistemas Embebidos	65
4.1. Introducción	65
4.1.1. La Iniciativa CDIO	66
4.1.2. Estructura del Plan de Estudios CDIO	67
4.2. Implementación del Plan de Estudios CDIO	68
4.2.1. Definición e Identificación de las Habilidades CDIO	69
4.3. Integración de las Habilidades CDIO al Plan de Estudios	71
4.3.1. Metodología de Diseño	71
4.3.2. Objetivos de Aprendizaje - Dominio Cognitivo	71
4.3.3. Dominio Afectivo	76
4.3.4. Dominio Psicomotor	77
4.3.5. Metodología	77
4.3.6. Integración de SIE con los Cursos de la Línea de Electrónica Digital	79
4.4. Desarrollo de Métodos de Evaluación	82
4.5. Actividades	83
4.6. Discusión	85
5. El Conocimiento Como Bien Público y su Papel en la Difusión Tecnológica	88
5.1. Introducción	88
5.2. El Conocimiento como Bien Público	89
5.2.1. Auto-Gobierno	89
5.3. Movimiento Software Libre y Código Abierto (FOSS)	92
5.4. Hardware Copyleft	95
5.4.1. Reglas	96
5.4.2. Recursos	99
5.5. Flujo de Diseño Utilizando <i>Hardware Copyleft</i>	102
5.6. Discusión	104
6. Conclusiones y trabajo Futuro	107
6.1. Trabajo Futuro	109
A. Relaciones con la industria y la Academia.	111
A.1. Introducción	111
A.2. Relaciones con la industria	111
A.2.1. emQbit ltda.	111
A.2.2. e-modust	124
A.2.3. Importex	124
A.2.4. em-electronics	124
A.2.5. Microensamble	124
A.2.6. SAR SA	125
A.3. Relaciones con la academia	127
A. Artículos	128
A.1. Aplicaciones	128
A.1.1. En Hardware Evolutivo	128
A.1.2. Aplicaciones en Robótica	128

A.2. Sobre la Metodología 128

Bibliografía 129

Lista de Figuras

1.1. Clasificación general de los bienes. fuente: [1]	8
1.2. Modelo de transferencia tecnológica.	9
2.1. Modelo del proceso de transferencia de tecnología indicando la composición de los canales formales e informales	13
2.2. Resumen de las recomendaciones para una transferencia tecnológica exitosa	20
2.3. Etapas de la metodología propuesta para la transferencia tecnológica y de conocimientos en el área de diseño de sistemas embebidos	21
2.4. Canales formales e informales del proceso de transferencia tecnológica	24
3.1. Arquitectura de un Sistema Embebido	28
3.2. Aplicaciones de los Sistema Embebidos Fuente: TATA Consultancy services	29
3.3. Flujo de diseño de un sistema embebido [2]	30
3.4. Comparación del uso de sistemas operativos Fuente: Venture Development Corp	36
3.5. Arquitectura típica de un sistema embebido que utiliza SoC con memoria volátil interna	40
3.6. Arquitectura típica de un sistema embebido que utiliza SoC sin memoria volátil interna	41
3.7. Flujo de diseño software para creación de aplicaciones.	42
3.8. Inicialización de un SoC cuando las memorias no volátiles no están programadas.	43
3.9. Inicialización de un SoC cuando la memoria no volátil está programada, parte 1	44
3.10. Inicialización de un SoC cuando la memoria no volátil está programada, parte 2	44
3.11. Cadena Boundary Scan fuente: Texas Instruments.	45
3.12. Estructura del kernel de Linux.	46
3.13. Plataforma de desarrollo ECB_ARM7	48
3.14. Plataforma de desarrollo UNAL_UIS_XPORT	49
3.15. Plataforma de desarrollo ECB_AT91 V1	49
3.16. Plataforma de desarrollo ECB_AT91 V2	50
3.17. Plataforma de desarrollo ECBOT	51
3.18. Plataforma de desarrollo ECB_BF532	51
3.19. Plataforma de desarrollo SIE	52
3.20. Plataforma de desarrollo STAMP	52
3.21. Flujo de diseño para las placas de circuito impreso	53
3.22. Flujo de actividades para la apropiación de conocimiento	55
3.23. Funcionalidad del sistema distribuido de control de versiones git	59
4.1. Bloques constructores de conocimiento, habilidades y actitudes necesarias para concebir, diseñar, implementar y operar sistemas en el contexto social y empresarial fuente:[3]	67
4.2. Objetivos, actividades, y evaluación:	68
4.3. Metodología de diseño para el área de Sistemas Digitales	71

4.4. Estructura de la plataforma de desarrollo SIE	79
4.5. Flujo de diseño hardware	80
4.6. Flujo de diseño hardware/software al utilizar un procesador <i>softcore</i>	81
5.1. Marco para analizar el conocimiento como bien común aplicado al movimiento FOSS. Fuente: [4] pág 46	93
5.2. Marco para analizar el conocimiento como bien común aplicado al movimiento hardware copyleft. Adaptación al hardware copyleft de: [4]	96
5.3. Miembros de la comunidad linuxencaja.	101
5.4. Adaptación de la plataforma de desarrollo ECB_AT91 (a) a un monitor de signos vitales (b) Fuente: [5]	102
5.5. Flujo de diseño al utilizar hardware copyleft	103
5.6. Modelo de transferencia tecnológica aplicado a la empresa emQbit y a la academia .	105

Lista de Tablas

3.1. Arquitectura de las plataformas comerciales utilizadas en la etapa de absorción	41
3.2. Conocimientos Adquiridos durante la etapa de adaptación	46
3.3. Características de las plataformas de desarrollo concebidas, diseñadas e implementadas	53
3.4. Comparación de precios entre la transferencia ofrecida por Corea y el costo utilizando los conocimientos generados en este trabajo	63
4.1. Competencias para los niveles 2 y 3 CDIO	70
4.2. Competencias para CDIO	70
5.1. Características de las plataformas de desarrollo <i>hardware copyleft</i> proporcionadas por el presente estudio	104

Siglas

3G	3rd generation mobile telecommunications. 102
ADC	Analog-to-digital converter. 40
ASIC	Application-specific integrated circuit. 27, 31–33
BSP	Board Support Package. 103
CAD	Computer-Aided Design. 98
CAN	Controller area network. 42
CDIO	Concebir, Diseñar, Implementar, Operar. 10, 23, 56, 65–68
CEO	Chief executive officer. 34
CIIU	CODIGO INDUSTRIAL INTERNACIONAL UNIFORME. 57
CMUA	Centro de Microelectronica de la Universidad de los Andes. 101
CODEC	codificador-decodificador. 42
COTS	Commercial off-the-shelf. 37
CPLD	Complex programmable logic device. 27, 28, 32, 54
CPU	Central processing unit. 27, 41, 42, 53, 54, 80
DDR	Double Data Rate). 40, 41, 52
DIEE	Departamento de Ingeniería Eléctrica y Electrónica. 65, 86
DMA	Direct memory access. 40
DSP	Digital signal processor. 27, 31, 33–35, 50
ECI	Escuela Colombiana de Ingeniería. 86
EEPROM	Electrically-Erasable Programmable Read-Only Memory. 40
ENAP	Escuela Naval Almirante Padilla. 86
ES	Sistema Embebido. 27
FOSS	Free and Open-Source Software. 3, 9, 88, 92–95
FPGA	Field Programmable Gate Array. 27, 28, 31–35, 39, 50, 51, 54, 78, 79, 81, 100, 104
GNU	GNU is Not Unix. 42, 80, 92
GPIO	General Purpose Input/Output. 49, 50, 78
GPL	General Public License. 94, 97, 98

GPS	Global Positioning System. 102
GSM	Global System for Mobile communication. 102
HDL	Hardware Description Language. 32
HTML	HyperText Markup Language. 61
HTTP	Hypertext Transfer Protocol. 124
I2C	Inter-Integrated Circuit. 40, 42, 49, 78
ICE	In Circuit Emulation. 45
IP	Intellectual Property. 37
ISP	In System Programming. 49
IT	Information Technology. 15
JTAG	Joint Test Action Group. 44, 45, 50, 78, 79
LCD	Liquid Crystal Display. 37, 40, 42, 78
LED	Light-Emitting Diode. 37, 50, 60, 99
MARTE	Modeling and Analysis of Real-Time and Embedded Systems. 29
MIT	Massachusetts Institute of Technology. 66
MMU	Memory management unit. 40
OEM	Original Equipment Manufacturer. 38
OS	Operating System. 39, 53
OSH	Open Source Hardware. 95
PC	Program Counter. 2, 44
PCB	Printed Circuit Board. 31, 34, 47, 53, 60, 78, 124
PIB	Producto Interno Bruto. 6
PLD	Programmable Logic Device. 27, 28, 31, 48, 74, 77, 78, 100
RAM	Random Access Memory. 40, 41, 43–46, 60, 104
ROM	Read Only Memory. 42, 43
RTL	Register-transfer level. 37
SBC	Single Board Computer. 101
SD	Secure Digital. 40, 42, 43, 52
SDRAM	Synchronous dynamic random-access memory. 40, 41, 52, 78
SoC	System On a Chip. 27, 28, 32–35, 37–46, 48–50, 56, 60, 72–75, 77, 80, 81, 100, 104
SPI	Serial Peripheral Interface. 40, 42
SSI	Serial Synchronous Interface. 40
UDFJC	Universidad Distrital Francisco José de Caldas. 23, 86

UIS	Universidad Industrial de Santander. 11, 23, 86
ULA	Universidad de los Andes. 23, 86
UML	Unified Modeling Language. 29
UNAL	Universidad Nacional de Colombia. 23, 86
USART	Universal asynchronous receiver/transmitter. 40
USB	Universal Serial Bus. 35, 40, 43, 78, 79, 104
USD	United States dollar. 33, 35, 38
USTA	Universidad Santo Tomás de Aquino. 23, 86
WiFi	Tecnología de redes inalámbricas. 102

1. Introducción

El gran avance de las técnicas de fabricación de circuitos integrados ha permitido que los sistemas digitales sean parte fundamental de la vida humana, facilitando las tareas cotidianas. Los niveles de integración actuales permiten construir sistemas cada vez más pequeños, veloces y de menor consumo de potencia, lo cual ha favorecido su difusión a casi todas las actividades humanas. A medida que se extiende el campo de aplicación de los sistemas digitales, lo hacen las exigencias de funcionamiento a ellos impuestas; nuevos retos en el diseño se presentan a medida que los sistemas embebidos se integran a las actividades humanas. Observando la tendencia actual de los sistemas electrónicos, se puede especular que el computador tal como se conoce actualmente desaparecerá [6], ya que estará en todas partes, ubicuo, interactuando con los seres humanos para realzar su mundo. Se pasará de un esquema en el que existe un computador para uno o varios usuarios (Program Counter (PC), mainframe) a uno en el que existan muchos computadores para un usuario. Estos computadores disponen de grandes capacidades de cálculo y de comunicación, pero a la vez, poseen un grado de integración tal que serán invisibles; para aclarar como se puede lograr esta invisibilidad, imagine que existen sistemas embebidos construidos con técnicas de micro-fabricación y que son capaces de tomar su energía de fuentes alternas como la temperatura, la radiación solar, o a partir de fenómenos químicos; debido a su reducido tamaño, estos sistemas pueden integrarse a objetos o pintarse sobre ellos, de tal forma que no sean visibles ante los ojos humanos. Esta desaparición no solo será una consecuencia de la tecnología, sino de la sicología humana; cuando las personas asimilan perfectamente algo y se convierte en parte de la vida diaria no se es consciente de su existencia y utilización.

Esta situación ha dado origen a una industria gigantesca que proporciona dispositivos que satisfacen las nuevas necesidades humanas, el volumen de producción de este tipo de aplicaciones es tal, que el número de procesadores utilizados en ellas ha superado ampliamente a los procesadores utilizados en aplicaciones tradicionales (como estaciones de trabajo y computadores personales); en muchas áreas, los procesadores para aplicaciones específicas los han desplazado por completo (por ejemplo, puntos de venta, entretenimiento, asistentes personales, equipos de automatización, etc). Esto, unido a la saturación en el mercado tradicional de software, ha originado una migración de muchas compañías y profesionales al diseño de aplicaciones específicas, donde el sistema está diseñado para cumplir una función determinada.

Gracias a esta enorme y creciente demanda, la industria de los semiconductores presenta una gran dinámica en la creación de nuevos productos que facilitan el desarrollo de nuevas aplicaciones; a diario se crean nuevos dispositivos de consumo masivo como reproductores MP4, cámaras de video, teléfonos celulares que se superan en capacidad y funcionalidad al paso de cortos períodos de tiempo. Esto ha originado una disminución dramática en el costo de los dispositivos semiconductores necesarios para construir estos productos y, a su vez, un aumento en la funcionalidad de los mismos; esto, en respuesta a la necesidad de un número cada vez menor de componentes para implementar este tipo de aplicaciones. Por otro lado, se generó un cambio en la forma de suministrar información por parte de los fabricantes sobre la programación y funcionamiento de sus dispositivos, pasando de un esquema de confidencialidad a uno totalmente abierto donde se suministra todo lo necesario

(manuales de programación, notas de aplicación, diseños de referencia y software de desarrollo) para entender y utilizar sus productos. Esta dinámica también se presenta en las herramientas de desarrollo necesarias para crear la funcionalidad deseada. Los sistemas propietarios se utilizan cada vez menos gracias a la oferta de entornos de desarrollo abiertos creados y mantenidos por una comunidad de desarrolladores que suministran sus conocimientos para el beneficio colectivo; fruto de esta tendencia se han creado aplicaciones tan importantes como el movimiento de software libre y de código abierto (Free and Open-Source Software (FOSS) por sus siglas en inglés), el sistema operativo Linux, y recientemente la plataforma para dispositivos móviles Android.

Todo lo anterior ha generado un clima adecuado para la creación de empresas que utilizan estas facilidades en el desarrollo de productos que puedan ser comercializados local o globalmente, y de esta forma contribuir a mejorar la calidad de vida de la sociedad. Un ejemplo de esto se puede observar en algunos países asiáticos, donde se fortaleció la industria manufacturera de dispositivos electrónicos hasta el punto de dar empleo a una gran cantidad de personas con diferentes niveles de formación y desplazar casi por completo la manufactura de este tipo de productos en el resto del mundo.

Desafortunadamente, la situación en países en vías de desarrollo como Colombia es totalmente diferente; el uso de tecnologías, metodologías de diseño y técnicas de fabricación obsoletas; falta de políticas gubernamentales que estimulen la transferencia real de tecnologías y del conocimiento asociado a ellas; la desconexión entre la academia con los problemas sociales del país y las necesidades de las empresas. Lo anterior ha sumido a la industria electrónica en un atraso de forma tal que depende por completo de productos manufacturados en el exterior, y la mayoría de empresas de base tecnológica se convierten en representantes de firmas extranjeras sin capacidad de creación local.

Los canales de transferencia tradicionales que se limitan a la compra de equipo y a la capacitación para usarlo y mantenerlo en operación no suministra los conocimientos necesarios para que en el país se generen productos similares adaptados al entorno social, político y económico local. Por lo que es necesario crear metodologías de transferencia que permitan asimilar el conocimiento asociado y pueda ser aplicado por diferentes sectores de la sociedad para mejorar su calidad de vida.

Este trabajo propone una metodología para la transferencia tecnológica y de conocimientos que busca a partir de interacciones locales generar una absorción y asimilación global de los conocimientos necesarios para el diseño, construcción y comercialización de sistemas digitales que utilicen los últimos avances en la industria de los semiconductores. Esta metodología se aplicó inicialmente en una universidad consolidada¹ y en una empresa de base tecnológica *emQbit Ltda*, para formar una comunidad que se encargue de crear herramientas para la difusión de los conocimientos adquiridos durante el proceso de transferencia (recurso). El carácter de la comunidad y del conocimiento que utiliza/genera/mantiene/depura es totalmente abierto, y cualquiera que esté interesado puede utilizar/generar/mantener/depurar dicho recurso y ser miembro de esta comunidad; es decir, el conocimiento es considerado como un bien público y el acceso a él es un derecho que debe ser garantizado.

1.0.1. Sistemas Embebidos

En la actualidad, cada vez con más frecuencia, se notan signos de la invasión digital, por ejemplo, en el aumento de chips embebidos en los dispositivos que se utilizan a diario. Se ha demostrado [7] que una persona que vive en un país industrializado se ve confrontada con un

¹Departamento de Ingeniería Eléctrica y Electrónica de la Universidad Nacional de Colombia

promedio de 40 chips al día, de los cuales 5 son capaces de comunicarse en redes. Se estima que en la actualidad una persona está en contacto con cientos de estos chips, la mayoría de los cuales acceden a densas redes de información[8]. Muchos de estos artefactos toman la apariencia de objetos que se utilizan en la vida diaria (herramientas, vestuario, electrodomésticos, etc) pero son mejorados con sensores, actuadores, procesadores y software embebido. Una de las razones de la aparición de estos sistemas es económica; las industrias han visto como se muestran signos de recesión en los mercados tradicionales. Por lo tanto, buscan nuevos productos en los que pueden ser embebidos chips y software.

La aparición de los dispositivos digitales en las actividades humanas, es una consecuencia lógica de la evolución de las relaciones entre los usuarios y los sistemas digitales, los cuales se han caracterizado por una democratización de acceso a los equipos y una descentralización de la infraestructura subyacente. En los 90s la aparición de Internet permitió compartir recursos a través de un computador personal. Internet no es más que un paso adelante hacia la llegada de los sistemas de computación ubicua. La misma filosofía de simplificación y descentralización prevalece hasta hoy y conduce a una situación donde miles de dispositivos computacionales estarán disponibles para realizar tareas cotidianas y se compartirán recursos a través de redes más intrincadas que Internet.

1.0.2. ¿Por qué los Sistemas Embebidos?

Mercado de los sistemas embebidos

Según BCC, inc.² el mercado de los sistemas embebidos es de \$92 billones de dólares y se estima que sea de \$112.5 billones en 2013, con una tasa compuesta de crecimiento anual (CAGR) del 4.1 %. El hardware embebido tuvo un valor de \$89.8 billones en 2008 y se espera que crezca a un CAGR del 4.1 % para alcanzar \$109.6 billones en 2013. El software embebido generó \$2.2 billones en 2008 y podrá aumentar a \$2.9 billones en 2013 para un CAGR del 5.6 %. Este estudio revela que USA, Canadá, y los países asiáticos dominan este creciente mercado.

Penetración de los sistemas embebidos

Más de un billón de dispositivos embebidos fueron vendidos en el 2004, siendo sus principales áreas de aplicación los computadores, las telecomunicaciones, consumo, automóviles, medicina, oficina, industria militar. Según BBC, inc. las ganancias en el 2009 son de aproximadamente 100.000 millones de dólares.

1.1. Estado de la Industria Electrónica en Colombia

En la actualidad Colombia atraviesa por una “crisis” a nivel de diseño de sistemas digitales, existe un atraso muy grande en esta área. Existen dos grandes responsables de esta situación; por un lado, las políticas de la mayoría de las industrias al no realizar inversión de capital en sus departamentos de I+D; por otro lado, algunas universidades no cuentan con programas actualizados que permitan explotar los avances realizados en las industrias electrónica y de semiconductores. La situación se agrava aún más al ver el estado de la relación entre la universidad y la industria, la cual no existe en algunos casos. Desde el punto de vista industrial los resultados obtenidos en la academia parten de entornos ideales y no se tienen en cuenta las características de los entornos industriales.

²<http://www.bccresearch.com/>

Apropiación de conocimiento y transferencia tecnológica

El estudio en [9] asegura que para que Colombia llegue a ser generador de productos tecnológicos, es necesario generar un conocimiento que permita esta transición. “Para que el conocimiento sea motor de desarrollo es necesario el traspaso desde sus creadores a la sociedad, mediante la conversión a tecnologías que produzcan cambios radicales que incrementen la producción. Esa transmisión de tecnología generadora de crecimiento económico está influenciada por diversos factores: medio geográfico, leyes de propiedad industrial, costos laborales, nivel de ciencia y tecnología, religión, tipos de instituciones, resistencia a innovar, políticas de estado, guerras, factores demográficos, entre otros” [9].

Pero ¿cómo obtener este conocimiento? Arrow [10] afirma que la obtención de conocimiento puede efectuarse de varias formas: “aprender haciendo”, “aprender usando”, “aprender leyendo”. Cuando una empresa decide transmitir su conocimiento disponible, lo hace en procesos de investigaciones conjuntas, en actividades de producción, y distribución, mercadeo, servicio y soporte operativo o riesgo compartido. También se presentan alianzas entre firmas como: contratos de I+D, acuerdos de licencias y licencias cruzadas. La conformación de estas asociaciones permite crear redes tecnológicas dominadas por países industrializados con sus respectivas empresas multinacionales monopolizando conocimiento [9].

El problema colombiano radica en que las empresas nacionales no adquieren el conocimiento necesario para innovar; de forma que puedan ser competitivas y puedan acceder a mercados internacionales ofreciendo nuevos productos de calidad y a precios competitivos, con efectos directos como: generación de empleos especializados, desarrollo tecnológico e industrial sostenido, ampliación del conocimiento nacional y disminución de la salida de divisas [9].

Según los estudios realizados por Martínez [9], con base en registros del Decreto 259/92, del Incomex, la importación de conocimiento no está siendo empleada con el propósito de utilizar tecnologías de punta que permitan efectuar innovaciones al interior de las empresas y de los sectores; las empresas nacionales se limitan a comprar un determinado dispositivo o equipo, sin tener el conocimiento para operarlo, hacerle mantenimiento, mejorarlo, ni mucho menos producirlo, por lo que se ven obligadas a contratar con el vendedor para dicho fin; esto indica, que la adquisición de tecnología no se realiza con base en un programa desarrollado de antemano, sino son una respuesta a cambios en el mercado, lo cual evidencia la inexistencia de programas de innovación encaminados a la disminución de la brecha tecnológica.

Situación de la industria electrónica en Colombia

La industria electrónica nacional no es ajena a las políticas que siguen las empresas nacionales en cuanto a la apropiación de tecnología; Colombia depende totalmente de economías más desarrolladas para el suministro de dispositivos electrónicos en diversas áreas (comunicaciones, entretenimiento, industria, medicina, etc). Mientras en otros sectores de la economía (Alimentos, textiles, bebidas) han pasado de ser consumidores a exportadores, y adquieren nuevas tecnologías para ser más competitivos, el sector electrónico del país ha reducido sus actividades de investigación y desarrollo hasta el punto de depender totalmente de productos externos, de los cuales, algunos son de baja calidad y no suplen los requerimientos del mercado local, pero son utilizados porque son muy económicos.

En la actualidad la industria electrónica presenta una gran dinámica a nivel mundial [11]. La inversión de capital necesaria para el diseño de sistemas digitales es relativamente baja, gracias a la gran demanda originada y a la constante reducción de los costos asociados a los servicios

de fabricación; adicionalmente, las herramientas de desarrollo necesarias para la programación y depuración de este tipo de sistemas son de libre distribución, lo que permite reducir de forma considerable la inversión en este ítem.

Según ASESEL³ en el 2001⁴ existían 154 empresas productoras de componentes y equipos de la cadena electrónica; dentro de los productos que la industria electrónica exporta se encuentran registrados: circuitos integrados, circuitos impresos, micro-estructuras, (no son fabricados en el país, son compradas en el exterior y vendidas nuevamente) instrumentos para medida y control, instrumentos y aparatos eléctricos o electrónicos. Según Proexport, el 91 % de las exportaciones son realizadas por Bogotá y los destinos se encuentran en países cercanos como Venezuela, Perú, Ecuador y USA. La producción durante el período 2000 - 2007 aumentó en un 4 % llegando a 936 mil millones de pesos, las exportaciones aumentaron en un 15 % llegando a 1093 mil millones de pesos y las importaciones subieron en un 18 % llegando a 13.262 mil millones en el mismo período de tiempo.

Causas del atraso tecnológico en Colombia

Estudios realizados en Colombia [12] [13] [14] y en otros países [15] [16] [9] identificaron los siguientes obstáculos para el desarrollo de la industria electrónica: deficientes relaciones universidad - empresa, pobre enfoque académico hacia la industria, baja calidad de los productos nacionales, falta de políticas gubernamentales, falta de cultura de investigación, reducida apropiación tecnológica, competencia con productos extranjeros, atraso tecnológico y limitado recurso humano con formación avanzada.

De los problemas expuestos anteriormente se pueden decir que los que más afecta al desarrollo de la industria electrónica en Colombia es el atraso tecnológico; no es posible ser competitivo en el mercado electrónico mundial con tecnologías y metodologías de diseño obsoletas. La culpa de este atraso tecnológico no es exclusiva de la industria, la falta de confiabilidad en los productos colombianos agrava este problema.

Otro actor que contribuye al retraso tecnológico es el sector académico; según el Sistema Nacional de Información Superior, durante los últimos 10 años se han abierto 230 programas relacionados con la industria electrónica, estos programas están repartidos entre programas de formación universitaria, tecnológica terminal y técnica profesional. La mayoría de los centros de formación se encuentran ubicados en 3 departamentos: Bogotá, Antioquia y Valle [14]. El número de ingenieros graduados en un año es entre 2 y 8 veces mayor que en los países en vías de desarrollo y doce veces mayor que los que se graduaron en los países desarrollados [13].

Por otro lado, en las universidades con poca consolidación, el contenido y metodologías de las asignaturas relacionadas directamente con la industria electrónica se encuentran desactualizados y fuera del contexto mundial; se utilizan metodologías de diseño antiguas en las que primaba la experiencia del diseñador, se realizan tareas manuales, repetitivas que pueden ser realizadas por herramientas de diseño moderno, hay poca experimentación y su estructuración y metodologías son anticuadas. Adicionalmente, muchos investigadores dedican sus estudios en proyectos que no aportan al desarrollo del país pero que les proporcionan más reconocimiento internacional. Otro

³ Asociación de entidades del Sector Electrónico

⁴ Desde este año no se ha realizado ningún estudio sobre la industria electrónica. Por otro lado, debido a que no existe un código para la clasificación industrial internacional uniforme (CIIU) específico para la industria electrónica, las empresas del sector electrónico se encuentran dispersas y la gran mayoría comparten la clasificación *no clasificadas previamente* (NCP), su estudio se dificulta. Adicionalmente, la industria electro-electrónica hace un aporte muy bajo al Producto Interno Bruto (PIB) y no tiene mayo peso en la industria nacional (fuente: ASESEL)

problema radica en la falta de experiencia en el sector productivo por parte del personal académico, una componente importante del grupo de profesores nunca ha sido parte de un proceso productivo o de un proceso de desarrollo que tenga como fin la creación de un producto comercial, razón por la cual se evita la experimentación y se da más énfasis al análisis, siendo la simulación el último paso del proceso de análisis. Se puede decir que en Colombia se presenta una sobre-oferta de profesionales en el área electrónica, formados con programas desactualizados que no tienen en cuenta los avances tecnológicos y metodológicos; lo que explica la falta de ingenieros con altos niveles de formación y la escasa demanda de profesionales en el área.

1.2. El Conocimiento como Bien Público

Es indudable que el desarrollo tecnológico de un país se encuentra ligado al mejoramiento de la calidad de vida de sus habitantes, y que para que en un país en vías de desarrollo se realice una transferencia tecnológica y de conocimientos asociados a la tecnología que se transfiere exitosos que permitan desarrollar productos similares, pero ajustados al contexto socio-económico local, es necesario que el país cuente con la capacidad de absorber las habilidades, técnicas, información y organización asociados. Esta absorción de conocimientos debe ser realizada por un gran número de personas y entidades para que la transferencia tenga un impacto significativo en la sociedad; para que esto suceda, este conocimiento debe ser considerado como un bien público, y como tal, el acceso a él, debe ser un derecho y por lo tanto, la sociedad debe garantizar los mecanismos de difusión para que llegue a los sectores interesados. De igual forma, es un deber de los sectores que utilizan este bien contribuir a su difusión, actualización, mejoramiento, y crecimiento. El conocimiento en esta propuesta se refiere a las ideas intangibles; información y datos en el que el conocimiento es expresado u obtenido, y como el entendimiento adquirido a través de la experiencia o estudio científico, académico o no académico.

Ostrom [1] propone dos variables generales para clasificar los diferentes tipos de bienes: *el nivel de exclusión* entendido como la dificultad para excluir el acceso a un recurso por parte de usuarios potenciales; y el *nivel de rivalidad* el cual se refiere al impacto que tiene el uso de un recurso por un individuo (o grupo) sobre el uso de otros usuarios. Ostrom, identifica cuatro tipos de bienes y/o servicios: *bienes públicos* bajo nivel de rivalidad y bajo nivel de exclusión; *bienes comunes* alto nivel de rivalidad y bajo nivel de exclusión; *club goods* bajo nivel de rivalidad y alto nivel de exclusión y *bienes privados* alto nivel de rivalidad y alto nivel de exclusión. El conocimiento es un recurso con un grado de rivalidad bajo, ya que no se afecta negativamente al aumentar el número de usuarios, lo que entonces deja la posibilidad de clasificarlo como un bien público o un bien tipo club (ver figura 1.1).

Idealmente, el conocimiento debería ser un bien público; esto es, no deberían existir restricciones para acceder a él; sin embargo, en la actualidad el acceso al conocimiento es restringido, ya sea por medio de patentes, derechos de propiedad intelectual (lo cual es muy común en el desarrollo de nuevos productos y tecnologías) o por que el acceso al conocimiento tiene un costo que no puede ser pagado por cualquier miembro de la sociedad. En Colombia el acceso a la educación técnica, tecnológica y superior es limitado, se tiene una cobertura del 37%⁵. La escasez de cupos en los centros de formación públicos y los altos costos de las matrículas en las universidades privadas (con niveles de calidad similares) han convertido al conocimiento en un bien tipo club (en Colombia);

⁵Según el observatorio de la universidad colombiana, esta cifra aumentó en los últimos años gracias a las matrículas del SENA

		Rivalidad	
		Rival	No rival
Exclusión	Exc.	Privado	Club
	No exc.	Común	Público

Figura 1.1.: Clasificación general de los bienes. fuente: [1]

afectando de forma considerable la cantidad de personal con las capacidades necesarias para absorber los conocimientos asociados a nuevas tecnologías; adicionalmente, según el ministerio de educación nacional, el 64 % de los matriculados cursan programas universitarios y solo el 24 % están en programas relacionados con ingeniería.

Otro tipo de fuentes de capacitación la suministra el Centro de Investigación y Desarrollo tecnológico de la industria Electrónica e Informática (CIDEI), a través de tres cursos: Diseño digital empleando dispositivos lógicos programables; diseño de software embebido para microcontroladores ARM y diseño de circuitos impresos con normas internacionales IPC; sin embargo, el elevado costo de estos cursos genera una barrera en la obtención de este conocimiento por parte de los interesados.

Basados en el estado de la industria electrónica nacional y de la capacidad del país para la formación de personal calificado, se puede afirmar que el principal problema que presenta la industria electrónica nacional es la falta de conocimientos sobre procesos de diseño y fabricación, debido en parte a la fuerte exclusión que se tiene al acceso de la información relacionada con estos procesos, lo que se traduce en la incapacidad de producción local de productos que cumplan con los estándares internacionales. Para contribuir en la solución de este problema, este trabajo proporciona un recurso público basado en el conocimiento necesario para concebir, diseñar, implementar y operar sistemas digitales que utilicen tecnología de punta y método logías de diseño modernas proporcionando:

- Diseños de referencia de sistemas digitales funcionales que puedan ser utilizados en la generación de nuevos productos.
- Repositorios donde se puedan descargar los archivos necesarios para entender, reproducir y modificar estos diseños de referencia.
- Una lista de discusión que proporcione soporte sobre el proceso de diseño.
- Un programa académico que actualice los contenidos y la metodología de las asignaturas del área de electrónica digital.
- Documentación que puede ser utilizada por las empresas para capacitar a su personal.

Este recurso estará representado por: la información necesaria para entender, reproducir y modificar plataformas de desarrollo que pueden ser utilizadas para la creación de nuevos productos

que den solución a problemas locales; la documentación necesaria para entender el proceso de diseño de un sistema digital complejo; una comunidad que crea/utiliza/mantiene el recurso; medios de comunicación y de almacenamiento para los proyectos existentes y facilidades que permitan la creación de nuevos proyectos. Cada uno de estos proyectos representa la aplicación de técnicas de diseño y de fabricación modernas de sistemas embebidos que pueden ser utilizados por los industriales y por la academia como herramientas de capacitación; su carácter público permite el acceso y uso por parte de cualquier interesado; proporcionando un canal de acceso al conocimiento sin restricciones.

Tomando como fuente de inspiración el movimiento de software libre y de código abierto (FOSS), se definió un concepto similar pero aplicado al desarrollo hardware, permitiendo la generación, distribución, estudio y modificación de plataformas físicas (placas de circuito impreso y dispositivos funcionales). Este nuevo concepto recibe el nombre de *hardware copyleft*, su principal objetivo es servir como canal para la transferencia del conocimiento necesario para diseñar y producir sistemas digitales a todo sector de la sociedad que esté interesado. Con esto, se pretende generar en el país los conocimientos necesarios para aumentar la oferta de productos tecnológicos producidos localmente, lo que se traducirá en un aumento de la oferta local de bienes y servicios relacionados con la manufactura de sistemas digitales, generando empleo para personas con diferentes niveles de formación.

1.3. Propuesta de Metodología para la Transferencia Tecnológica y de Conocimientos

Este trabajo proporciona una metodología novedosa para la transferencia de los resultados de la investigación a la academia y a la industria; con su aplicación a nivel nacional se pretende contribuir a la situación de la industria electrónica nacional. La figura 1.2 ilustra los pasos de esta metodología.

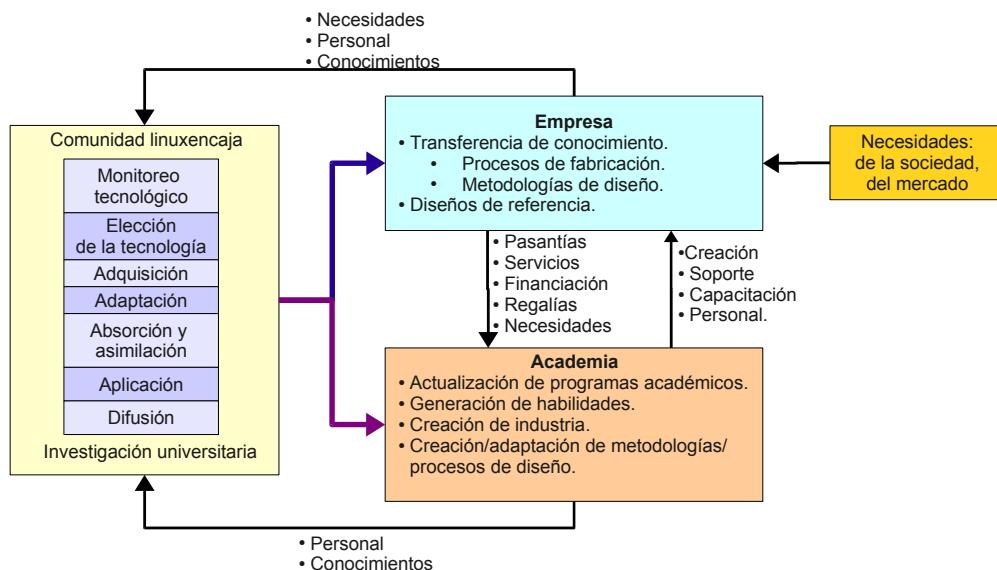


Figura 1.2.: Modelo de transferencia tecnológica.

En el capítulo 2 se realizará una descripción de los canales de transferencia tecnológica

tradicionales, evaluando su efectividad en la transferencia de conocimientos hacia la industria electrónica nacional e identificando los obstáculos para que se logre una transferencia exitosa. Tomando como base estudios realizados en el país y en países con problemáticas similares, se enumeraran una serie de recomendaciones dirigidas a la academia, la industria y las entidades gubernamentales que tienen como objetivo permitir que se alcancen los objetivos de la transferencia; de esta lista de recomendaciones se obtuvieron las actividades prioritarias que se realizaron durante el desarrollo de esta investigación. Con estas actividades y con la experiencia de procesos similares en otros países se formuló una metodología que permite la transferencia tecnológica y de conocimientos en el área de diseño de sistemas digitales. Para su validación se ejecutaron todas sus etapas; los resultados de este proceso (ver capítulo 3) y los conocimientos adquiridos fueron aplicados en la academia y a la industria con el fin de determinar la pertinencia, importancia y utilidad de los mismos.

Como se mencionó anteriormente, es importante que la sociedad cuente con las habilidades necesarias para entender, adaptarse a los cambios de la tecnología y generar productos que satisfagan necesidades locales, por este motivo se creó un programa académico que ayuda a la creación de estas habilidades en los estudiantes de las carreras relacionadas con el diseño digital; este programa académico sigue los lineamientos de la iniciativa Concebir, Diseñar, Implementar, Operar (CDIO)⁶ y será explicada en el capítulo 4.

La etapa de *difusión* es una de las más importantes ya que permite que el conocimiento adquirido llegue a todos los sectores de la sociedad interesados en él y ayuda a crear conciencia de su importancia. Utilizando los contactos existentes entre la Universidad Nacional de Colombia (UNAL) y las universidades Distrital Francisco José de Caldas (UDFJC), Industrial de Santander (UIS), de los Andes (ULA), Santo Tomás de Aquino (USTA), Escuela Colombiana de Ingeniería (ECI) y Escuela Naval Almirante Padilla (ENAP); se transmitieron parte de los conocimientos generados en este estudio para que sean parte los programas curriculares de las asignaturas relacionadas con el diseño digital. Adicionalmente, se formuló un proyecto presentado al SENA para capacitar a empresas de base tecnológica en el diseño, implementación y producción de sistemas digitales.

Se creó la empresa de base tecnológica **emQbit** formada por egresados de las carrera de ingeniería electrónica y de sistemas de la Universidad Nacional de Colombia, a la que se le suministró el conocimiento necesario para generar productos que utilizan los conocimientos adquiridos en este trabajo. Esto con el fin de determinar las necesidades de las nuevas empresas y como puede ayudar la universidad a su consolidación; los resultados de esta interacción pueden verse en el capítulo 3.

Para asegurar la eficacia de esta metodología es necesaria la interacción entre la academia, la industria y la sociedad. En este trabajo se muestra como los resultados de investigaciones pueden enriquecer los programas académicos de las instituciones de educación superior y como se puede transmitir a la industria y de esta forma ayudar a dar solución a problemas de la sociedad. Al crear la oferta de personal calificado por parte de la academia se establece un flujo de conocimientos desde esta hacia la industria, ya sea en forma de asesoría o de personal con las habilidades requeridas; a cambio, la academia recibe regalías de proyectos conjuntos y bienes y servicios más económicos.

Al permitir la utilización de los resultados de esta investigación por quien esté interesado y considerar este conocimiento como un bien público se aumentan las posibilidades de difusión. Para proteger la propiedad intelectual de los creadores de este recurso y para asegurar el crecimiento del mismo, se trabajó con el esquema de licencias *Creative Commons CC-BY-SA* [17], el cual permite la distribución, replica y modificación (incluso para fines comerciales) de los productos generados de

⁶Concebir - Diseñar - Implementar - Operar; creada por el Massachusetts Institute of Technology (MIT)

esta investigación sujeto a las condiciones:

- Dar crédito al autor del trabajo original.
- Toda obra derivada de este trabajo debe distribuirse con el mismo esquema de licencias (esquema de licencia viral).

Para ayudar al proceso de difusión, y siguiendo la metodología de los proyectos de software libre, se creó una comunidad conformada por: estudiantes y profesores de las universidades Nacional, Andes, Javeriana, Santo Tomás, Distrital, Escuela Colombiana, Universidad Industrial de Santander (UIS), Nacional del Comahue (Argentina); empleados de las empresas emQbit, e-modust, ingeniería inversa (argentina); profesionales y aficionados de diferentes ciudades del país⁷. La incorporación de centros de formación e industriales a esta comunidad posibilita identificar necesidades de la industria y ofertas de la academia permitiendo la creación de un vínculo entre ambos sectores. Los aficionados, realizan labores fundamentales de vigilancia tecnológica y de mantenimiento y depuración del contenido. La filosofía de esta comunidad, sus recursos, sus miembros y sus reglas se muestran en el capítulo 5.

Adicionalmente, se diseñaron, construyeron, probaron y programaron una serie de plataformas hardware que pueden ser utilizadas como base de nuevos productos comercializables o como herramientas didácticas en la capacitación a nivel académico o industrial. Estas plataformas son distribuidas con los archivos y documentación necesarios para su estudio, reproducción y modificación; esto unido al uso de software libre en todo el proceso de diseño y programación define el concepto *hardware copyleft*. El uso de plataformas *hardware copyleft* permite ahorrar tiempo y dinero en el desarrollo de aplicaciones proporcionando diseños de referencia modificables que pueden ser utilizados como base para futuros desarrollos. La diferencia importante, entre el uso de plataformas *hardware copyleft* y el uso de plataformas de desarrollo disponibles comercialmente es la posibilidad de modificación, lo que permite optimizar los componentes a una aplicación determinada eliminando el costo de los componentes necesarios en la adaptación de estas plataformas genéricas; adicionalmente, se aumenta la oferta de productos generados localmente, lo que puede generar un aumento de la oferta de bienes y servicios en la manufactura de sistemas digitales (como ya ha ocurrido en otros países de la región). Las plataformas suministradas como parte de este trabajo cubren un amplio rango de aplicaciones y pueden ser utilizadas por cualquiera que este interesado para aumentar los conocimientos en el diseño e implementación de sistemas digitales.

Como se demostrará en el capítulo 6 este trabajo proporciona los conocimientos que suministraría una transferencia tecnológica realizada por un país industrializado. Al momento de escribir este trabajo, se está evaluando la posibilidad de realizar una transferencia tecnológica entre el estado Colombiano y dos centros de desarrollo coreanos Electronics and Telecommunications Research Institute (ETRI) y Korea Electronics Technology Institute (KETI) en el área de aplicación de televisión digital; como parte de este proyecto se contempla transferir los conocimientos necesarios para realizar la fabricación de una plataforma de desarrollo que pueda ser utilizada en este tipo de aplicaciones. Se estima que se puede ahorrar por lo menos 800.000 USD, en esta transferencia, al tiempo que se suministran herramientas para la formación de personal calificado.

⁷Esto gracias a la difusión realizada en diferentes eventos como el Campus Party, FLISOL y conferencias sobre los movimientos abiertos

2. Transferencia Tecnológica

2.1. Introducción

La transferencia de tecnología ha introducido técnicas de alta productividad y en muchos casos cambios técnicos en países menos desarrollados que Colombia. La adquisición de tecnología foránea contribuye a mejorar la competitividad en los mercados locales e internacionales en estos países, en los que debe ser considerada como un proceso vital. Este proceso presenta problemas cuando se pierde capacidad de absorción por parte del país receptor y la renuencia del país que transfiere a suministrar el *know-how*. Por lo que es necesario que estos países promuevan sus capacidades tecnológicas con el fin de absorber las tecnologías foráneas de forma eficiente en función de sus necesidades locales y de esta forma generar un rápido proceso de industrialización. La transferencia de tecnología según Van Gigch [18] involucra la adquisición de “actividad inventiva” por parte de usuarios secundarios. Es decir, la transferencia tecnológica no involucra necesariamente maquinaria o dispositivos físicos; el conocimiento puede ser transferido a través de entrenamiento y educación, y puede incluir temas como manejo efectivo de procesos y cambios tecnológicos [19]. No debe confundirse la transferencia tecnológica con la apropiación de tecnología que se define como el proceso de interacción con la tecnología, la modificación de la forma como es usada y el marco social dentro del cual es usada.

La transferencia tecnológica es un proceso dinámico que debe ser re-evaluado periódicamente, requiere una infraestructura adecuada que involucra instituciones, centros de formación vocacional, técnica y administrativa, personal con diferentes especialidades y un entorno cultural adecuado. Es difícil que la tecnología desarrollada en un entorno determinado pueda ser transferida sin realizar modificaciones en la escala de producción y la adopción de productos al mercado local.

2.2. Tecnología

La tecnología es definida como el factor más significativo para mejorar la productividad, calidad y competitividad [20] y puede verse como un proceso de transformación de recursos que tiene como entrada recursos naturales, bienes, o productos semi-manufacturados y como salida se obtienen bienes consumibles de capital y semi-manufacturados. El *Technology Atlas team* identifica cuatro componentes de la tecnología [19] (ver figura 2.1):

- *Techno-ware* Relacionado con objetos: Herramientas, equipos, máquinas, vehículos, facilidades físicas, instrumentos, dispositivos y fábricas
- *Human-ware* Relacionado con personas: Habilidades en conocimiento experimental, sabiduría, creatividad, experiencia y competencia.
- *Info-ware* Relacionado con la información: Incluye todo tipo de documentación y datos acumulados relacionados con especificación de procesos, procedimientos, diseños, teorías, y observaciones.

- *orga-ware* Relacionado con la Organización: Acuerdos y alianzas necesarias para facilitar la integración de los componentes técnico, humano, y de información. Involucra asignación, sistematización, organización y redes de comunicación.

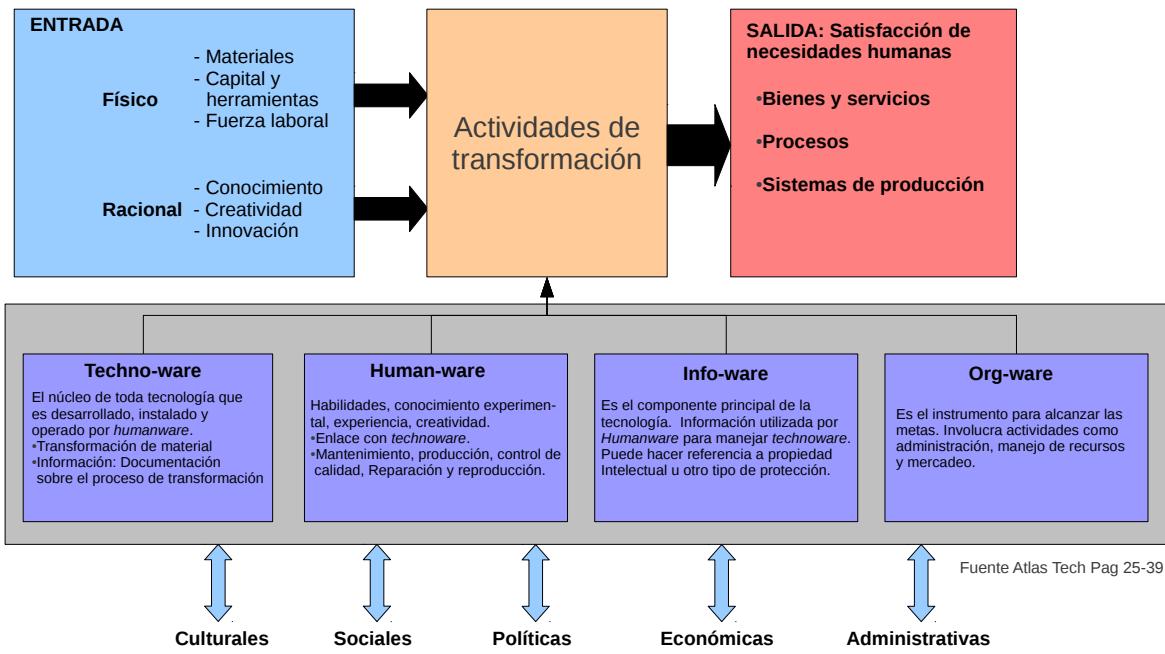


Figura 2.1.: Modelo del proceso de transferencia de tecnología indicando la composición de los canales formales e informales

El uso efectivo de estos cuatro componentes requiere el cumplimiento de ciertas condiciones: el componente técnico requiere de personal con habilidades específicas; para mejorar la eficiencia del sistema, el componente humano necesita de adaptación y motivación; se debe actualizar el sector de la información a medida que la organización cambia para adaptarse a nuevas condiciones o requerimientos. **No es posible realizar operaciones de transformación ante la ausencia de uno de estos cuatro componentes.** La interacción de estos cuatro componentes puedes ser resumida de la siguiente forma:

- *Tecnoware* constituye el núcleo de cualquier tecnología, es decir, una habilidad de transformación, y es desarrollada, instalada y operada por *humanware*.
- *Humanware* o las habilidades individuales representan el elemento clave de cualquier operación de transformación guiada por el *infoware*.
- *Infoware* almacena conocimiento acumulado para ahorrar tiempo en el aprendizaje individual. Es generado y utilizado por *humanware* para los procesos de toma de decisiones y operaciones.
- *Orgaware*, o el marco de trabajo administrativo, adquiere y administra el *tecnoware*, *humanware* e *infoware* con el fin de realizar la operación. *Orgaware* se compone de las actividades de planeación, organización, activación, motivación y control de operaciones.

La tecnología no está asociada a un sistema socio-económico abstracto; se encuentra fuertemente relacionada con un espectro amplio de las necesidades humanas, influenciadas por las condiciones físicas existentes o por factores culturales derivados de las especificidades históricas de diferentes grupos sociales [21].

2.3. Concepto de Transferencia Tecnológica

Odedra [22] define la transferencia tecnológica como el problema de transferencia de conocimiento (o know-how) sobre como funciona un determinado sistema, como operarlo y desarrollar sus aplicaciones, como mantenerlo y si es necesario, como producir sus componentes y montar un sistema similar. La transferencia tecnológica se considera exitosa cuando los receptores de la tecnología asimilan estos conceptos para suplir sus necesidades locales.

Según Jolly [23] la innovación tecnológica es entendida como un nuevo método, medio o capacidad del individuo para realizar una determinada actividad. El resultado de la transferencia tecnológica puede ser la aceptación de una práctica común en otros lugares, o la aplicación de una técnica diseñada para otro uso en la solución de problemas locales; debe distinguirse del proceso general de difusión de tecnología: un movimiento no planeado de artículos sociales y tecnológicos de un lugar a otro sin ningún esfuerzo centrado en la transferencia. La transferencia tecnológica incluye la difusión de conocimiento científico y la preocupación por la transformación del conocimiento en innovaciones útiles. El conocimiento es lo que queda al final de un proceso documentado y difundido de forma apropiada. Para que la transferencia tecnológica sea exitosa es necesario transferir los componentes de la tecnología, es decir: los conocimientos técnicos, las habilidades humanas, la información y la estructura de la organización.

2.3.1. Tipos de Transferencia Tecnológica

La transferencia tecnológica se clasifica en [24] material, de diseño y de conocimientos:

- Transferencia material: Artefactos tecnológicos, materiales, productos finales, componentes y equipos.
- Transferencia de diseño: Diseños, proyectos y know-how para fabricar productos diseñados previamente (los productos son copiados para producirlos localmente).
- Transferencia de capacidades: Proporciona know-how y software no solo para fabricar componentes existentes, sino, innovar y adaptar tecnologías existentes para producir nuevos productos; y transferencia de material, diseño y capacidades
-

La transferencia de material no constituye una transferencia tecnológica real, ya que no genera el conocimiento necesario para transformarlos y generar nuevos productos que cumplan con las necesidades locales. La transferencia de diseños permite adquirir mayor conocimiento sobre la tecnología transferida, sin embargo, es necesario que el país receptor cuente con la plataforma tecnológica adecuada para absorber estos conocimientos, de lo contrario no se generarán nuevos productos y las actividades se limitarán al ensamblaje de productos pre-manufacturados. La transferencia de capacidades es ideal, ya que proporciona las herramientas necesarias para que la

transferencia sea exitosa, está asociada a una transferencia de conocimiento, lo cual es vital para entender plenamente la tecnología, mejorando las habilidades de los profesionales del país receptor.

2.3.2. Canales para la Transferencia de Tecnología

Erdilek y Rapoport [25] clasifican los mecanismos en *formales* e *informales*:

- Formales: Acuerdos de licenciamiento, inversión extranjera, compañías conjuntas, acuerdos de cooperación en investigación y arreglos de producción conjunta
- Informales: no involucran acuerdos entre las partes y son difíciles de detectar y monitorear, por ejemplo: exportación de productos tecnológicos o bienes de capital; ingeniería inversa; intercambio de personal técnico y científico; conferencias de ciencia y tecnología; ferias y exposiciones; educación y entrenamiento realizado por extranjeros; visitas comerciales; literatura abierta (artículos, revistas, libros técnicos) y espionaje industrial.

Adicionalmente, existe una división basada en la naturaleza de la institución que proporciona los recursos para que se realice la transferencia, la institución puede ser de carácter: *abierta* o *privada*:

- Abierta: La tecnología y el conocimiento son considerados bienes públicos, no existen restricciones para acceder a la información necesaria para adquirir, usar y transformar estos conocimientos en productos comerciales, y su éxito radica en obtener la máxima difusión posible para que los usuarios de este conocimiento mejoren el material existente y contribuyan con experiencias personales.
- Privada: La tecnología y el conocimiento se genera para fines privados, la utilización de este conocimiento está sometida a acuerdos comerciales, no es posible entender las bases de la tecnología, por lo que no se pueden generar productos derivados

A continuación se realiza una descripción de los canales más utilizados para la transferencia de tecnología y conocimiento en países en vías de desarrollo ([20] [22] [26] [15]) indicando en cada caso sus ventajas, limitaciones y desventajas.

- **Adquisición de Information Technology (IT)** La adquisición de equipo ha sido uno de los mecanismos de transferencia más importantes para los países en desarrollo; la experiencia de países que lograron un rápido desarrollo económico e industrial muestra que la adquisición de una gran cantidad de tecnología foránea jugó un papel importante en este proceso. Sin embargo, si la compra de equipo no está unida a la transferencia de los conocimientos asociados a dicha tecnología, solo se adquiere el conocimiento necesario para utilizar estas máquinas, y por lo tanto, no se puede fabricar una máquina similar. Sin embargo, este conocimiento sobre el sistema puede ayudar a concienciarse sobre la importancia de la tecnología e impulsar la formación de capital humano.

En el campo de aplicación de la industria electrónica, las grandes multinacionales dominan el mercado de software y hardware y hacen que sea imposible el ingreso de pequeñas compañías locales, lo que se traduce en que el mantenimiento y servicios asociados al hardware, así como los ajustes de software, sean realizados por los proveedores de las multinacionales y en muy pocos casos los usuarios de esta tecnología adquieran habilidades para sostener el equipo. La transferencia se realiza a subsidiarias de las multinacionales, con lo que la transferencia es poca.

- **Educación y Entrenamiento** Educar a las personas a través de cursos y entrenamiento en el país y enviándolas al extranjero para otros estudios es una forma de adquirir know-how sobre nuevas tecnologías, o tecnologías que no se utilizan en el país de origen. Muchas instituciones ofrecen carreras en ingeniería electrónica, ciencias de la computación y afines, algunas de ellas utilizan modelos pedagógicos utilizados en países desarrollados, los que no han sido adaptados plenamente a la infraestructura tecnológica local, y no es raro encontrar estudiantes que no están satisfechos con su profesión al finalizar los cursos[22]. La transferencia tecnológica no ocurre cuando estudiantes formados en el exterior no pueden aplicar sus conocimientos en su país de origen, por lo que es necesario crear políticas que definan que áreas de estudio son prioritarias para el país.

Las multinacionales también ofrecen cursos de capacitación, sin embargo, se limitan al uso de sus productos, creando dependencia hacia sus herramientas. Adicionalmente, existen centros privados de capacitación que ofrecen cursos para el manejo de paquetes y lenguajes de programación; estos centros aprovechan la falta de centros de enseñanza tecnológica y personal calificado para cobrar altas sumas de dinero, lo cual limita el acceso. Programas académicos inapropiados, acceso limitado a libros y computadores, falta de facilidades para capacitación, reduce la efectividad de la educación y capacitación como canal para la transferencia tecnológica.

- **Asistencia Técnica** La ventaja de contratar consultores externos radica en el ahorro de tiempo y dinero, ya que, utilizar personal local implicaría un gran esfuerzo y posiblemente se tendrían que asumir errores costosos en el proceso; sin embargo, no es bueno confiar a consultores externos la responsabilidad de construir habilidades locales, ya que reduce el desarrollo de estas habilidades, especialmente, la del personal encargado de manejar proyectos. En algunas ocasiones los consultores no están familiarizados con las condiciones y requerimientos locales, con lo que diseñan soluciones que no se ajustan a las necesidades, lo que significa que el sistema es sub-utilizado y la transferencia de tecnología es poca. La falta de personal calificado hace que los consultores se encarguen de todas las tareas del proyecto, lo que aumenta su carga de trabajo y disminuye la posibilidad de entrenamiento de personal local[22].

En algunas ocasiones los consultores son representantes de grandes multinacionales y todas sus acciones están dirigidas a aumentar la dependencia con los productos generados por dichas transnacionales y a ignorar de forma sistemática opciones que pueden ayudar a la transferencia de conocimiento, llegando hasta el punto de influir en la formulación de políticas para transferencia tecnológica.

- **Licenciamiento** El licenciamiento es un canal que se utiliza para transferencia de know-how sobre productos o procesos, es aplicado de forma individual o en combinación con otros instrumentos como investigación extranjera, importación de maquinaria o de técnicos. Sin embargo, no es efectivo si no se acompaña de habilidades administrativas y de producción; adicionalmente, es necesario contar con una infraestructura tecnológica adecuada, capacidades locales de fabricación de hardware y software y políticas de gobierno adecuadas [22].

- **Inversión Extranjera Directa** La inversión directa de multinacionales asegura una rápida transferencia de información tecnológica, pero no necesariamente del entendimiento o know-how. Lo que hace que la tecnología transferida a través de este canal sea mínima. Las grandes multinacionales pueden tener cierto control político en los países en vías de desarrollo, hasta

tal punto que son asesores de instituciones encargadas de fijar políticas para la transferencia tecnológica [22].

La efectividad de cada canal depende de la naturaleza de la tecnología que se va a adquirir, el tipo de organización y de las capacidades de absorción del receptor. La tecnología es efectiva únicamente cuando la economía del país es capaz de utilizarla; cuando se transfiere una tecnología se debe contar con la capacidad para adquirirla y se deben generar las actividades necesarias para mejorar la plataforma tecnológica, incluyendo la educación y la capacitación, de tal forma que el país sea capaz de absorberla y generar nuevos productos que satisfagan necesidades locales.

La aproximación de este trabajo se centra en la educación y entrenamiento como canal para la transferencia tecnológica; una de las razones para esta elección es la dificultad que se presenta en Colombia para acceder a información especializada. Suministrando una serie de conocimientos que pueden usarse como punto de partida para el desarrollo de aplicaciones comerciales o como material para la capacitación de personal especializado proporcionando un canal de comunicación entre los diseñadores locales y la industria manufacturera; adicionalmente, se proporcionará una metodología de diseño basada en el uso de herramientas abiertas lo cual reduce de forma considerable la inversión de capital en herramientas de desarrollo.

2.4. Obstáculos para una Transferencia Exitosa

La cantidad de conocimiento y tecnología transferida se ve afectada por políticas gubernamentales, la situación económica, **facilidades de educación y capacitación**, personal calificado, aspectos organizacionales y sociales, proveedores de tecnología e infraestructura tecnológica. El gobierno juega un papel importante en el proceso de transferencia tecnológica ya que puede invertir en la infraestructura para impulsar una determinada tecnología o des-estimular su uso. Estas políticas son dependientes de la situación económica del país y del entendimiento de la importancia de la transferencia por parte de sus dirigentes. Las personas son las que finalmente absorben el know-how tecnológico, si no existe el suficiente personal disponible y dispuesto, el proceso de transferencia se detendrá.

La administración a nivel de organización juega el papel más importante en el proceso de transferencia tecnológica. La resistencia o el desconocimiento a la tecnología, la adquisición de tecnología por motivos particulares que no contemplan la implementación y la capacitación. Por esta razón, es necesario que los encargados de tomar las decisiones y trazar políticas, conozcan la tecnología, o que estén conscientes de su importancia. La transferencia tecnológica debe ser un proceso de dos vías, por lo que es indispensable tener habilidades adecuadas en investigación, capacidades organizacionales y de ingeniería para que estos conocimientos sean asimilados y utilizados en la solución de problemas locales. Es necesario que la adquisición de tecnología obedezca a un plan y que esta tecnología supla una necesidad real, de lo contrario los equipos adquiridos y la capacitación recibida no serán utilizados. Por otro lado, la tecnología adquirida que no es asimilada y transformada en herramienta para la solución de problemas locales aumenta el grado de dependencia, lo que representa justamente lo contrario a lo que se debe buscar en una actividad de transferencia tecnológica.

Los procesos de transferencia tecnológica son influenciados de forma directa o indirecta por las infraestructuras organizacionales y tecnológicas de los países, los cuales, deben exceder sus capacidades para absorber la tecnología transferida. Esta transferencia es efectiva solo si la economía del receptor es capaz de utilizarla. Si un país cuenta con los recursos económicos necesarios para

adquirir la tecnología, debe mejorar la infraestructura para soportarla, incluyendo la educación y las facilidades de entrenamiento, así como los enlaces de telecomunicaciones [15].

La falta de facilidades de educación y capacitación afecta la transferencia del know-how, obstaculizando el desarrollo de habilidades a través del proceso de aprendizaje. La carencia de estas facilidades limita la difusión del conocimiento; la pérdida de estas habilidades se puede originar porque la transferencia no se realizó o porque la infraestructura no lo permite.

2.5. Recomendaciones para Obtener una Transferencia de Tecnología Exitosa

Estudios consultados [15] [16] [13] [14] [12] [9] [27] coinciden en que para que se presente una transferencia tecnológica exitosa, es decir, para que los elementos técnicos, habilidades humanas, la documentación y la organización asociadas a una determinada tecnología, puedan ser asimilados por personal calificado (disponible y dispuesto) para posteriormente transformar estos conocimientos en la creación de nuevos productos o servicios que suplan necesidades locales es necesario:

Fomentar la creación de empresas de base tecnológica El gobierno debe crear facilidades y créditos para que empresas tecnológicas con la capacidad de innovación puedan realizar su actividad comercial (productos o servicios) y de esta forma crear nuevos empleos, y aumentar la demanda de servicios tecnológicos. Así mismo, las universidades deben fomentar la creación de empresas que comercialicen productos derivados de sus actividades de investigación y acompañarlas en el proceso de consolidación.

Promoción de la transferencia tecnológica El gobierno y las Universidades deben centrar sus esfuerzos en identificar las necesidades de la sociedad y cambiar sus prioridades para darles solución, las universidades deben realizar proyectos de aplicación que puedan ser utilizados por el sector productivo a corto o mediano plazo; crear políticas que permitan hacer llegar el conocimiento generado a los diferentes sectores de la sociedad. Las políticas de gobierno deben desalentar la compra de equipo que solo transfiera conocimiento sobre su operación y no permita la creación de nuevos conocimientos a partir de ellos; así mismo, debe formular políticas que protejan las empresas locales de base tecnológica des-estimulando el ingreso de productos similares provenientes del exterior, premiando a las industrias nacionales que realicen productos innovadores ya sea con beneficios tributarios temporales o con la adjudicación de créditos condonables destinados al desarrollo de nuevos productos. Universidad, gobierno e industria deben trazar políticas que definan las áreas en las que se deben formar los profesionales en el exterior, las cuales deben estar en sintonía con el estado de la plataforma tecnológica, el sector productivo y el entorno social del país; estas políticas deben cambiar a medida que se mejora la plataforma tecnológica local y se presentan cambios en el entorno mundial. Se debe trabajar en la creación de una cultura de la transferencia tecnológica, resaltando su importancia para el desarrollo del país.

Promover la excelencia académica Debe existir una evaluación continua de los planes académicos para que se adapten a las necesidades del sistema productivo local, proporcionando a sus profesionales las habilidades requeridas por la industria, en especial las requeridas para crear líderes emprendedores que puedan crear nuevas empresas y que sean conscientes de la importancia del aprendizaje continuo. Por otro lado, es necesaria la creación de maestrías y doctorados que sigan políticas nacionales encaminadas al desarrollo económico, orientados a conectar la investigación con el sector productivo local y crear mecanismos de medición que permitan comparar y clasificar las instituciones académicas según las competencias de las habilidades (liderazgo y emprendimiento)

de sus egresados y de esta forma determinar que instituciones son merecedoras de créditos, becas, o financiación para desarrollar actividades. Por último, y no menos importante, difundir habilidades generales para el uso de nuevas tecnologías a los diferentes sectores de la población.

Los centros de educación de diferentes niveles deben trabajar de forma conjunta para definir los objetivos y habilidades que requiere el sector productivo a nivel de formación tecnológica y profesional, con el fin de delimitar sus funciones para que no interfieran en el mercado laboral. Por otro lado, es necesario unificar contenidos de carreras similares y crear programas donde participen diferentes centros de formación con el fin de aprovechar los escasos recursos suministrados por el estado; creando canales de comunicación que permitan compartir resultados de investigaciones para evitar repetirlos.

Promover la relación universidad empresa El sector productivo debe invertir en las actividades de transferencia tecnológica e investigación y desarrollo, ya que es uno de los directamente beneficiado con ellas. El gobierno debe desalentar las prácticas comerciales que no generan actividades de I+D, en especial las que solo comercializan productos manufacturados en países asiáticos ya que esto hace que la industria no vea la necesidad de crear productos propios y por lo tanto no se invierte en investigación y desarrollo, ni se contrata personal o bienes y servicios especializados. La academia debe proporcionar a la industria herramientas y profesionales que le permitan competir con productos provenientes del extranjero. Se debe crear conciencia en la industria de las ventajas de tener productos diseñados localmente, resaltando los servicios adicionales que pueden generarse al personalizar estos productos y proporcionar servicios derivados de su uso. Adicionalmente, se deben crear espacios donde los empresarios participen en los procesos de toma de decisiones y creación de políticas gubernamentales sobre educación e investigación y desarrollo. Para esto, es vital determinar que actividades económicas contribuyen al desarrollo tecnológico, cuales son generadoras de conocimientos y de esta forma incentivar su práctica. Por otra parte, las universidades deben continuar con sus labores de investigación en temas de actualidad y aumentar la visibilidad de la academia colombiana en el entorno científico mundial; así mismo, deben trabajar en problemas del entorno local, que aunque no tienen mucho reconocimiento a nivel internacional si refleja un grado de compromiso con el entorno social en donde ellas operan.

Alianzas para obtener y compartir recursos Como se mencionó anteriormente Colombia es el país de Sur-América que menos invierte en investigación y desarrollo, por esta razón es necesario crear alianzas estratégicas para compartir los escasos recursos disponibles. Esto, debido a que las comunidades científicas del país no están acostumbradas a unir esfuerzos, recursos y proyectos en programas colectivos; sus intereses particulares prevalecen sobre el interés de la comunidad [27]. Lo anterior, unido a la estructura de gobierno de los organismos gubernamentales que hacen ciencia y tecnología donde los investigadores son enfrentados entre si por los escasos recursos del estado, promueven estrategias individuales para conseguir dichos recursos impidiendo la cooperación.

En la Figura 2.2 se hace un resumen de las recomendaciones formuladas anteriormente para lograr una transferencia tecnológica exitosa y como estas están relacionadas con actividades requeridas para el mejoramiento de la plataforma tecnológica y la creación de una cultura de transferencia de tecnología en el área de diseño de sistemas embebidos. Área en la que el país puede competir a corto plazo con productos provenientes de países industrializados.

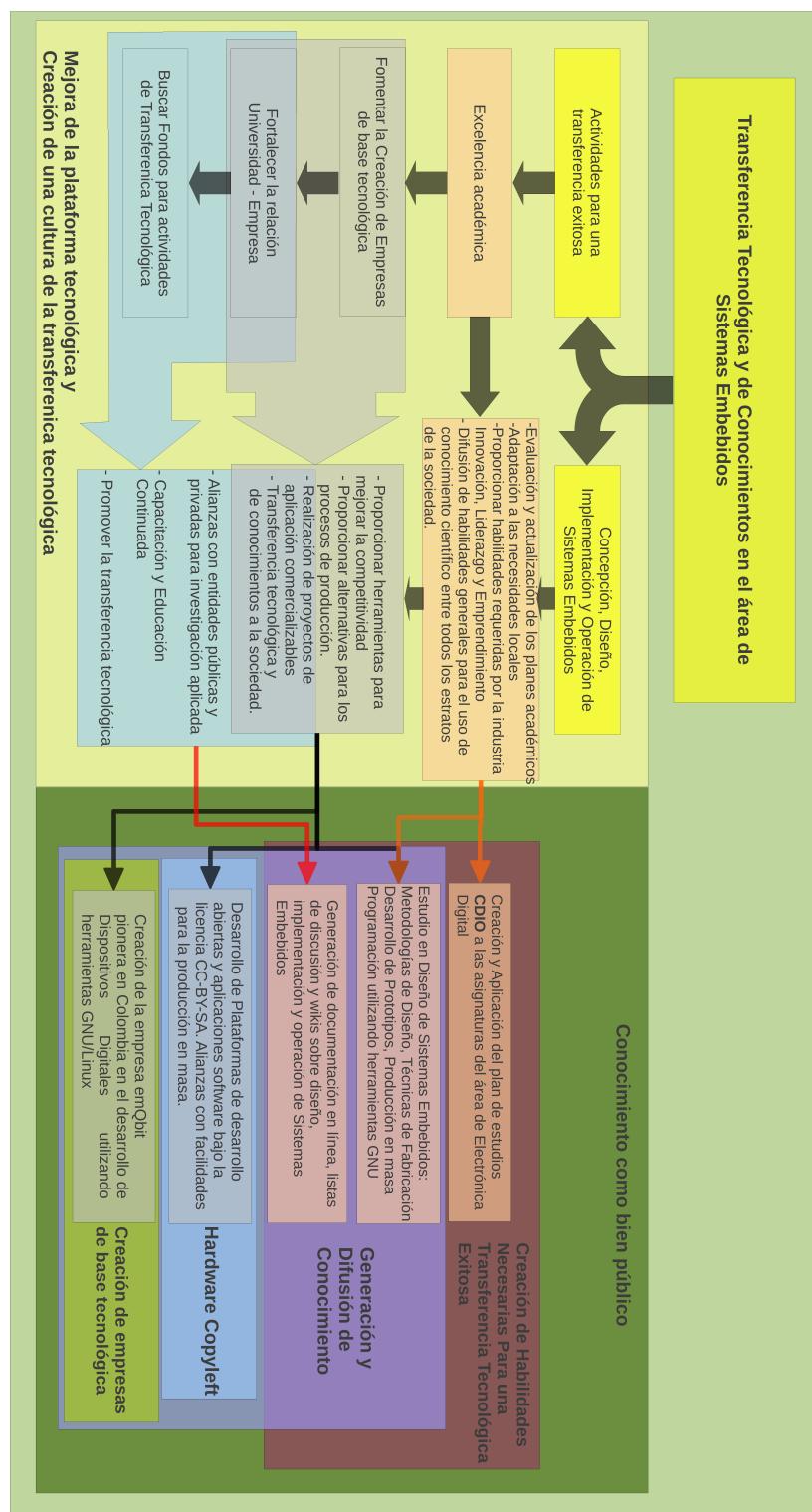


Figura 2.2.: Resumen de las recomendaciones para una transferencia tecnológica exitosas

2.6. Metodología para la Transferencia Tecnológica y de Conocimientos en el Diseño de Sistemas Embebidos

Debido a que esta investigación no cuenta con la participación de sectores gubernamentales generadores de políticas ni con empresas privadas que financien actividades de transferencia con grandes compañías en el exterior; la metodología que se presenta aquí se clasifica como un mecanismo informal realizado por una entidad pública.

2.6.1. Etapas de la Metodología

La metodología utilizada en este trabajo se basa en la formulada por Cohen [20], la cual ha sido utilizada en trabajos recientes en diferentes partes del mundo [28] [29] [30] [31]. En la figura 2.3 se muestran las etapas que componen esta metodología; en las cinco primeras se realizan acciones encaminadas a entender, asimilar, aplicar y desarrollar la tecnología y en la última se realizan actividades de difusión que permitan el acceso de los conocimientos generados a quien esté interesado.

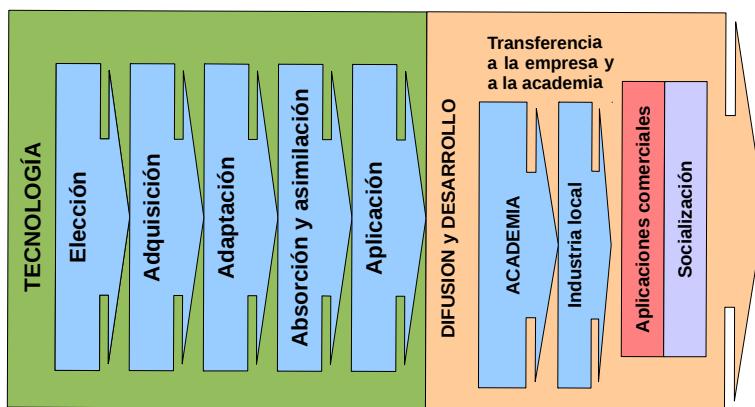


Figura 2.3.: Etapas de la metodología propuesta para la transferencia tecnológica y de conocimientos en el área de diseño de sistemas embebidos

A continuación se describirán brevemente las etapas de la metodología; en el capítulo 3 se realiza una descripción detallada de las actividades realizadas en cada una de ellas al ser aplicada en el diseño de sistemas digitales. Las primeras etapas elección y adquisición están relacionadas con la elección del contenido a ser transferido, mientras que la adaptación, absorción, asimilación y aplicación están relacionadas con la capacidad de absorción y aplicación del conocimiento; la difusión es el proceso por el cual los resultados son comunicados a través de ciertos canales a la sociedad.

Elección

En la primera etapa se evalúa el estado de la plataforma tecnológica existente para identificar facilidades y necesidades; con esta información se busca una tecnología que pueda ser implementada con el estado actual de la plataforma tecnológica; se identificarán los niveles de complejidad de dicha tecnología para determinar una alternativa que pueda implementarse y de resultados a mediano y corto plazo con baja inversión de capital.

Adquisición

En esta etapa se adquieren equipos que utilicen la tecnología que se desea transferir; gracias a que el campo de aplicación de esta tecnología involucra una gran parte de las actividades humanas, es fácil obtener (comprar) dispositivos comerciales que la utilizan, por lo que no es necesario realizar ningún tipo de acuerdo con proveedores extranjeros. Adicionalmente, se realizará la adquisición de plataformas adecuadas de desarrollo hardware y software y se identificarán las herramientas de desarrollo que permitan su estudio y desarrollo de nuevas aplicaciones.

Adaptación

La adaptación se presenta cuando la sociedad encuentra posible y deseable realizar cambios para involucrar usos particulares de la tecnología; es un proceso dinámico que está formado por [28]: una etapa inicial donde se proporcionan procedimientos adecuados para la transferencia física; seguida de una fase donde se proporciona conocimiento, habilidades, datos, y las herramientas necesarias para la evaluación; y una etapa de planeación, diseño e implementación del sistema de transferencia, planes y estrategias.

Para determinar si es posible y conveniente el uso de esta tecnología en la industria y la academia nacionales, se realizaron una serie de actividades encaminadas a adquirir conocimiento y habilidades que permitieron operar y programar los dispositivos digitales adquiridos previamente. Para esto se diseñó una metodología que permite el estudio gradual de diferentes tópicos asociados a esta tecnología (arquitectura, metodologías de diseño, programación básica, generación de aplicaciones, producción); la primera etapa consiste en la **adquisición** de un dispositivo comercial que permita realizar el estudio deseado; a continuación, con el uso de ingeniería inversa se identifica su arquitectura, las herramientas y vías para cambiar su funcionamiento, también se desarrollaron métodos para generación de aplicaciones similares a la original (**adaptación**). El siguiente paso consiste en la adaptación de técnicas de fabricación que permitan el diseño y construcción local de plataformas que tengan la misma funcionalidad y permitan su estudio, fácil adaptación y desarrollo de aplicaciones; en este punto, cuando se reunieron los conocimientos necesarios para desarrollar aplicaciones propias, se transmitieron a la academia y a la industria para que hagan parte de sus procesos de enseñanza y de diseño y producción respectivamente; finalmente se realizó la documentación del proceso de diseño de las plataformas de desarrollo y de las aplicaciones; toda esta información es distribuida bajo el esquema de licencias *Creative Commons (CC) BY-SA: atribución (BY)*, se permite la distribución dando crédito al autor; *compartir de la misma forma (SA)*, exige que todo trabajo derivado del uso de proyectos que con este esquema de licencias deben tener la misma licencia de los trabajos originales (esquema de licencias viral).

Absorción y asimilación

Los procesos de absorción y asimilación describen la capacidad del receptor para utilizar la tecnología y permiten identificar la efectividad y el éxito de la transferencia; la absorción es la capacidad del receptor de absorber tecnología de un sector y la asimilación es la capacidad de asimilar (analizar, procesar, interpretar y entender) y utilizar la tecnología absorbida en otro sector [28]. Es importante generar dos tipos de habilidades para soportar la tecnología: Técnicas: hardware, sistemas operativos, redes, tecnologías de la comunicación, aplicaciones software; Humanas: Habilidades y conocimientos necesarios para desarrollar, mantener, y manipular; habilidad para adaptar tecnologías al entorno local y futuro desarrollo. Es crucial contar con mecanismos de aprendizaje que permitan

adquirir los conocimientos necesarios para operar y cambiar la nueva tecnología; este aprendizaje puede adquirirse por entrenamiento, seleccionando y contratando personal calificado o buscando posibles soluciones en recursos publicados. En este estudio se creó un banco de proyectos que pueden ser utilizados como base de futuros desarrollos y se diseñaron cursos para la enseñanza de metodologías de diseño y procesos de fabricación que usan la tecnología en estudio. Para la transferencia en diseño de sistemas embebidos se desarrollaron o adaptaron metodologías de diseño y procesos de fabricación que culminaron en el diseño e implementación de productos tecnológicos propios y en el desarrollo de proyectos académicos.

Aplicación

En esta etapa se incorporará y aplicará el conocimiento adquirido y asimilado en las etapas anteriores a la industria y a la academia; se realizó una prueba piloto con la participación de la Universidad Nacional de Colombia y la empresa colombiana radicada en Bogotá emQbit ltda.

La línea de electrónica digital del Departamento de Ingeniería Eléctrica y Electrónica de la Universidad Nacional de Colombia está compuesta por cuatro asignaturas: Electrónica digital I, II, Sistemas Embebidos y Técnicas de integración; durante un período de dos años se incorporaron metodologías de diseño y de fabricación en estas asignaturas; los cambios se hicieron de forma gradual iniciando con la asignatura electrónica digital I, continuando con electrónica digital II, hasta llegar a sistemas embebidos. En el capítulo 4 se explica detalladamente el contenido de estas asignaturas y su adaptación a la iniciativa CDIO.

En el campo comercial, se trabajó con la empresa de base tecnológica emQbit para evaluar el impacto del uso de esta tecnología en la industria Colombiana; adicionalmente, emQbit proporcionó información sobre el estado de la industria electrónica en el país ayudando a detectar los obstáculos que enfrentan nuevas empresas en su ejercicio de suministrar soluciones a problemas locales, en el capítulo 3, se realiza un resumen de las actividades realizadas con esta empresa y se enumeran los obstáculos detectados durante su operación.

Difusión y Desarrollo

Esta etapa (en ejecución actualmente) busca involucrar a la mayor cantidad de centros educativos para diferentes niveles de formación (educación básica, media, técnica y profesional); empresas de base tecnológica e institutos gubernamentales en actividades que ayuden a concienciar a la comunidad académica de la importancia del uso de esta tecnología como parte de sus procesos académicos.

La tarea más importante en la fase de difusión, es la creación de una comunidad que utilice los conocimientos generados en todo el proceso, proporcione nuevo conocimiento que haga parte de este bien público y depure las herramientas y el contenido del mismo; por esto, es importante vincular a personas con diferentes intereses y niveles de formación; la primera comunidad se formó con profesores y estudiantes de las universidades Universidad Nacional de Colombia (UNAL), UIS, Universidad Santo Tomás de Aquino (USTA), Universidad de los Andes (ULA), Universidad Distrital Francisco José de Caldas (UDFJC), y las empresas emQbit y CorreLibre, los cuales han realizado hasta el momento la documentación de proyectos realizados durante los cursos del área de digitales; la publicación de todos los archivos necesarios para estudiar, entender, reproducir y programar 6 plataformas de desarrollo *copyleft hardware*. Para ayudar a la comunicación entre usuarios del recurso y a la difusión de sus proyectos, se creó una lista de correo donde se pueden discutir temas

relacionados y se dispone de un wiki en donde se documenta el proceso de diseño de los proyectos; se configuró un servidor que permite subir archivos, trabajo multi-usuario y control de revisiones, lo que facilita el trabajo en equipo para usuarios que se encuentran separados geográficamente.

2.6.2. Factores Formales e Informales

Un programa de transferencia tecnológica debe incluir mecanismos que unan de forma eficiente la fuente del conocimiento con la utilización del mismo. Estos canales de comunicación son mecanismos de recursos humanos que pueden ser incorporados tanto en la fuente como en el destino [23].

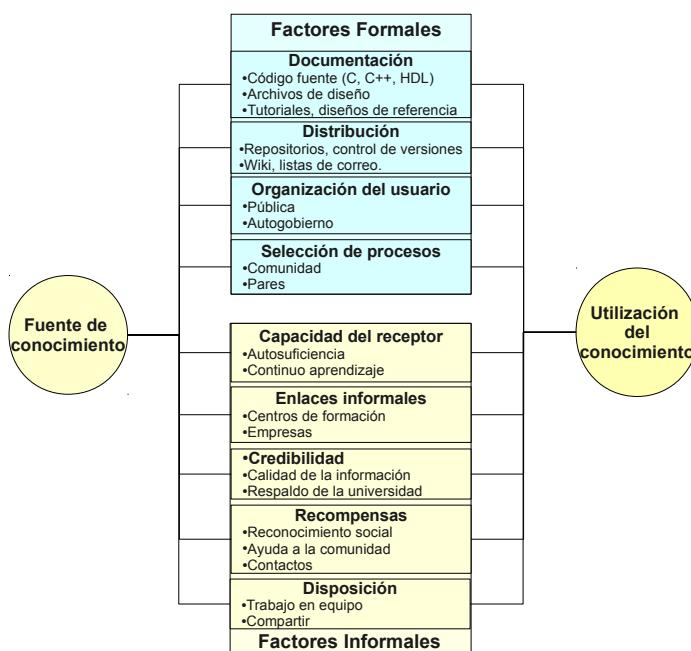


Figura 2.4.: Canales formales e informales del proceso de transferencia tecnológica

. Fuente: [23]

Factores formales

Están formados por procedimientos para difusión, clasificación, almacenamiento y recuperación de conocimientos. En la metodología propuesta estos factores son (ver figura 2.4):

- **Documentación** Es el formato, organización, o presentación de la tecnología que será transferida. El formato y el lenguaje se relacionan directamente con el entendimiento del material por parte del receptor. En este caso se utilizan dos tipos de formatos para la documentación:

- Código fuente. Archivos de texto correspondientes a: archivos de diseño de las placas de circuito impreso (esquemáticos, lista de materiales, layout), código en lenguaje de alto nivel (C, C++, LUA, Python, Perl, Java) para aplicaciones software, y código que utiliza lenguaje de descripción de hardware (VHDL, Verilog) para implementación en dispositivos lógicos programables.

- **Tutoriales.** Información en línea sobre: proceso de diseño, principios de funcionamiento, técnicas de fabricación; suministrados como páginas web, videos y documentos escritos.
- **Distribución** Constituye el canal físico a través del cual fluye la tecnología. Una medida de la efectividad del sistema de información tecnológica es la capacidad de permitir el contacto entre personas con necesidades y con posibles soluciones [32]. Para que el proceso de distribución sea exitoso, debe considerarse el intercambio personal; este canal ayuda a eliminar retardos en la investigación, ya que permite determinar el estado del arte de una determinada actividad o área de trabajo. Esta propuesta cuenta con 4 canales públicos que permiten el flujo de la información y facilitan la comunicación:
 - *Repositorio.* Permite la descarga de los archivos necesarios para estudiar, entender y modificar los conocimientos adquiridos en el presente trabajo.
 - *Sistema de control de versiones.* Permite coordinar el trabajo distribuido entre personas separadas geográficamente.
 - *Lista de correo.* Permite la comunicación entre un grupo de personas.
 - *Wiki.* Permite la realización de documentos útiles en la difusión y en la creación de nuevos proyectos.
- **Organización** El carácter público de este proceso de transferencia hace que los recursos se encuentren disponibles para todo el que esté interesado permitiendo el acceso a la información de forma fácil; evitando crear barreras para su difusión y crecimiento.
- **Selección de Proyectos** Proceso de selección para proyectos de investigación y desarrollo realizado por el proveedor con ayuda del receptor. Aunque esta iniciativa es pública y se proporcionan herramientas públicas para la creación de nuevos proyectos, la comunidad decidirá su calidad y decidirá cuales pueden ser considerados útiles y cuales no. Acá el papel del proveedor de la tecnología es asumido inicialmente por este proyecto, pero con el paso del tiempo esta función será asumida por los miembros de la comunidad; cualquier persona puede iniciar un nuevo proyecto utilizando los recursos disponibles; los miembros de la comunidad deciden contribuir o no a su desarrollo. Es importante que nuevos proyectos o investigaciones en nuevos tópicos comiencen como respuesta a una necesidad de la sociedad.

Factores informales

Los factores informales son de naturaleza sociológica y/o comportamental, y contribuyen fuertemente al éxito de la utilización del conocimiento por una determinada organización. En esta categoría se encuentran la comunicación, creencias y sentimientos sobre la fuente del conocimiento, percepción sobre la organización, supervisores y pares.

- **Capacidad** La capacidad del usuario para utilizar nuevas ideas que cubren un amplio espectro de rasgos que incluyen riesgo, poder, educación, experiencia, edad y confianza en sí mismos. La autosuficiencia es quizás el rasgo más valorado en un receptor de un proceso de transferencia de conocimientos, el programa académico presentado en el capítulo 4 busca crear esta habilidad en los profesionales para que puedan liderar procesos de continuo auto-aprendizaje.
- **Credibilidad** La credibilidad es una evaluación por parte del usuario, de la confiabilidad de la información. Es evaluada analizando la fuente y el canal del mensaje. La opinión cambia

dependiendo de la fuente de la información, es decir, la credibilidad es influenciada por su fuente. Por esta razón, es importante controlar la calidad del material que se suministra como parte del recurso, si este proceso no es muy exigente, usuarios inconformes pueden generar un clima de incertidumbre sobre la utilidad de los conocimientos que se pretenden transferir.

- **Recompensa** Reconocimiento del sistema social del cual hace parte un individuo ante un comportamiento innovador. Ser parte de una comunidad formada por diferentes sectores de la sociedad permite crear vínculos entre sus miembros. Los diferentes aportes que se realizan por parte de los miembros ayudan a definir su perfil y la forma en que la comunidad los define; de aquí pueden nacer vínculos laborales o académicos.
- **Disposición** La habilidad y/o deseo del individuo para aceptar un cambio en la organización de la cual es miembro; así como la capacidad de adopción de nuevas ideas. Aunque una idea haya sido aceptada intelectualmente, toma un período de tiempo antes de ser incorporada en la forma de pensar. Ser consciente de la importancia de una nueva idea no es suficiente para asegurar su uso; debe existir una disposición, un interés, una motivación personal para utilizar un mejor método, proceso o concepto.

Esta propuesta implica que un gran esfuerzo personal en la concepción, diseño, construcción y programación de un dispositivo digital, sea suministrado a la sociedad de forma gratuita; permitiendo que cualquiera pueda beneficiarse de él y a cambio “solo” se recibe el crédito por el trabajo realizado. Al permitir que un trabajo sea utilizado por otras personas y estas personas sepan quien es su creador, puede convertirse en una referencia y es posible obtener beneficios económicos realizando ajustes o nuevas versiones para personas o empresas que no están interesadas en el desarrollo.

3. Implementación de la Metodología de Transferencia Tecnológica Propuesta en el diseño de Sistemas Embebidos

3.1. Introducción

3.1.1. Qué es un Sistema Embebido

Un Sistema Embebido (Sistema Embebido (ES)) es un sistema de propósito específico en el cual, el computador es encapsulado completamente por el dispositivo que el controla. A diferencia de los computadores de propósito general, los sistemas embebidos son diseñados para una aplicación específica, es decir, estos sistemas realizan un grupo de funciones previamente definidas y una vez el sistema es diseñado, no se puede cambiar fácilmente su funcionalidad; debido a su interacción con el entorno deben cumplir restricciones temporales estrictas, el término *sistemas de tiempo real* es utilizado para enfatizar este aspecto; son heterogéneos, es decir, están compuestos por componentes hardware (Programmable Logic Device (PLD)s, Application-specific integrated circuit (ASIC)s) y software (μ -controladores, μ -procesadores, Digital signal processor (DSP)s); tienen grandes requerimientos en términos de confiabilidad.

3.1.2. Arquitectura

En la Figura 3.1 se muestra la arquitectura típica de un Sistema Embebido. La cual integra un componente hardware (HW), implementado ya sea en un (Complex programmable logic device (CPLD), Field Programmable Gate Array (FPGA)) o en un ASIC, (conocido con el nombre de tarea hardware) y un componente software (SW) capaz de ejecutar software; la parte del procesador está dividida en la Central processing unit (CPU) (En algunos casos posee una caché) y las unidades de Memoria.

Al momento de diseñar un sistema embebido se encuentran diferentes opciones de implementación, la más adecuada, resultará de un análisis económico donde se valora el costo de la solución ante el cumplimiento de los requerimientos del sistema; estas opciones son:

- Componente HW y SW integrado en un dispositivo semiconductor (System on Chip - System On a Chip (SoC), Circuito integrado de aplicación específica - ASIC): En la actualidad existen muchas compañías que fabrican procesadores de 32 bits conectados a una gran variedad de periféricos y fabricados en un mismo circuito integrado, lo que simplifica el diseño y reduce costos de materiales. Este tipo de implementación es muy popular en los dispositivos de consumo masivo (reproductores de MP3, consolas de juego, teléfonos celulares, etc.), debido a los grandes niveles de producción (del orden de millones de unidades) resulta más económico contar con un dispositivo que integre el mayor número de funcionalidades que disminuye el costo de componentes y reduce el área de circuito impreso.

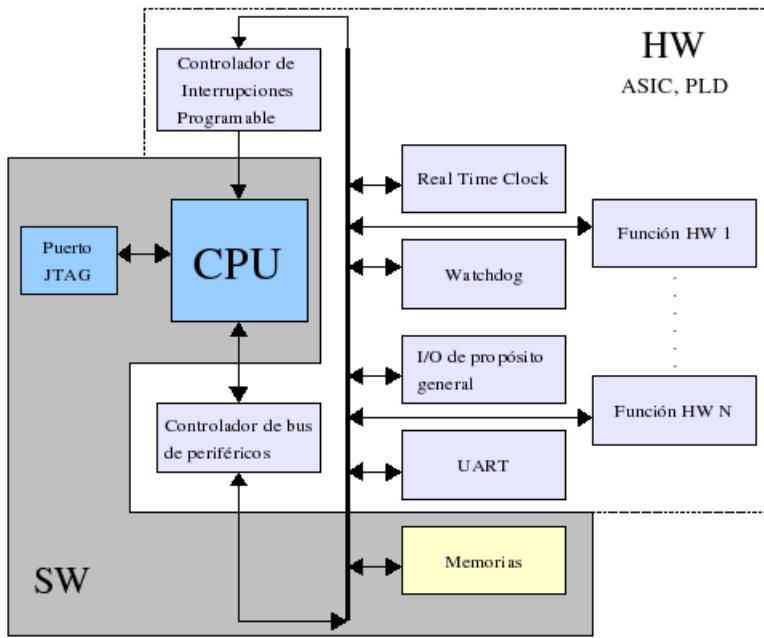


Figura 3.1.: Arquitectura de un Sistema Embebido

- Componente SW en un SoC y componente HW en un dispositivo lógico programable (CPLD, FPGA): Cuando no existe en el mercado un SoC con la cantidad de periféricos requerida para una determinada aplicación, o con una funcionalidad específica, es necesario recurrir a la utilización de dispositivos comerciales que implementen dicha función; en algunas ocasiones el periférico puede realizar funciones poco comunes y no se proporciona comercialmente, la solución es entonces, implementar estas funcionalidades en un dispositivo lógico programable (PLD). También se recomienda la utilización de PLDs en sistemas que requieren la utilización de la misma funcionalidad un gran número de veces (puertos seriales, pines de entrada/salida). Esta decisión está atada al nivel de producción, ya que al incluir un PLD aumenta el costo global del proyecto y el consumo de potencia (el consumo de las FPGAs actuales las hace pocas prácticas para aplicaciones móviles).
- Componente SW y HW en una FPGA: Esta es la opción más flexible, pero la de menor desempeño, ya que al utilizar los recursos lógicos de la FPGA para la implementación del procesador (*softcore*) la longitud y capacitancia asociada a los caminos de interconexión entre los bloques lógicos aumentan el retardo de las señales, lo que disminuye la máxima velocidad de funcionamiento. Los procesadores *softcore* más populares en la actualidad son: Microblaze y Picoblaze de Xilinx, Leon de Gaisler Research y Lattice-Mico32 de Lattice Semiconductors.

3.1.3. Aplicaciones

Los sistemas embebidos se encuentran en casi todas las actividades humanas, a diario se interactúa con ellos, aún sin darse cuenta, ya sea porque son parte de la vida diaria o porque hacen parte de aparatos que se utilizan a diario. La figura 3.2 muestra los campos de aplicación de los

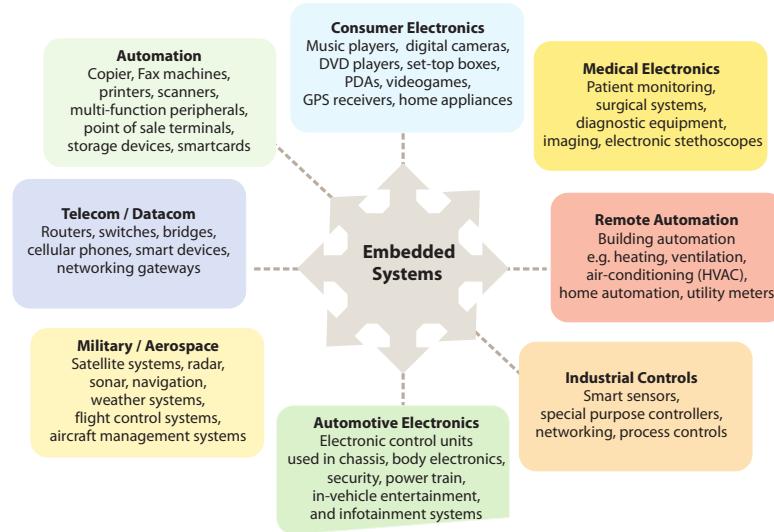


Figura 3.2.: Aplicaciones de los Sistema Embebidos Fuente: TATA Consultancy services

sistemas embebidos.

3.1.4. Metodología de Diseño

El proceso de diseño de un sistema embebido comienza con la *especificación del sistema*, (ver Figura 3.3), en este punto se describe la funcionalidad y se definen las restricciones físicas, eléctricas y económicas del sistema. Esta especificación debe ser muy general y no deben existir dependencias tecnológicas de ningún tipo, se suelen utilizar lenguajes de alto nivel, como Unified Modeling Language (UML), Modeling and Analysis of Real-Time and Embedded Systems (MARTE) C++, System-C, Spec-C. La especificación puede ser verificada a través de una serie de pasos de análisis cuyo objetivo es determinar la validez de los algoritmos seleccionados, por ejemplo, determinar si el algoritmo converge o si sus resultados satisfacen las especificaciones. Desde el punto de vista de la re-utilización, algunas partes del funcionamiento global pueden tomarse de una librería de algoritmos existentes.

Una vez definidas las especificaciones del sistema, se debe realizar un modelamiento que permita extraer de estas su funcionalidad. El modelamiento es crucial en el diseño ya que de él depende el paso exitoso de la especificación a la implementación. Es importante definir que modelo matemático debe soportar el entorno de diseño; cada modelo posee propiedades matemáticas que pueden explotarse de forma eficiente para responder preguntas sobre la funcionalidad del sistema sin llevar a cabo dispendiosas tareas de verificación. Todo modelo obtenido debe ser verificado para comprobar que cumple con las restricciones del sistema.

Una vez se ha obtenido el modelo del sistema se procede a determinar su *arquitectura*, esto es, el número y tipo de componentes y su inter-conexión; este paso no es mas que una exploración del espacio de diseño en búsqueda de soluciones que permitan la implementación de una funcionalidad dada, y puede realizarse con varios criterios en mente: costos, confiabilidad y viabilidad comercial.

Utilizando como base la arquitectura obtenida en el paso anterior las tareas del modelo del sistemas son implementadas en los componentes; esto es, asignación de funciones a los componentes

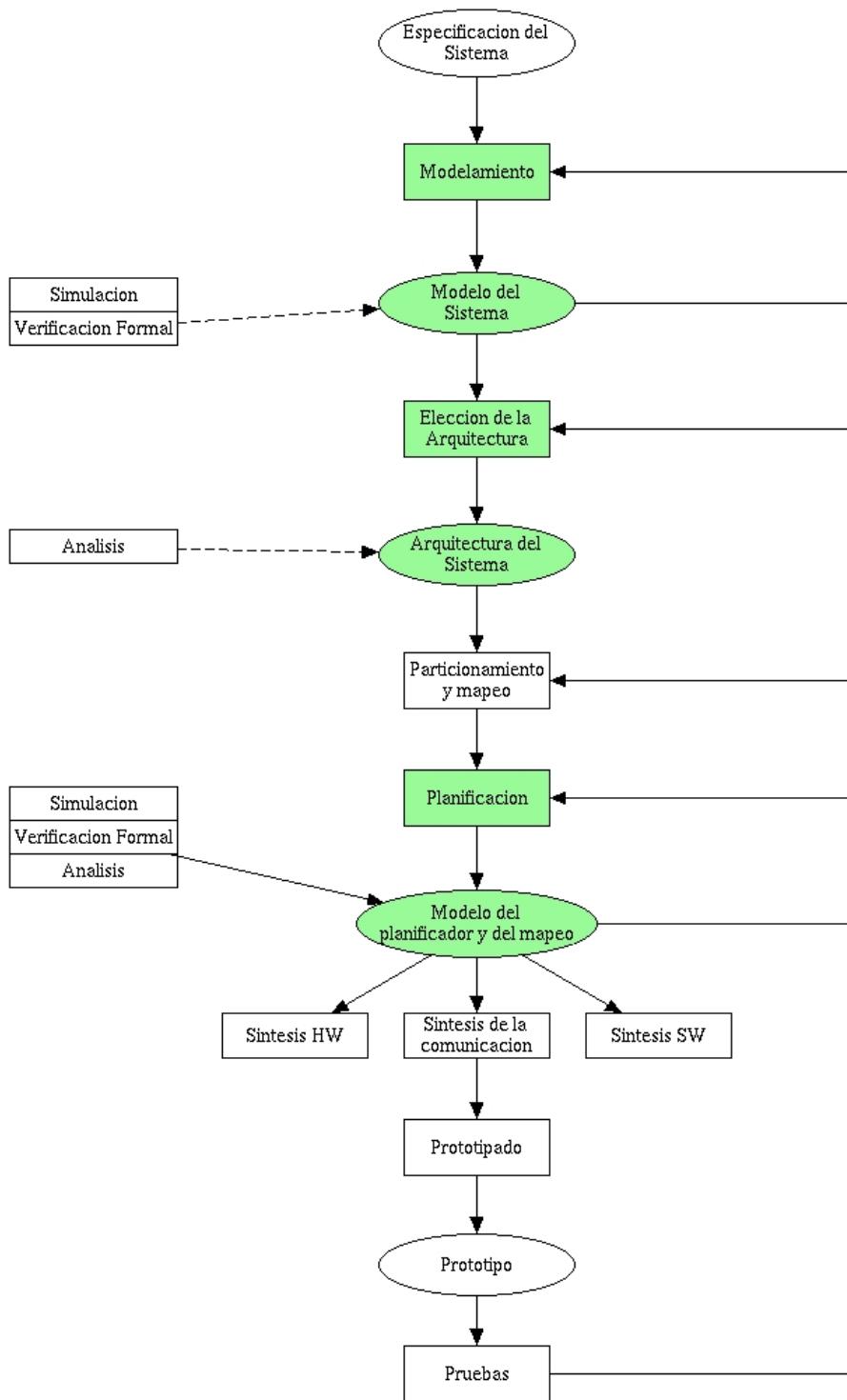


Figura 3.3.: Flujo de diseño de un sistema embebido [2]

de la arquitectura. Existen dos opciones a la hora de implementar las tareas o procesos:

1. Implementación software: La tarea se va a ejecutar en un micro-procesador, micro-controlador o DSP.
2. Implementación hardware: La tarea se va a ejecutar en un sistema digital dedicado ASIC o PLD.

Para cumplir las especificaciones del sistema algunas tareas deben ser implementadas en hardware, esto con el fin de no ocupar al procesador en tareas cíclicas o que requieran mucho poder computacional, un ejemplo típico de estas tareas es la generación de bases de tiempos o la implementación de protocolos de comunicación. La decisión de que tareas se implementan en SW y que tareas se implementan en HW recibe el nombre de *particionamiento*; esta selección depende fuertemente de restricciones económicas y temporales.

Las tareas software deben compartir los recursos que existan en el sistema (procesador y memoria), por lo tanto se deben tomar decisiones sobre el orden de ejecución y la prioridad de estas. Este proceso recibe el nombre de *planificación*. En este punto del diseño el modelo debe incluir información sobre el mapeo, el particionamiento y la planificación del sistema.

Las siguientes fases corresponden a la implementación del modelo, para esto las tareas hardware deben ser llevadas al dispositivo elegido (ASIC, FPGA, micro-controlador, micro-procesador, DSP) y se debe obtener el *archivo binario* con las instrucciones que implementan la funcionalidad de las tareas software y un *archivo de configuración* para implementarla funcionalidad de las tareas hardware en el PLD, este proceso recibe el nombre de *síntesis HW y SW* respectivamente, así mismo se deben sintetizar los mecanismos de comunicación entre las tareas hardware y software.

El proceso de prototipado consiste en la realización física del sistema, finalmente el sistema físico debe someterse a pruebas para verificar que cumple con las especificaciones iniciales.

Como puede verse en el flujo de diseño existen realimentaciones, estas permiten depurar el resultado de pasos anteriores en el caso de no cumplirse con las especificaciones iniciales.

3.2. Implementación de la Metodología Propuesta para la Transferencia Tecnológica en Diseño de Sistemas Embebidos

3.2.1. Selección de la Tecnología

El Plan Estratégico del Programa Nacional de Desarrollo Tecnológico Industrial y Calidad 2005 - 2015 [33] identifica 5 *tecnologías* con mucho potencial en el área de la electrónica, estas son: microelectrónica, diseño y construcción de circuitos integrados, diseño digital con FPGAs, Printed Circuit Board (PCB) multi-capa y dispositivos de montaje superficial. Aunque todos ellos no representan líneas de estudio, todas hacen parte del mismo tema: diseño y fabricación de sistemas digitales. Se nota cierto desconocimiento en el tema ya que separan el área de la microelectrónica y el diseño de circuitos integrados (la microelectrónica estudia el diseño y fabricación de ASICs); las otras tecnologías en realidad no lo son, el diseño de PCBs, uso de dispositivos de montaje superficial y uso de FPGAs, no representan líneas de investigación, sino herramientas útiles para la implementación de sistemas digitales. Esto demuestra la desconexión que existe entre los entes generadores de políticas y la academia.

Niveles de complejidad de la tecnología

Existen varias alternativas para la implementación de un sistema embebido: dispositivos lógicos programables (FPGAs, CPLDs), sistema sobre silicio (SoC), micro-controlador, micro-procesador, SoC + FPGA y ASIC; su utilización está determinada por el cumplimiento de restricciones temporales, funcionales y económicas. La opción tecnológicamente más avanzada es el uso de un Circuito Integrado de Aplicación Específica (ASIC) que implemente las tareas hardware y software en un solo dispositivo semiconductor; sin embargo, se estima que solo a partir de 10 mil unidades es conveniente utilizar un ASIC para reducir los costos de producción; esta es una cantidad muy alta para las pequeñas industrias electrónicas nacionales, y hasta donde llega el conocimiento del autor de este trabajo, por el momento no se conoce el primer circuito integrado fabricado y comercializado por una empresa colombiana o de existir, no es una práctica común. Por otro lado, las herramientas de desarrollo para el diseño de circuitos integrados son muy costosas y el grado de conocimientos de los diseñadores es mayor que en otro tipo de implementación.

Un proyecto vigente promovido por la Unión Europea llamado *Iberchip* empezó desde hace 17 años un proceso de transferencia tecnológica en el diseño de circuitos integrados de aplicación específica (ASICs) hacia los países iberoamericanos; gracias a esta iniciativa se incluyeron asignaturas relacionadas con el diseño de los sistemas embebidos en la mayoría de las carreras de centros de formación consolidados. Sin embargo; esto no ayudó a aumentar la demanda de estos dispositivos por parte de la industria.

En todas las universidades consolidadas del país es común el uso de los lenguajes de descripción de hardware (Hardware Description Language (HDL)) como herramienta para la implementación de aplicaciones; es normal encontrar trabajos de pregrado y posgrado que utilizan familias de FPGA que incluyen procesadores *hardcore* o *softcore* con periféricos dedicados en aplicaciones de procesamiento de imágenes, o de señales. Sin embargo, aún existen muchos centros de formación que continúan utilizando tecnologías y metodologías de diseño obsoletas y pocas industrias locales reportan el uso de estos dispositivos para desarrollar productos comerciales. Por lo tanto, es necesario crear mecanismos que permitan difundir aún más el uso de las FPGAs. Aunque en la actualidad existe una oferta considerable de cursos de extensión para la capacitación en el uso de dispositivos lógicos programables y lenguajes de descripción de hardware, el uso de esta tecnología debe ser justificada por los requerimientos de la aplicación. Debido a que el uso de estas tecnologías requiere un mayor nivel de conocimiento de los sistemas digitales, es necesario realizar rigurosos procesos de verificación para comprobar su correcto funcionamiento; su depuración es un poco tediosa (en comparación con las tareas software). En conclusión, si se desea impulsar el uso de esta tecnología se debe ser muy cuidadoso al momento de elegir las aplicaciones que serán implementadas, esto con el fin de no desalentar a los usuarios de la misma.

Las FPGAs proporcionan una alternativa flexible para prototipado de ASICs, ya que permiten cumplir de forma rápida con los requerimientos del mercado (el proceso de fabricación de un ASIC toma varios meses). Sin embargo, para que un producto sea viable económico es necesaria una solución ASIC de bajo costo; en la actualidad existe la posibilidad de bajar los costos de producción gracias a la demanda de los mismos y a la utilización de una tecnología intermedia llamada *arreglo de compuertas*, la arquitectura de estos dispositivos proporciona una gran cantidad de transistores en arreglos genéricos en un substrato común; y pueden ser utilizados para la implementación de *celdas estándar* o diseños *full custom*; utilizando esta técnica, es posible reducir el número de unidades necesarias para encontrar un punto económico viable de 5000 unidades (fuente Silicon-Pro).

Según Smith [34], la opción más económica para bajos volúmenes de producción son las

FPGAs, a medida que la producción aumenta se produce un punto de quiebra entre las FPGAs y el arreglo de compuertas cerca a las 5000 unidades, y el segundo punto de ruptura se produce alrededor de las 50000 unidades, donde es más rentable la producción de un ASIC basado en celdas estándar. Es muy importante tener en mente estas cifras ya que ellas determinan la tecnología a utilizar. No obstante, vale la pena aclarar que en esta comparación no se tiene en cuenta la utilización de SoC, micro-controladores o micro-procesadores comerciales, por lo que no es necesariamente cierto que a bajos niveles de producción la opción más rentable sea la utilización de FPGAs; adicionalmente, debido a su alto consumo de potencia (del orden de 10 veces mayor que un ASIC) no es posible su utilización en aplicaciones móviles.

En Colombia es muy común el uso de micro-controladores y micro-procesadores de 8 bits para la implementación de sistemas digitales, la mayor parte de los centros de educación del país (de todo nivel) proporcionan cursos de programación y es posible encontrar en el mercado un suministro continuo de ciertas referencias. Gracias a esto, existen numerosos desarrollos basados en ellos; sin embargo, debido a los limitados recursos de estos dispositivos (velocidad, periféricos, herramientas de programación) no es posible utilizarlos para la implementación de aplicaciones con las necesidades actuales como conectividad con diferentes redes, manejo de pantallas de cristal líquido, aplicaciones multimedia, y diversos medios de almacenamiento de información.

Los *System on Chip* (SoC) proporcionan una excelente alternativa para la implementación de aplicaciones modernas; integran un procesador de 32 bits o un DSP que corre a frecuencias que van desde los 75 MHz hasta los 800 MHz y poseen periféricos que permiten controlar directamente una gran cantidad de dispositivos; muchos de ellos están diseñados para aplicaciones que requieren manejo de pantallas táctiles de cristal líquido, conexión a internet, diferentes medios de almacenamiento, reproducción de audio, manejo de sensores de imagen, entre otros; muchas de estas tareas son realizadas por procesadores dedicados diferentes al procesador principal del SoC. Adicionalmente, existe una gran gama de productos ofrecidos por diversos fabricantes como Freescale, NXP, Ingenic, Analog Devices, Altera, Marvell; por otro lado, su uso en aplicaciones de consumo masivo ha reducido el costo de estos dispositivos y es posible comprarlos en cualquier cantidad a precios que oscilan entre 4 y 20 United States dollar (USD).

Diagnóstico de la industria local

La empresa emQbit desarrolló una serie de proyectos y actividades que ayudaron a entender e identificar los siguientes obstáculos para el desarrollo y comercialización de sistemas digitales: Falta de proveedores de bienes y servicios relacionados con la actividad (venta de dispositivos semiconductores, fabricación de placas de circuito impreso, montaje automático de componentes, etc); desconocimiento de la tecnología (alcances y limitaciones) debido al uso de tecnologías y metodologías de diseño obsoletas; competencia con productos asiáticos de muy bajo costo; falta de confianza en los productos nacionales; desconexión de la academia con el sistema productivo; inexistencia de reglamentación de la industria de manufactura electrónica; profesionales con pocos conocimientos en procesos de diseño y fabricación. Estas observaciones coinciden con los resultados de estudios consultados [15] [16] [13] [14] [12] [9].

Diagnóstico de la academia

Basándose en conversaciones con encargados de la línea de electrónica digital de varias universidades, se puede decir que la tendencia moderna en los programas académicos a la utilización

de herramientas de alto nivel para la enseñanza en áreas afines al desarrollo de dispositivos digitales [35] ocasiona que los profesionales no adquieran ciertas habilidades necesarias para completar la cadena concepción - diseño - implementación - operación. En la mayoría de los casos se generan habilidades para la concepción y el diseño a alto nivel y dejan los otros pasos en manos de herramientas especializadas y/o a empresas asiáticas. Esta situación resulta la más atractiva desde el punto de vista económico, ya que no es necesario adquirir maquinaria costosa ni contratar técnicos especializados para operarlas; ; limitando la generación de empleo local a personas con alto nivel de formación [36] generando desempleo en las personas menos capacitadas. Según John Hall presidente y Chief executive officer (CEO) de Linux International “ algunas facultades preparan a la gente en el uso de productos en vez de tecnologías de nivel básico” [35]. Esta situación unida al abandono de la implementación hace que la dependencia con las empresas manufactureras asiáticas aumente cada vez más.

Por otro lado, en muchas instituciones educativas de poca consolidación se utilizan: tecnologías y metodologías de diseño obsoletas (Familias 74XXX o 40XXX, lenguaje ensamblador, mapas de karnaugh); y programas académicos centrados en el análisis y no el diseño, donde el paso final es la simulación. Esto, unido al poco contacto por parte del personal docente con el sector productivo, origina una deficiencia de habilidades necesarias para realizar el proceso completo para el diseño de dispositivos, lo que se traduce en profesionales que no disponen de las herramientas necesarias para resolver los problemas del país y al mismo tiempo competir con los productos asiáticos.

Centros de desarrollo tecnológico

Según el XII Encuentro Colombia compite (patrocinado por el Ministerio de comercio industria y turismo - 2010) el sector electrónico cuenta con solo un centro de desarrollo el CIDEI (Centro de Investigación y Desarrollo de la Industria Electro Electrónica e Informática), el cual ha sido beneficiado con recursos cercanos a los dos mil millones de pesos para la ejecución de proyectos de desarrollo empresarial y con 1200 millones para proyectos de desarrollo tecnológico. Proporciona 2 servicios a la comunidad: diseño, fabricación y montaje de PCBs y capacitación. Los conocimientos generados en estos proyectos no son difundidos a las universidades ni a la sociedad de forma gratuita, lo que limita la difusión de estos conocimientos a quien pueda pagar por ello; por otro lado, la Universidad Nacional proporcionó la capacitación para algunos de estos proyectos, lo que indica que este centro no genera conocimiento propio y los temas de sus investigaciones ya han sido tratados por varias universidades del país.

Discusión

Como se vio anteriormente, la opción más económica para niveles de producción inferiores a 5000 unidades son las FPGAs; sin embargo; esto es cierto únicamente si no existe un dispositivo comercial como SoC, DSP, micro-controlador, o micro-procesador que permita el cumplimiento de las restricciones del sistema. Esto, debido a que el costo de las FPGAs es más elevado y como se dijo anteriormente su consumo de potencia impide su utilización en aplicaciones móviles (por esta razón no es común encontrar FPGAs en dispositivos de consumo masivo). Una revisión de los circuitos integrados utilizados en dispositivos como reproductores MP3, juegos, routers, y algunos teléfonos móviles, revela el uso de SoCs comerciales de diferentes fabricantes. Los fabricantes de SoC se adaptan rápidamente a los requerimientos del mercado y proporcionan dispositivos con los periféricos necesarios para una determinada aplicación; un ejemplo de esto se puede observar en las diferentes

versiones de reproductores MP3 de la compañía *AINOL*, la primera versión evaluada contenía un DSP de Analog Devices, un codec de audio externo y un controlador para el puerto Universal Serial Bus (USB) (Universal Serial Bus); en la siguiente versión evaluada solo se encontró un circuito integrado del fabricante *Ingenic*; una revisión de la hoja de especificaciones de este último circuito integrado indicaba que este ya poseía el codec de audio y el controlador de la interfaz USB como periféricos internos; adicionalmente, su costo es de 3.5 USD en grandes volúmenes. Los grandes fabricantes como Atmel, Freescale, Marvell, NXP, *Ingenic*, etc, de forma dinámica ajustan sus productos a los requerimientos del mercado, agregando el soporte que demandan las aplicaciones. Gracias a esto y a la creciente demanda es posible encontrar SoCs muy baratos, con grandes capacidad de cómputo que pueden ser utilizados en muchas aplicaciones. Por esta razón y por la poca utilización de los SoC en el país, se trabajará en el estudio de técnicas de fabricación y metodologías de diseño basadas en estos dispositivos; esto, sin descuidar el fomento de la utilización de las FPGAs a nivel académico e industrial y el estudio de técnicas de diseño y fabricación de circuitos integrados para estar preparados a un aumento en la producción que justifique su uso.

La hipótesis de este trabajo es que el aumento de diseños locales que utilicen SoC y metodologías de diseño modernas permitirán a la industria electrónica local competir con productos importados; debido a que, el país no cuenta con una oferta considerable de bienes y servicios relacionados con la industria manufacturera de dispositivos digitales (sólo existen 2 empresas), es necesario, utilizar inicialmente los servicios de la industria asiática para construir dispositivos diseñados en el país, que puedan ser configurados para las necesidades exactas del entorno social local y de esta forma aumentar la demanda interna. Finalmente, se debe continuar con el estudio de técnicas de fabricación de circuitos integrados para una futura demanda de la industria local.

3.2.2. Adquisición de Tecnología

Herramientas de desarrollo

Las herramientas de desarrollo son fundamentales en el proceso de diseño, de su estado y capacidades depende el tiempo necesario para completar un determinado diseño; la disponibilidad de aplicaciones y librerías que permitan acelerar el proceso de diseño son puntos claves a la hora de seleccionar el entorno de desarrollo; otro factor importante es su costo, ya que pequeñas y medianas empresas no pueden invertir grandes sumas de dinero en su adquisición; adicionalmente, es crucial contar con una adecuada documentación e información que ayude a resolver problemas que se presenten en el ciclo de diseño. Se pueden clasificar estas herramientas en *propietarias y abiertas*, las primeras requieren la compra de licencias para su uso y es necesario pagar por soporte; las segundas, son distribuidas de forma gratuita y existe una gran cantidad de listas de discusión donde puede encontrarse respuesta a una gran variedad de problemas o pueden ser formuladas nuevas preguntas a un grupo especializado de usuarios.

La utilización de herramientas abiertas reduce de forma considerable la inversión en la plataforma de desarrollo; pero, ¿es posible realizar el flujo completo de concepción, diseño e implementación utilizando software abierto?, ¿el estado de desarrollo de las mismas facilita el diseño?, ¿existen dispositivos comerciales desarrollados con estas herramientas?. Para resolver estas dudas se consultaron varias encuestas realizadas por compañías y sitios especializados para observar la tendencia en utilización de sistemas operativos; los sitios consultados (Venture Development Corp, linuxfordevices) indican que el 27.9 % de los diseñadores utiliza sistemas operativos licenciados comercialmente, el 23.5 % sistemas operativos obtenidos públicamente, 15.9 % desarrollan su propio

sistema operativo, el 12.1 % utiliza sistemas operativos comerciales basados en proyectos abiertos y el 30 % restante no utiliza un sistema operativo (ver figura 3.4; el porcentaje de utilización de sistemas operativos basados en proyectos abiertos es del 35.6 %, lo que supera a los sistemas operativos comerciales; es interesante observar que casi el 70 % de los encuestados utilizan algún tipo de sistema operativo, lo que da un claro indicio de la necesidad de este en el ciclo de diseño).

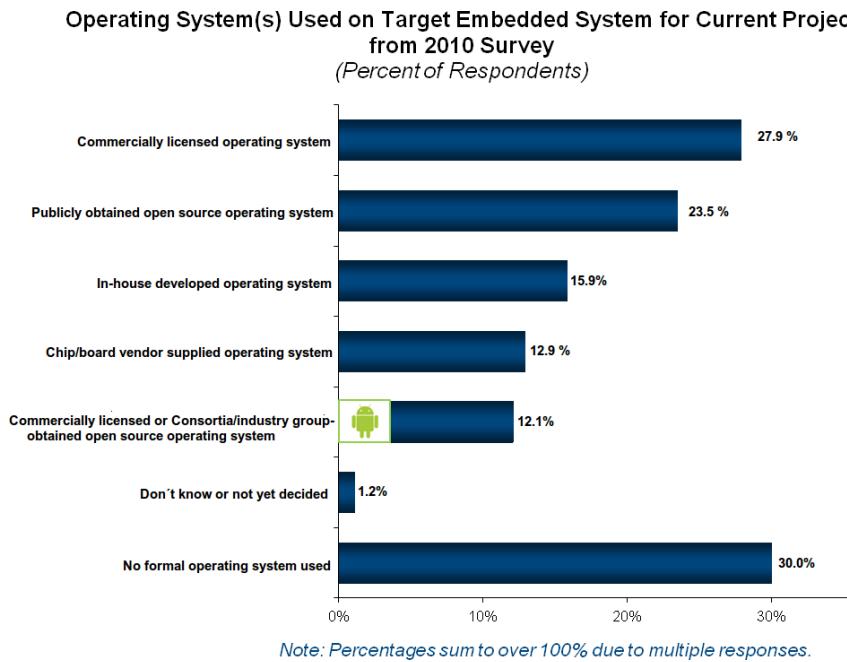


Figura 3.4.: Comparación del uso de sistemas operativos Fuente: Venture Development Corp

De lo anterior se puede afirmar que más de la mitad de los diseñadores que utilizan sistemas operativos para sus aplicaciones eligen proyectos abiertos, lo que indica que estas herramientas tienen el grado de madurez necesaria para su uso en aplicaciones comerciales; por otro lado, una revisión del mercado de los teléfonos móviles realizada por *Admob* indica que android superó a los sistemas operativos de Apple y RIM (utilizado en los blackberry) y se proyecta que en el 2014 igualará a Symbian de Nokia. Android utiliza el kernel de Linux como base de sus aplicaciones y utiliza herramientas abiertas para su desarrollo; otras empresas como Motorola y Nokia utilizan Linux como plataforma de algunas de sus aplicaciones; así mismo, muchos routers basados en procesadores ARM o MIPS; una gran variedad de reproductores multimedia, tablets y mini-laptops; todo esto, unido a la disponibilidad de foros de discusión donde programadores expertos y creadores de una gran variedad de aplicaciones brindan soporte a quien este interesado, hace de las herramientas abiertas y de Linux, una alternativa muy atractiva para desarrollar una metodología de diseño en torno a ella y adaptarla a las necesidades del país.

Linux Foundation publicó un estudio [37] donde calcula que el valor del kernel de Linux es de USD\$1400 millones; y son necesarios USD\$10.800 millones para desarrollar el stack completo de componentes desde cero; por este motivo, el uso de Linux reduce de forma considerable los costos finales del proyecto, *Black Duck Software*¹ posee la más completa base de datos de proyectos

¹<http://www.blackducksoftware.com> Líder mundial en el suministro de productos y servicios que aceleran el desarrollo

abiertos, representados en 200.000 proyectos, 4.9 billones de líneas de código; utilizando su detallado conocimiento de los proyectos abiertos y aplicando técnicas estándar de estimación de costos, calculan que el costo de desarrollo total del proyecto FOSS excede los USD\$387000 millones y representa la inversión colectiva de mas de dos millones de desarrolladores al año. Un análisis adicional, estima que el 10% de las aplicaciones utilizadas para el desarrollo de aplicaciones en tecnología de la información se pueden reemplazar por proyectos abiertos², lo que ahorraría mas de USD\$22 billones al año.

Los proyectos de código abierto permiten a las organizaciones ahorrar tiempo y dinero en muchos aspectos, al no tener que pagar por las herramientas de desarrollo y por librerías y aplicaciones que pueden utilizar para la implementación de nuevos productos; permitiendo la inversión de tiempo y esfuerzo en proyectos que pueden ser comercializados rápidamente.

Dispositivos semiconductores

En la actualidad existe una gran oferta de SoCs, grandes compañías proporcionan constantemente nuevos dispositivos con una gran variedad de periféricos para diferentes aplicaciones. El procesador más utilizado para aplicaciones embebidas es el procesador ARM (Advanced RISC Machine). ARM no fabrica circuitos integrados, suministra sus diseños en forma de *netlist* a nivel de compuertas o a nivel de Lógica de Transferencia de Registros (Register-transfer level (RTL)) en un lenguaje de descripción de hardware. Estas descripciones pueden ser utilizados en el proceso de diseño ASIC, permitiendo su integración con una gran variedad de núcleos Intellectual Property (IP) (Intellectual Property); compañías como Atmel, Marvell, Freescale, NXP, Cirrus Logic, Samsung, Texas Instruments adquieren licencias que les permiten utilizar estos núcleos lógicos en la fabricación de sus SoCs.

Encuestas realizadas a diseñadores por el portal *linuxfordevices*² sobre sus preferencias en el procesador utilizado en sus proyectos; como se dijo anteriormente. ARM es el más utilizado (30 %) seguido de cerca por los basados en x86 (25 %), la arquitectura POWERPC (15 %), MIPS (10 %), DSPs (5 %). Por este motivo, en esta investigación se utilizaron dispositivos basados en procesadores ARM (AT91RM9200 y SAM7 de Atmel, imx233 de Freescale), MIPS (JZ4740 de Ingénierie) y el DSP de Analog devices BF532, cubriendo de esta forma un gran campo de aplicaciones.

Dispositivos disponibles

Existen dos tipos de dispositivos que pueden ser utilizados en el proceso de transferencia tecnológica y de conocimientos: productos de consumo, productos fabricados a grandes volúmenes y que se pueden adquirir fácilmente a bajo precio; y dispositivos *Commercial off-the-shelf* (*Commercial off-the-shelf* (COTS)), diseñados para ser utilizados como punto de partida en el desarrollo de una aplicación, se pueden encontrar en la forma de tarjetas de desarrollo, las cuales incluyen una gran variedad de dispositivos hardware (puertos de comunicación, Light-Emitting Diode (LED)s, Displays, Liquid Crystal Display (LCD)s, pulsadores) que permiten explorar la capacidad de un determinado SoC, o se pueden encontrar en forma de unidades genéricas que proporcionan las conexiones para su funcionamiento básico (alimentación, memorias), proporcionando todas las señales que controlan periféricos externos para que sean utilizadas en una tarjeta que integre los componentes requeridos

²software utilizando software libre

²Linuxfordevices es un sitio de encuentro y centro de noticias para la comunidad que utiliza Linux en dispositivos digitales.

en una determinada aplicación. El término Original Equipment Manufacturer (OEM) (Original Equipment Manufacturer) se aplica a las organizaciones que compran estos artículos y los revenden; en algunas ocasiones se realizan mejoras como valor agregado.

En la actualidad existe una gran oferta de este tipo de dispositivos a nivel mundial, muchas compañías realizan diseños para que sean utilizados como punto de partida de productos comerciales, o que sean parte de ellos, lo que reduce los costos y tiempos de desarrollo; desafortunadamente en el país no existe una gran demanda, por lo que no existen proveedores locales y deben ser importados, lo que hace que su costo se eleve por lo menos en un 26 % (arancel e IVA).

Durante la primera etapa de este estudio se utilizaron los siguientes dispositivos: GameBoy + Xport, es una combinación de la consola de juegos ©NINTENDO (basado en ARM7) y la plataforma *XPORT* de la compañía norteamericana Charmedlabs (basado en una FPGA Spartan3); la plataforma de desarrollo de la compañía china *Embest* basada en el SoC AT91R40008 (ARM7); la agenda electrónica Zaurus de Toshiba (basado en el StrongARM de Intel SA-1110); el dispositivo chumby (basado en un ARM de freescale iMX21); el MP4 de Ainol V2000 que posee un procesador MIPS fabricado por la compañía china Ingénic; la agenda iPAQ H3600 fabricada por Compaq basada en un procesador Strong ARM de Intel; el porta-retratos digital de la empresa china *SUNGALE* basada en un procesador MIPS de la empresa Ingénic y finalmente el lector de libros electrónicos de Barnes & Noble basado en un procesador ARM fabricado por Samsung.

Discusión

En la actualidad es posible adquirir dispositivos semiconductores y demás componentes necesarios para la fabricación de sistemas digitales, sin embargo, para poder competir en precios con productos del mercado asiático, es necesario entender la dinámica de su industria manufacturera, y establecer relaciones con los proveedores de bienes y servicios en esta región. El mercado asiático se ha especializado en el aprovechamiento de recursos y posee diferentes niveles de calidad de los componentes y productos; por esta razón, es posible encontrar teléfonos móviles desde USD \$10 hasta los más costosos de cientos de USDs; los bajos costos son posibles gracias al reciclaje de componentes, una gran cantidad de trabajadores desmontan circuitos integrados y componentes de dispositivos desecharados, y son re-utilizados en nuevos productos, el nivel de reciclaje es tan alto que únicamente las placas de circuito impreso son descartadas; obviamente la confiabilidad de estos dispositivos es menor que la de los dispositivos nuevos, por lo cual es muy importante contar con fuentes de suministros confiables. Por otro lado, en el mercado asiático es posible conseguir dispositivos semiconductores que no se conocen en occidente, y normalmente son más baratos que productos similares de compañías reconocidas mundialmente, sin embargo, muchos de estos dispositivos se encuentran disponibles únicamente para ciudadanos chinos, lo que hace imprescindible el establecimiento de relaciones con facilitadores asiáticos.

3.2.3. Adaptación

En esta etapa se utilizaron las plataformas comerciales mencionadas anteriormente y las herramientas de desarrollo abiertas para: explorar las diferentes alternativas de implementación; entender y aplicar la metodología de diseño adoptada internacionalmente; generación de conocimiento sobre el uso de las herramientas de desarrollo.

Los dispositivos comerciales con grandes niveles de producción son un buen punto de partida para el estudio de la arquitectura de sistemas digitales que utilizan SoCs, ya que están optimizadas

(en número de componentes) para ser utilizadas en una aplicación específica, gracias a esto, su costo es muy bajo (en comparación con plataformas de desarrollo). Por otro lado, es posible encontrar estos dispositivos en el mercado local; reduciendo de forma considerable la inversión necesaria para realizar las actividades de esta etapa.

Metodología

Se utilizó ingeniería inversa para determinar como están construidos, como funcionan y como son programados los productos adquiridos. Esta tarea no se ejecutó en un único instante de tiempo, ya que el estudio se realizó de forma gradual, de tal forma que cada vez que se abordaba un nuevo tópico se repetía el proceso con un producto que permitiera su estudio.

Como se puede ver en la figura 3.4 existen dos formas de realizar las aplicaciones de un sistema embebido: con sistema operativo (OS) y sin Operating System (OS) (*standalone*). Las aplicaciones standalone son muy eficientes debido a que son escritas teniendo en cuenta los recursos hardware (memoria, velocidad del procesador y periféricos) de la plataforma donde van a ser ejecutadas, sin embargo, requiere el desarrollo completo de toda la funcionalidad. Por otro lado, los sistemas operativos proporcionan servicios (manejo de periféricos, capacidad de ejecución multi-tarea, manejo de sistemas de archivos y librerías) que facilitan el desarrollo de aplicaciones; sin embargo, al ser diseñados para ser ejecutados en cualquier dispositivo, es necesario cumplir con unos requerimientos de recursos mínimos para que se puedan ejecutar las aplicaciones. En este estudio se trabajó con dos sistemas operativos abiertos: *eCos* desarrollado por Redhat y *Linux* desarrollado por Linus Torvalds. A continuación se muestran los temas de estudio que se abordaron en esta etapa.

- Arquitectura: Determinar los componentes más utilizados y las diferentes topologías.
- Programación: Mecanismos que permiten cambiar el *firmware* original del dispositivo.
- Aplicaciones sin sistema operativo: Estudio de herramientas que permiten generar nuevas aplicaciones en los dispositivos sin el uso de sistemas operativos.
- Aplicaciones con sistema operativo: Uso de sistemas operativos para acelerar el proceso de desarrollo de aplicaciones, utilizando las facilidades que ellos proporcionan.
 - *eCos*: Sistema operativo de tiempo real altamente configurable que permite el uso eficiente de los recursos hardware.
 - *Linux*: Sistema operativo ampliamente utilizado que posee una gran cantidad de aplicaciones y un número considerable de desarrolladores.
 - Inicialización: Requerimientos mínimos para la ejecución de *Linux*.
 - Imagen del kernel: Adaptación del kernel de *Linux* a una determinada plataforma.
 - Sistema de archivos: Distribuciones y aplicaciones mínimas necesarias para la ejecución de *Linux*.
 - Drivers de dispositivos: Como dar soporte a nuevos periféricos.
 - Comunicación con periféricos dedicados implementados en FPGAs: Comunicación entre FPGAs y SoCs.
 - Aplicaciones gráficas: Uso de librerías gráficas para el desarrollo de aplicaciones.

Arquitectura: SoC, memorias, periféricos

Los SoCs comerciales se pueden dividir en dos grandes grupos dependiendo de la existencia o no de memoria no volátil para el almacenamiento del programa (memoria de instrucciones) dentro del SoC. Los que poseen memoria no volátil (hasta 512 Kbytes) normalmente incorporan una memoria Random Access Memory (RAM) (hasta 32 kbytes) junto con una serie de periféricos (timers, Inter-Integrated Circuit (I2C), Serial Peripheral Interface (SPI), Universal asynchronous receiver/transmitter (USART)s, Analog-to-digital converter (ADC)s, Watchdog, USB device, canales para acceso directo a memoria - Direct memory access (DMA)); están diseñados para no utilizar componentes externos; normalmente este tipo de dispositivos utilizan procesadores que no tienen unidad de manejo de memoria ³ (MMU) como la familia ARM7, cuyas velocidades de ejecución varían entre los 50 y 70MHz. En la figura 3.5 se muestra la arquitectura típica de un sistema basado en estos dispositivos.

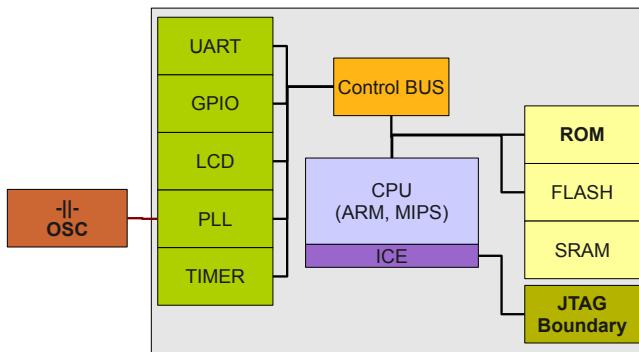


Figura 3.5.: Arquitectura típica de un sistema embebido que utiliza SoC con memoria volátil interna

Los procesadores que no poseen memoria no volátil interna se dividen en dos grupos: los que poseen o no unidad de manejo de memoria; en ambos casos, se cuenta con una memoria RAM interna (del orden de cientos de Kbytes) y adicionalmente a los periféricos mencionados anteriormente se suministran controladores para USB host, puertos Serial Synchronous Interface (SSI), controlador de LCD, codecs de audio, controlador de touch screen; debido a la ausencia de memoria no volátil interna, estos dispositivos poseen periféricos dedicados al manejo de memorias no volátiles NAND flash, NOR flash, SPI, I2C y Secure Digital (SD); y memorias volátiles Synchronous dynamic random-access memory (SDRAM) y Double Data Rate (DDR); su velocidad de operación varía entre los 75MHz y 800MHz. En la figura 3.6 se muestra la arquitectura típica de un sistema basado en estos procesadores.

Debido a la falta de memoria volátil, las aplicaciones de este tipo de dispositivos requieren una memoria externa para almacenar las aplicaciones básicas y datos, en la actualidad las más populares son las memorias NAND flash, NOR flash, SPI, Electrically-Erasable Programmable Read-Only Memory (EEPROM) y SD. Normalmente, este tipo de procesadores son utilizados en aplicaciones que utilizan sistemas operativos, como se verá más adelante. Para que ciertos sistemas operativos (Linux, Windows CE) puedan ejecutarse se requiere una mínima cantidad de memoria RAM (del

³La Memory management unit (MMU) permite el manejo de memoria, dentro de sus funciones se encuentra el traslado de la memoria física a virtual, protección de la memoria, control de cache, control de buses

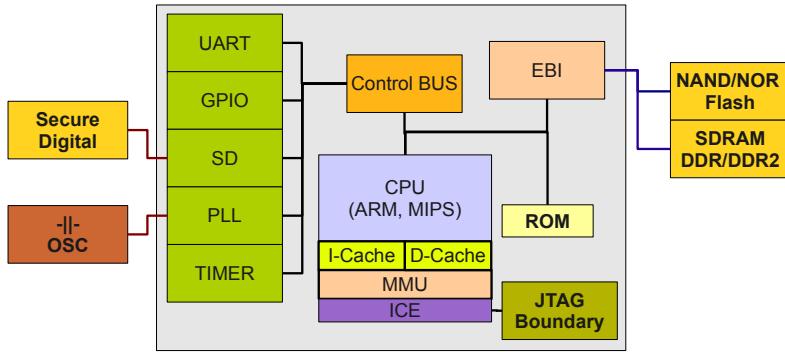


Figura 3.6.: Arquitectura típica de un sistema embebido que utiliza SoC sin memoria volátil interna

orden de los Mbytes), por esta razón es necesario incluir una memoria RAM externa, en la actualidad las más utilizadas son las SDRAM, DDR y DDR2.

Como conclusión, se puede decir que en el mercado existen diferentes arquitecturas de SoCs que permiten realizar proyectos con diferentes grados de complejidad y que se ajustan a las opciones más utilizadas por los desarrolladores; la opción más económica es la utilización de un SoC que incluya las memorias no volátiles y RAM internamente; sin embargo, hasta el momento no existen dispositivos con grandes capacidades de memoria Flash y RAM internas, por lo que no es recomendado su uso en ciertas aplicaciones. Utilizar un SoC que no integre las memorias no volátiles proporciona una mayor flexibilidad, ya que estos dispositivos proporcionan periféricos que pueden controlar varios tipos de memorias, y se puede elegir la más económica, algo similar ocurre con la memoria RAM; sin embargo, el costo total de las memorias externas, SoC y área de circuito impreso es mayor que en el caso anterior. En la tabla 3.1 se resume la arquitectura de las plataformas utilizadas.

Tabla 3.1.: Arquitectura de las plataformas comerciales utilizadas en la etapa de absorción

Plataforma	CPU	Mem. volátil	Mem. no volátil	MMU	OS
Game Boy	ARM 7	Externa NOR	Interna	NO	Propietario
Zaurus	Strong ARM	Externa NAND	Externa SDRAM	SI	Linux
iPAQ H3600	Strong ARM	Externa NAND	Externa SDRAM	SI	Windows CE
Chumby	ARM MC94MX21	Externa NAND	Externa SDRAM	SI	Linux
Ainol V2000	MIPS JZ4740	Externa NAND	Externa SDRAM	SI	Linux
SUNGALE DPF	MIPS JZ4740	Externa NAND	Externa SDRAM	SI	Linux
B&N NOOK	ARM S3C6410	Externa SD	Externa SDRAM	SI	Android

Aunque estos procesadores operan a velocidades entre los 75 y 800 MHz, no todos los componentes del SoC operan a esta frecuencia, el componente externo que requiere la mayor velocidad de operación es la memoria RAM y puede estar entre los 50 y 130 MHz, los demás periféricos funcionan a frecuencias del orden de las decenas de MHz o KHz; por esta razón estos SoC suministran un circuito PLL que permite generar la frecuencia de operación a partir de cristales de frecuencias del orden de las decenas de MHz, lo que facilita el diseño de la placa de circuito impreso.

Cada periférico requiere una conexión específica con el dispositivo que controla, los SoC

modernos incluyen la mayor parte del circuito internamente con el objetivo de minimizar las conexiones y dispositivos adicionales. Existen tendencias de los fabricantes a agrupar periféricos teniendo en mente dos aplicaciones: multimedia, e industriales; para aplicaciones multimedia se proporcionan controladores de LCDs, ratón, teclado, pantalla táctil, codificador-decodificador (CODEC)s de audio, control de potencia, relojes de tiempo real, control de carga de baterías entre otros; para aplicaciones industriales se proporcionan controladores de red cableada, puertos Controller area network (CAN), I2C, y SPI.

Programación

Como se mencionó anteriormente, para este estudio se utilizaron herramientas abiertas para la creación de aplicaciones, en la figura 3.7 se muestra el flujo de creación de las tareas software usando la cadena de herramientas GNU is Not Unix (GNU) [38]. La ventaja de utilizar estas herramientas (adicional a la económica) es el soporte a diferentes procesadores (24 diferentes CPUs, incluyendo micro-controladores de 8 bits), lo que permite la fácil migración entre CPUs; adicionalmente su alto grado de configurabilidad permite el cambio de disposición de las memorias volátiles y no volátiles de forma fácil (a través del script de enlace). El proceso de generación del archivo binario que debe ser grabado en la memoria no volátil de la plataforma puede ser realizado en su totalidad por la cadena de herramientas GNU.

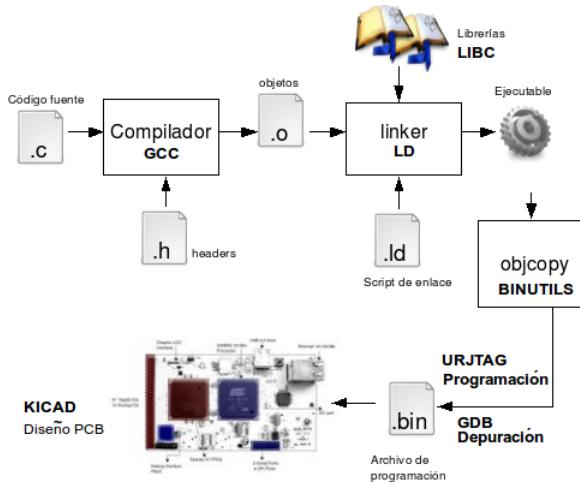


Figura 3.7.: Flujo de diseño software para creación de aplicaciones.

Los SoC poseen la capacidad de *iniciar* desde diferentes dispositivos; cuando se activa la señal de *reset* a un SoC que no posee memoria volátil interna, el primer programa en ejecutarse es el que reside en una memoria Read Only Memory (ROM) interna, este programa revisa varios periféricos en búsqueda de un programa válido; los periféricos soportados varían según el fabricante, pero por lo general siempre soportan el uso de memorias NOR Flash (paralelas) y en SoCs más recientes memorias NAND Flash, SPI, o SD; sin embargo, la mayoría de SoC soportan memorias que se encuentran soldadas en la placa de circuito impreso, lo que hace necesario buscar métodos de programación de estas memorias que no implique desmontarlas o el uso de costosos conectores. En la mayoría de los SoC, cuando el programa residente en la ROM no encuentra ninguna aplicación válida

en los periféricos soportados, establece una comunicación por uno de sus puertos seriales o USB y queda en espera del envío de un programa válido, el programa enviado es almacenado en la memoria RAM interna, y una vez finaliza su descarga se ejecuta desde la RAM interna. La figura 3.8 muestra este proceso.

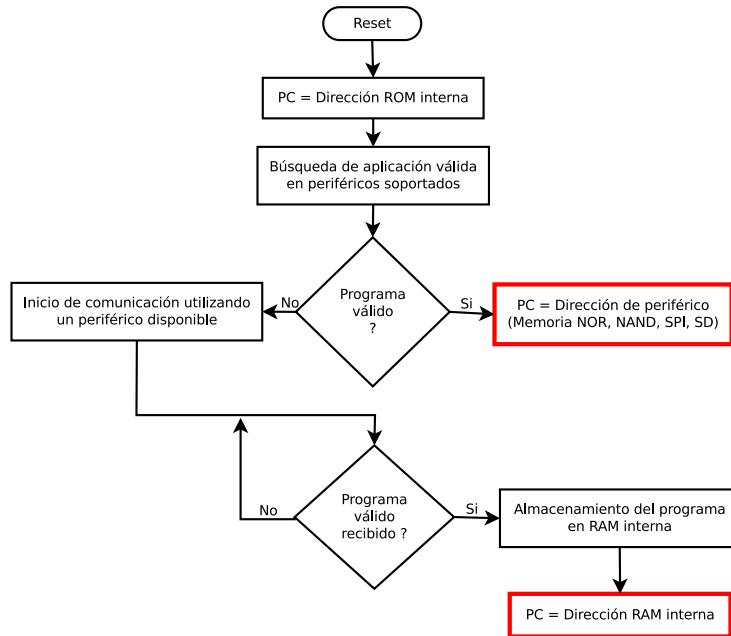


Figura 3.8.: Inicialización de un SoC cuando las memorias no volátiles no están programadas.

Debido a que la RAM interna normalmente es pequeña (del orden de decenas de Kbytes), no es posible cargar aplicaciones muy grandes en ella, por lo que es necesario realizar el proceso de programación en varias etapas: en la primera etapa se carga una aplicación (*first - stage bootloader*) que se encarga de configurar el procesador (pila, frecuencia de operación), configurar la memoria RAM externa y habilitar un canal de comunicación para descarga de aplicaciones, de esta forma, es posible almacenar aplicaciones tan extensas como la capacidad de la memoria RAM externa (del orden de MBytes). En la segunda etapa se descarga una aplicación a la memoria externa que tiene la capacidad de programar las memorias no volátiles externas con la información proveniente de los diferentes periféricos de comunicación del SoC (como puerto serial, memoria SD, USB), este segundo programa recibe el nombre de *bootloader* y se auto-almacena en las primeras posiciones de la memoria no volátil, de tal forma que sea ejecutado después de la activación de la señal de *reset* y de la búsqueda que realiza el programa interno de la ROM.

Una vez programada la memoria no volátil con una aplicación válida, los SoCs realizan una serie de pasos para ejecutar las aplicaciones almacenadas en ella, esto debido a la poca capacidad de la memoria RAM interna. Como se dijo anteriormente, una vez se activa la señal de *reset* se ejecuta un programa contenido en la memoria ROM interna del SoC (figura 3.9 (a)), esta aplicación configura un periférico que permite la comunicación con los dispositivos de almacenamiento masivo externos, y además copia una determinada cantidad de información desde la memoria no volátil externa a la memoria RAM interna (figura 3.9 (b)), esto se hace porque el programa en la ROM no conoce la configuración de la plataforma y esta puede cambiar según la aplicación; después de esto ejecuta la

aplicación copiada a la memoria RAM interna colocando en el contador de programa (PC) el valor correspondiente a la memoria RAM interna (figura 3.9 (c)).

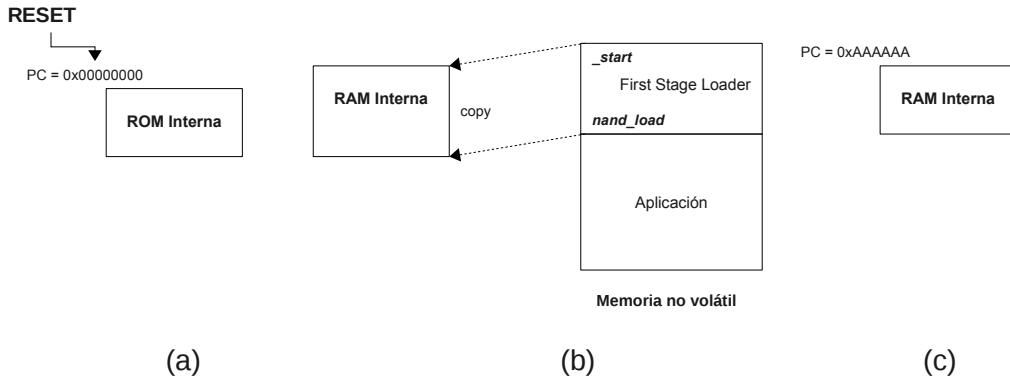


Figura 3.9.: Inicialización de un SoC cuando la memoria no volátil está programada, parte 1.

Este programa (*loader*) está encargado de: configurar la memoria RAM externa (su capacidad varía dependiendo de la aplicación) y de copiar la aplicación propiamente dicha desde la memoria no volátil a la memoria RAM externa, (con lo que es posible cargar aplicaciones de mayor tamaño que la memoria RAM interna); finalmente, el *loader* ejecuta la aplicación almacenada en la memoria RAM haciendo que el contador de programa (PC) sea igual a la dirección donde se almacenó esta aplicación (ver figura 3.10)

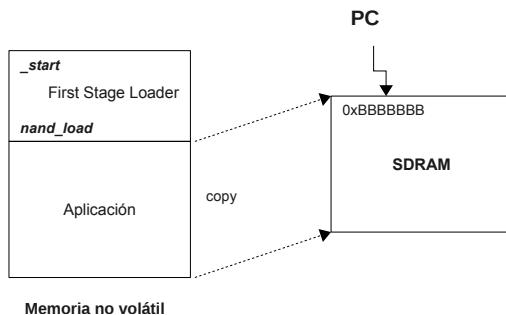


Figura 3.10.: Inicialización de un SoC cuando la memoria no volátil está programada, parte 2.

Programación utilizando el puerto JTAG

Algunos SoCs no suministran un camino para la programación de la memoria RAM interna, para estos casos, se puede utilizar un periférico que la mayoría de los dispositivos proporciona: el puerto Joint Test Action Group (JTAG) (creado inicialmente como un mecanismo para realizar pruebas en las tarjetas de circuito impreso para verificar la correcta conexión entre componentes, y verificar el correcto funcionamiento de los circuitos integrados) el cual, está formado por un registro de desplazamiento (ver figura 3.11) que controla el paso de información desde y hacia cada uno de los pines del circuito integrado, permitiendo realizar varias operaciones. Con el paso del tiempo, se han

adiccionado funcionalidades a este protocolo y una de ellas es el control de circuitos especializados dentro de los SoCs para realizar emulación en circuito (In Circuit Emulation (ICE)), suministrando un canal para la programación de la memoria RAM interna.

Algunos SoCs antiguos no poseen una unidad de emulación en circuito por lo que no es posible acceder a la memoria RAM interna, en estos casos es posible utilizar el protocolo JTAG para controlar directamente los pines del SoC conectados a las memorias no volátiles y ejecutar los protocolos de programación de las mismas; debido a que es necesario programar todos los registros de la cadena Boundary Scan, el tiempo de programación suele ser más largo que cuando se utiliza el ICE.

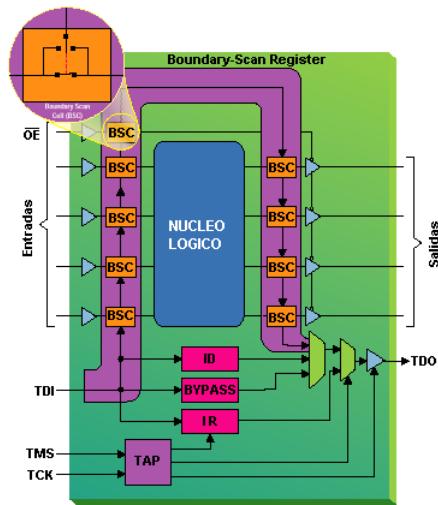


Figura 3.11.: Cadena Boundary Scan fuente: Texas Instruments.

Aplicaciones *standalone* vs aplicaciones con sistema operativo

Los sistemas operativos proporcionan facilidades al programador que permiten acelerar el desarrollo de aplicaciones, suministrando una capa de abstracción de hardware que permite manejar los periféricos a alto nivel sin preocuparse por el manejo tedioso a nivel de registros; adicionalmente, proporciona soporte para aplicaciones en red, manejo de sistemas de archivos, multitarea, seguridad, entre otras (ver figura 3.12; adicionalmente, existen librerías especializadas que ayudan al desarrollo en diferentes áreas. Sin embargo, el uso de sistemas operativos como Linux, Android, Mac OS o Windows, exige el cumplimiento de condiciones mínimas para su uso; por ejemplo, Linux requiere 8 Mbytes de memoria RAM y 2 Mbytes de memoria no volátil, Android requiere 128 Mbytes de memoria RAM y 32 Mbyte de memoria no volátil; por esta razón es necesario agregar dos memorias externas, lo que aumenta la complejidad de la placa de circuito impreso y el costo del dispositivo. Por otro lado, los sistemas operativos tienen una particularidad en su funcionamiento que recibe el nombre de *latencia*; y se define como el tiempo que transcurre entre la generación de un evento (interrupciones hardware o software) y la respuesta ante este evento, este tiempo varía según el estado de carga del sistema; en un sistema operativo de tiempo real esta latencia es conocida y no depende de la carga de sistema. Esta latencia en algunas aplicaciones hace imposible el manejo de eventos ya que es posible que se pierdan algunos cuando el sistema se encuentre muy cargado.

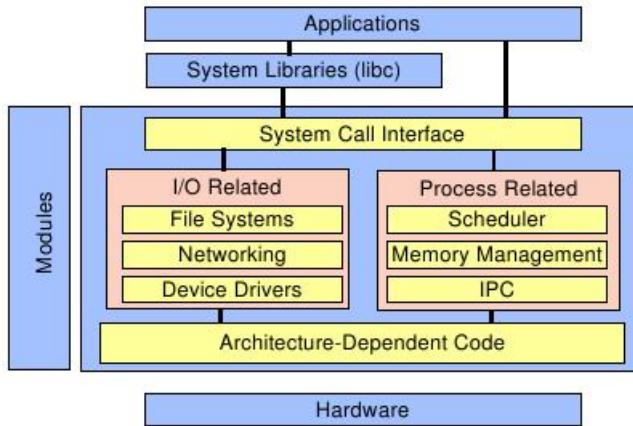


Figura 3.12.: Estructura del kernel de Linux.

Las aplicaciones *standalone* utilizan los recursos necesarios y dependiendo de su complejidad pueden ajustarse a un SoC que incorpore la memoria RAM y no volátil internamente; adicionalmente, su programación puede reducir el problema que se presenta en la latencia a las interrupciones en los sistemas operativos. Sin embargo; es necesario dar soporte a todos los periféricos que se utilizarán y se deben escribir todas las rutinas, lo que puede aumentar el tiempo de desarrollo.

En conclusión, el uso de sistemas operativos o aplicaciones standalone depende de la complejidad de la aplicación, y de consideraciones económicas como el *time to market* y costo de los desarrolladores.

Conocimientos adquiridos

En la tabla 3.2 se muestran los conocimientos adquiridos al aplicar ingeniería inversa a las plataformas comerciales utilizadas en esta etapa.

Tabla 3.2.: Conocimientos Adquiridos durante la etapa de adaptación

Plataforma	Herramientas de desarrollo	Programación	Sistema Operativo y Aplicaciones
Game Boy	ARM GNU toolchian	Puerto serie	eCos, implementación de periféricos en FPGAs.
Zaurus	ARM GNU toolchian	Puerto serie	Linux 2.4, sistema de archivos, QT.
iPAQ H3600	ARM GNU toolchian	Puerto serie	Linux 2.4, Buildroot, QT.
Chumby	ARM GNU toolchian	Puerto serie	Linux 2.6, u-boot, Open-Embedded, QT, flash.
Ainol V2000	MIPS - ELF GNU toolchian	Puerto serie	Linux 2.6, openwrt, QT.
SUNGALE DPF	MIPS - ELF GNU toolchian	Puerto serie	Linux 2.6, openwrt, QT.
B&N NOOK	ARM GNU toolchian	Puerto JTAG, serie	Linux 2.6, Android Dalvik (VM).

Discusión

En la etapa de adopción se utilizaron herramientas comerciales para aprender: como están construidos; como funcionan y como se programan dispositivos comerciales; su utilización facilita el aprendizaje del funcionamiento de la tecnología, permitiendo el estudio de metodologías de diseño y la evaluación de herramientas de desarrollo, reduciendo el tiempo y el costo de desarrollo de plataformas propias diseñadas para dicho fin; adicionalmente, pueden ser utilizadas como base de nuevos productos comerciales. Sin embargo, se notaron los siguientes inconvenientes durante su utilización: al ser diseñadas para una aplicación específica son difíciles de adaptar a nuevas aplicaciones; los fabricantes no proporcionan documentación sobre su arquitectura; su uso crea dependencia hacia el fabricante y puede reducir la demanda de diseñadores y proveedores de bienes y servicios de manufactura; solo pueden utilizarse en aplicaciones similares para las que fueron creadas.

En esta etapa se utilizaron plataformas de desarrollo comerciales diseñadas para proporcionar herramientas en el estudio de un determinado procesador, en la mayoría de los casos utilizan la mayor parte de sus recursos (del procesador) y ponen a disposición del usuario una gran cantidad de periféricos, conectores y señales, lo que facilita el aprendizaje y su uso en aplicaciones comerciales. Sin embargo, su costo es relativamente alto (en comparación con los dispositivos comerciales de alto nivel de producción); debido a su gran oferta de conexiones son tarjetas muy grandes y difíciles de adaptar en algunas aplicaciones reales; por otro lado, es necesario fabricar tarjetas adicionales para que se adapten a la aplicación lo que incrementa el costo final.

3.2.4. Absorción y Asimilación

En esta etapa se realizan actividades que integran nuevo conocimiento para el país pero que no es nuevo para el mundo y están encaminadas al desarrollo o adaptación de metodologías de diseño y procesos de fabricación, desarrollo de productos tecnológicos propios, enseñanza de metodologías de diseño y procesos de fabricación.

La utilización de plataformas de desarrollo o dispositivos comerciales como punto de partida para nuevos productos es viable en casos donde solo se requiera un cambio de software, ya que adiciones en el hardware aumentarían el costo total o simplemente no se pueden hacer debido a que no se tiene acceso a los archivos de diseño. Para poder ser competitivos, es necesario reducir los costos de componentes y de fabricación, para lograr esto es necesario utilizar la mínima cantidad de componentes para la aplicación, lo que implica la producción de placas de circuito impreso (PCBs). El país no cuenta con una buena oferta para la producción de PCBs, solo existen tres empresas que pueden ofrecer productos que cumplen con los requerimientos: Microensamble en Bogotá, Colcircuitos en Medellín y Microcircuito en Cali, de los cuales, solo las dos primeras proporcionan el servicio de tarjetas de 1 a 10 capas.

El diseño de PCBs es una actividad que da empleo a profesionales de diferentes niveles de formación (técnicos, ingenieros, diseñadores) en los países asiáticos; desafortunadamente en Colombia existen muy pocas empresas que proporcionan este servicio, lo que evidencia la falta de demanda interna en las actividades relacionadas con la manufactura de dispositivos digitales; adicionalmente, en la mayoría de las universidades no se enseñan normas para el diseño de PCBs ni se somete a los estudiantes a este proceso de diseño, aceptando soluciones intermedias como el uso de placas de proyectos (project boards) o baquelitas universales o simplemente se ignora la parte de implementación física y el último paso es la simulación. Por estos motivos, una parte importante en esta etapa es el diseño y construcción de tarjetas de desarrollo que serán utilizadas para explorar

las técnicas de fabricación y montaje y posteriormente ser utilizadas en el desarrollo de aplicaciones académicas.

Plataformas de desarrollo

La primera plataforma desarrollada en este proyecto utiliza el SoC AT91R40008 de ATMEL y se le dió el nombre de ECB_ARM7, el diagrama de bloques de la plataforma y la foto del prototipo se muestra en la figura 3.13, ECB_ARM7 posee componentes de montaje superficial y fue el punto de partida para el estudio de técnicas de fabricación de placas de circuito impreso, con ella se realizaron aplicaciones *standalone* y con el sistema operativo eCos. Dentro del proceso de diseño de sistemas digitales, algunas de las tareas deben ser implementadas en hardware para cumplir con los requerimientos temporales; por este motivo es importante realizar implementaciones de tareas hardware en dispositivos lógicos programables (PLDs); para estudiar la forma de comunicar estas tareas hardware con SoC comerciales se diseñó un clon de la tarjeta comercial XPORT de la empresa norteamericana Charmedlabs, que permite conectarse con el procesador de la plataforma comercial ©GameBoy Advance de Nintendo [39] (ver figura 3.14).

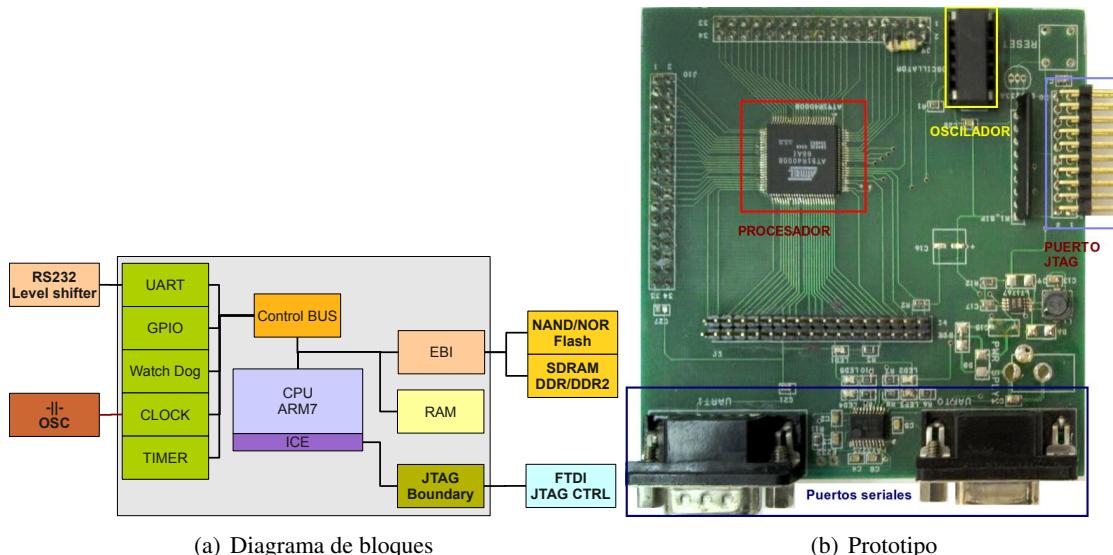


Figura 3.13.: Plataforma de desarrollo ECB_ARM7

El uso de Linux como herramienta de desarrollo ha aumentado en los últimos años, llegando a ser el sistema operativo más utilizado por los desarrolladores (ver figura 3.4); empresas como Nokia, Motorola, Google, Dlink, Hewlett Packard utilizan Linux para desarrollar el firmware de sus dispositivos; por este motivo, se diseñaron plataformas que soportan la ejecución de este sistema operativo. Se diseñó una familia de plataformas denominadas ECB_AT91 V1 (ver figura 3.15 [40] [41]), ECB_AT91 V2 (ver figura 3.16) y ECBOT (ver figura 3.17 [42] [43] [44]); esta familia de plataformas se encuentran registradas en el kernel oficial de Linux, por lo tanto, su soporte queda garantizado en futuras versiones del kernel. ECB_AT91 V1 fue el punto de partida para el estudio de técnicas de fabricación de placas de circuito impreso para SoC que operan a velocidades mayores de 180MHz, y para aprender el proceso completo de adaptar el kernel de Linux a una plataforma nueva, el proceso detallado se muestra en el apéndice ???. En la figura 3.17 (a) se muestra el diagrama

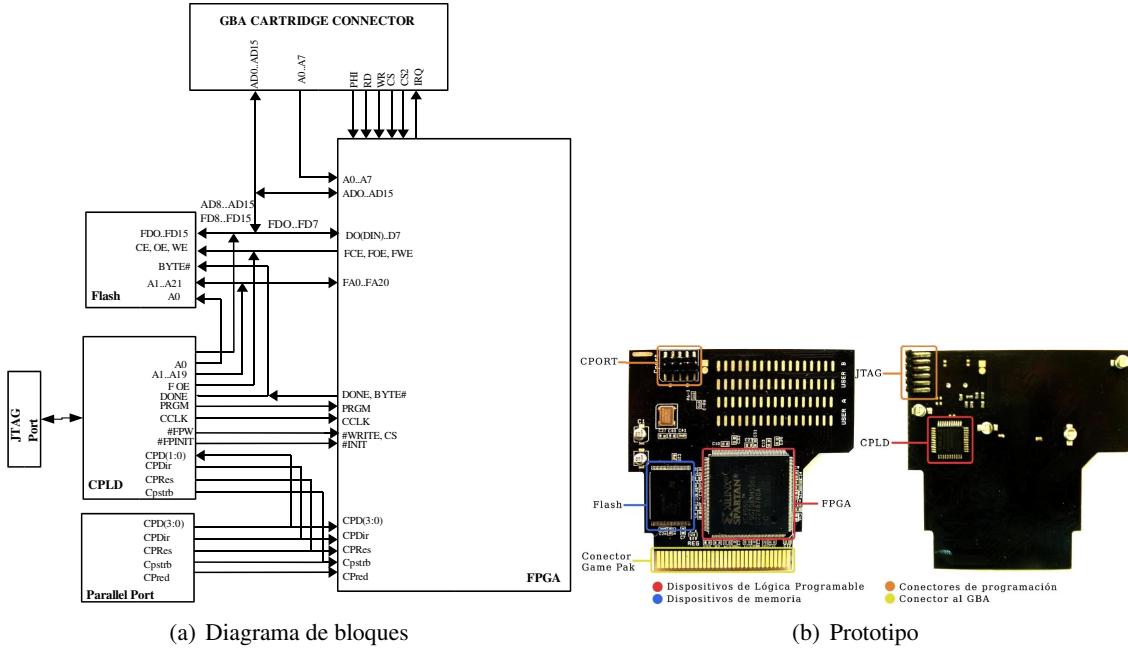


Figura 3.14.: Plataforma de desarrollo UNAL_UIS_XPORT

de bloques de estas tres arquitecturas; todas poseen el SoC de 180MHz de Atmel AT91RM9200, debido a que este SoC no posee conversores análogos - digitales internos fue necesario incluir un micro-controlador de 8 bits que se comunica con el SoC vía I2C, el SoC realiza la programación del microcontrolador emulando un puerto paralelo en sus pines de entrada/salida de propósito general (General Purpose Input/Output (GPIO)s) que van conectados a los pines del puerto In System Programming (ISP) del microcontrolador.

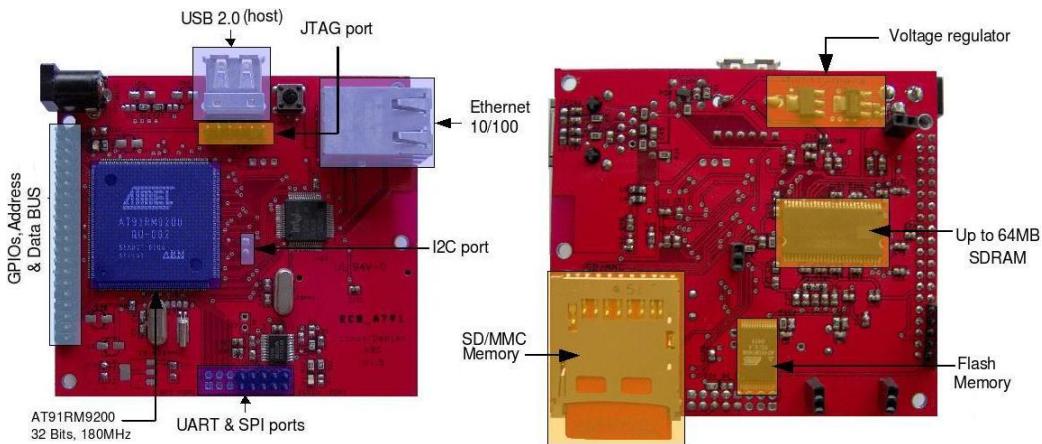


Figura 3.15.: Plataforma de desarrollo ECB_AT91 V1

La plataforma ECB_AT91 V2 es el resultado del interés de utilizar dispositivos lógicos programables y SoC comerciales para crear periféricos dedicados y la creación de nuevos drivers de Linux que los controlaran, su arquitectura es idéntica a la de la plataforma ECB_AT91 V1 pero

el bus de datos, dirección y control del SoC se conectan a la FPGA para permitir la comunicación con los periféricos implementados en ella; la FPGA es configurada por el SoC utilizando unos GPIOs que van conectados al puerto JTAG de la FPGA, lo que elimina la necesidad de cables adicionales de conexión, permitiendo su re-programación total de forma remota.

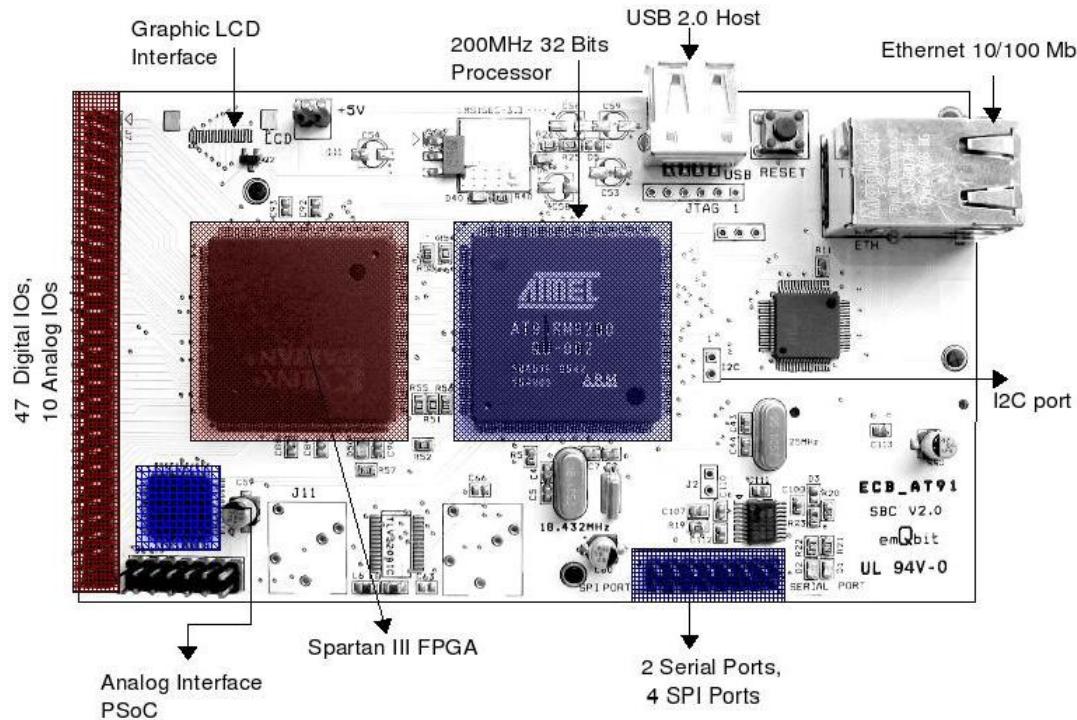


Figura 3.16.: Plataforma de desarrollo ECB_AT91 V2

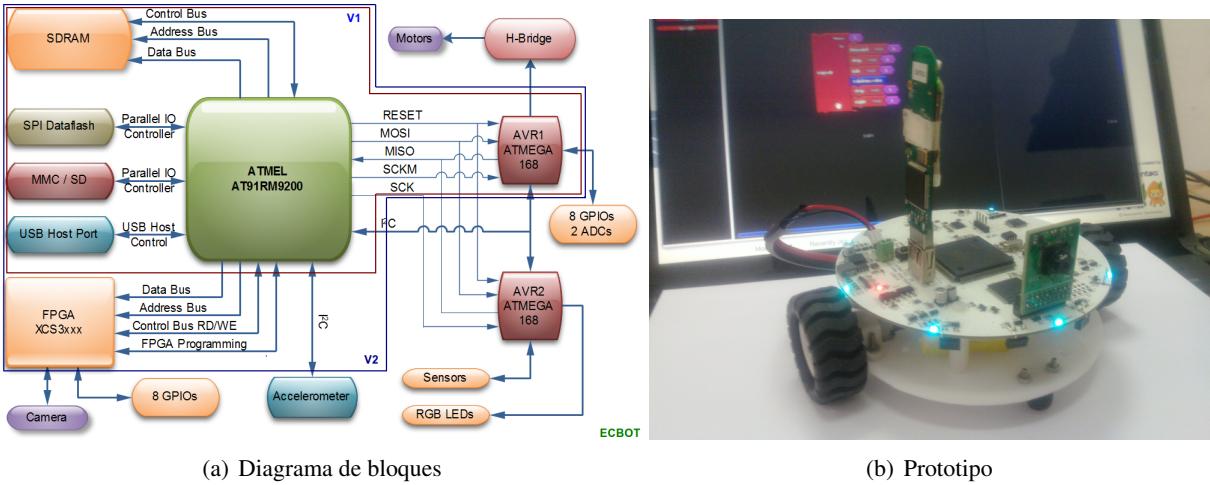
La plataforma ECBOT fue diseñada con el propósito de iniciar el estudio en robótica móvil y procesamiento de imágenes; para esto, se utilizó una arquitectura similar a la de la plataforma ECB_AT91_V2 adicionando una conexión dedicada a un sensor de imagen y circuitos especializados para el control de sensores (6 sensores de choque, 1 sensor de imagen, movimiento de los motores basado en BEMF⁴), actuadores (8 LEDs RGB, 2 motores DC) y un conversor DC/DC que garantiza el máximo uso de las baterías. Como parte de este trabajo, se realizó la adaptación de proyectos libres que facilitan la investigación en robótica como *player/stage*⁵ y *openblocks*⁶. En la actualidad ECBOT está siendo usado por los grupos de robótica de los departamentos de Sistemas y mecatrónica de la Universidad Nacional de Colombia.

Los SoC utilizados en las plataformas anteriores no pueden ser utilizados en aplicaciones de procesamiento de señales o de imágenes en tiempo real (a menos que se implemente algún periférico en la FPGA), ya que su arquitectura no está diseñada para este fin. Los DSPs son procesadores especializados para ejecutar las operaciones más comunes en procesamiento digital de señales; pensando en este tipo de aplicaciones se diseñó la plataforma ECB_BF532 [45] que utiliza el DSP BF532 de la familia Blackfin de Analog Devices. Esta plataforma será utilizada en los cursos

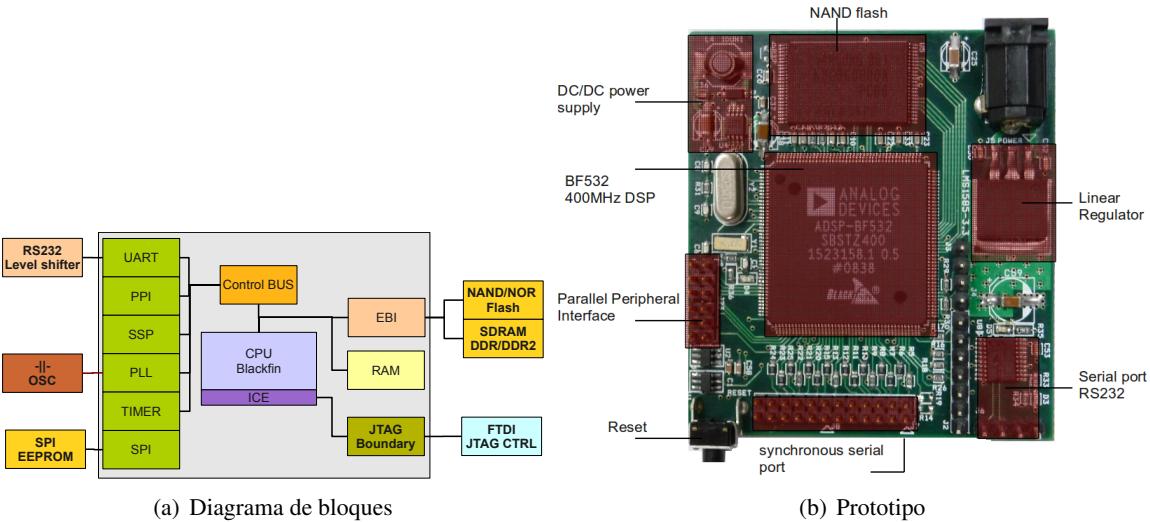
⁴Back Electro Magnetic Force

⁵<http://playerstage.sourceforge.net/>

⁶<http://dspace.mit.edu/handle/1721.1/41550>

**Figura 3.17.: Plataforma de desarrollo ECBOT**

de procesamiento digital de señales del Departamento de Ingeniería Eléctrica y Electrónica de la Universidad Nacional de Colombia.

**Figura 3.18.: Plataforma de desarrollo ECB_BF532**

La plataforma SIE[46], permite el desarrollo de aplicaciones gráficas; esta compuesta por un procesador MIPS fabricado por la compañía china Ingenic y por una FPGA (Spartan 3E de Xilinx) lo que posibilita la implementación de tareas hardware y la creación de controladores específicos en el sistema operativo Linux. En la actualidad esta pataforma esta siendo utilizada en tres de los cuatro cursos de la línea de electrónica digital en el Departamento de Ingeniería Eléctrica y Electrónica de la Universidad Nacional de Colombia. En el capítulo 4 se realizará una detallada explicación del uso de SIE en la enseñanza de diseño de sistemas embebidos.

La plataforma STAMP (se encuentra en proceso de pruebas al momento de escribir este documento) es el resultado del interés por el estudio del sistema operativo Android; debido a los

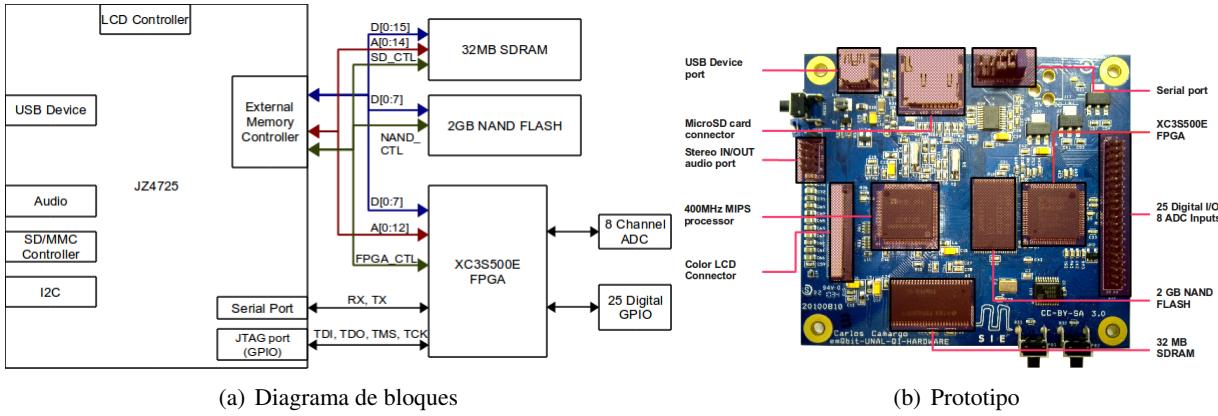


Figura 3.19.: Plataforma de desarrollo SIE

requerimientos de memoria de este sistema operativo, las plataformas anteriormente diseñadas no pueden ejecutar aplicaciones Android. Por este motivo se utilizó el SoC de Freescale iMX233 que incluye el controlador de memorias DDR (las que son más económicas y de mayor densidad que las SDRAM utilizadas en las plataformas existentes); adicionalmente, este SoC permite inicializar desde una memoria SD, lo que elimina la necesidad de utilizar memorias NAND, reduciendo de forma considerable el costo. En la figura 3.20 se puede observar el diagrama de bloques y una foto del prototipo de esta plataforma.

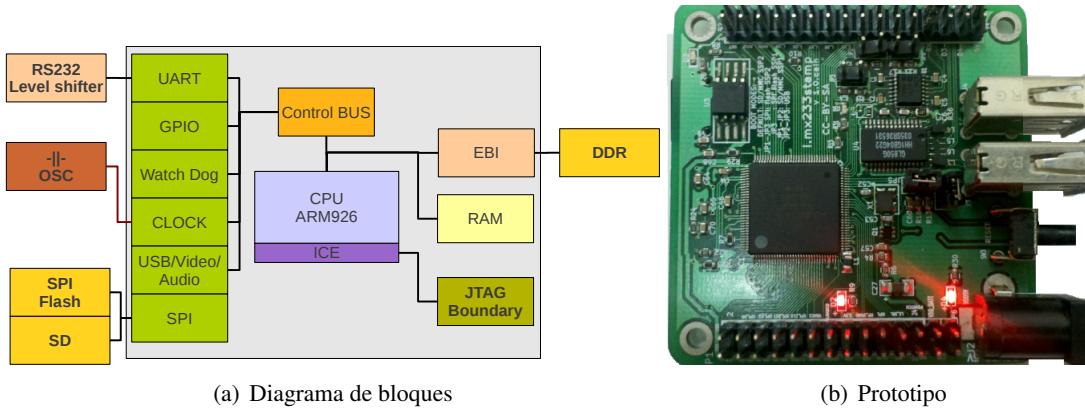


Figura 3.20.: Plataforma de desarrollo STAMP

Técnicas de fabricación

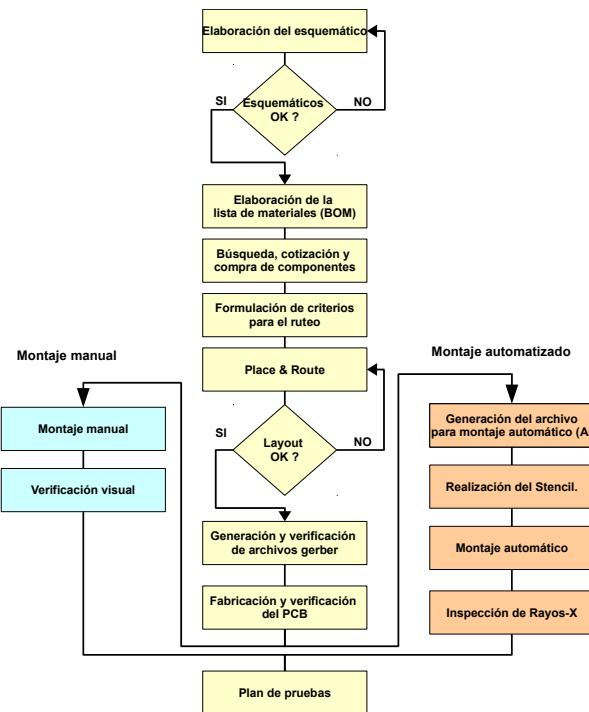
Durante el proceso de concepción, diseño, implementación y operación de estas plataformas se trabajó con diferentes proveedores de bienes y servicios nacionales y extranjeros, y se utilizaron diferentes técnicas para su construcción y montaje; en la tabla 3.3 se resumen las principales características de cada plataforma: CPU utilizada, número de capas del PCB, tipo de montaje: manual o automático, cantidad de unidades producidas, sistema operativo (OS) estudiado y adaptado totalmente a la plataforma y finalmente las universidades que utilizan estas plataformas.

Tabla 3.3.: Características de las plataformas de desarrollo concebidas, diseñadas e implementadas

Plataforma	CPU	Capas	Montaje	Cant.	OS	Usuario
ECB_ARM7	ARM7,33M	2	local Manual.	2	eCos	UN
UN_UIS_XPORT	ARM7,50M	2	local Manual.	2	eCos	UN, UIS
ECB_AT91_V1	ARM920,180M	2	local Manual/Autom.	100	Linux	UN, UIS, ULA, ENAP, UDFJC, USTA
ECB_AT91_V2	ARM920 180M	4	local Manual.	30	Linux	UN, UIS, ULA, ENAP, UDFJC
ECBOT	ARM920 180M	4	local Manual.	20	Linux	UN, UIS
ECB_BF532	Blackfin 400M	4	local Manual.	5	uCLinux	UN
SIE	MIPS32 300M	2	externo Autom.	80	Linux	UN, UIS, ULA, ECI
STAMP	ARM926 454M	2	local Manual.	2	Android	UN

Proceso de diseño y fabricación

La figura 3.21 muestra los pasos que se siguieron en la elaboración de las plataformas de desarrollo; esta metodología puede ser utilizada para la realización de cualquier nivel de producción; el montaje de prototipos puede realizarse de forma manual o automática, dependiendo de los recursos económicos, de la disponibilidad de equipo y de personal especializado.

**Figura 3.21.: Flujo de diseño para las placas de circuito impreso**

En esta etapa se adquirieron las siguientes habilidades:

- Concepción, diseño, implementación y operación de tarjetas electrónicas que utilizan SoC de 32 bits.

Técnicas de diseño de placas de circuito impreso.

Técnicas de montaje manual y automático.

Metodología para el diseño e implementación de tarjetas electrónicas.

Diseño de protocolo de pruebas.

- Adaptación del sistema operativo Linux a una nueva plataforma.

Aplicaciones académicas realizadas

Durante esta etapa (que se realiza de forma continua cada vez que se desea incluir un nuevo tema de estudio) se desarrollaron una serie de trabajos de grado a nivel de pregrado en las universidades Nacional de Colombia, de los Andes e Industrial de Santander y se diseñó un nuevo curso llamado *Sistemas Embebidos* que fue dictado en varias ocasiones como electiva en la Universidad de los Andes y es un curso optativo en la Universidad Nacional de Colombia. Se ayudó a dos egresados de la carrera de Ingeniería Electrónica en la creación de la empresa *EM Electrónica*, proporcionando la información necesaria para la construcción y modificación de tarjetas de desarrollo de dispositivos lógicos programables (CPLDs y FPGAs), las que fueron utilizadas en los cursos de la línea de electrónica digital en la Universidad Nacional. Como resultado de la utilización de estas plataformas se eliminó el uso de componentes discretos (familias lógicas TTL y CMOS 40XX y 74XX) y se comenzó el uso de lenguajes de descripción de hardware desde el primer curso de la línea. A continuación se listan los trabajos realizados durante esta etapa:

- Trabajos de grado: UIS: [39], UNAL:[47] [48] [49] [50] [51]
- Tesis de Maestría UN: [5] [52]
- Artículos [53] [54] [55] [56] [57] [58] [43] [46] [59] [60] [61] [62] [63] [44]

Todos estos productos académicos tienen como característica el desarrollo de placas de circuito impreso y la utilización de los conocimientos adquiridos durante este estudio. Inicialmente, las placas de circuito impreso fueron diseñadas con herramientas comerciales debido a la falta de una herramienta abierta con las características necesarias para realizar el proceso completo. En los últimos dos años el desarrollo de la herramienta *Kicad* permitió la eliminación de estas herramientas propietarias; a partir de ese momento solo se utilizaron herramientas abiertas en todo el flujo de diseño.

Metodología para la apropiación de conocimiento y generación de nuevos productos

En la figura 3.22 se resumen los pasos que se siguieron con las diferentes arquitecturas (CPUs) estudiadas; estas actividades conforman una metodología que minimiza el costo de inversión al utilizar software libre y productos de consumo masivo; adicionalmente, puede ser aplicada a cualquier dispositivo comercial y la curva de aprendizaje es tal que los pasos en las últimas CPUs estudiadas se realizaron en menos de un mes, mientras que la primera tomó cerca de 1 año.

Discusión

La fabricación de placas de circuito impreso a la medida permite reducir el costo final de un dispositivo digital; en su proceso de diseño y fabricación intervienen diferentes personas con diversos niveles de formación, lo que la hace muy atractiva como actividad generadora de empleo. La situación actual de las empresas manufactureras del país (solo 2) no permite la producción a grandes escalas

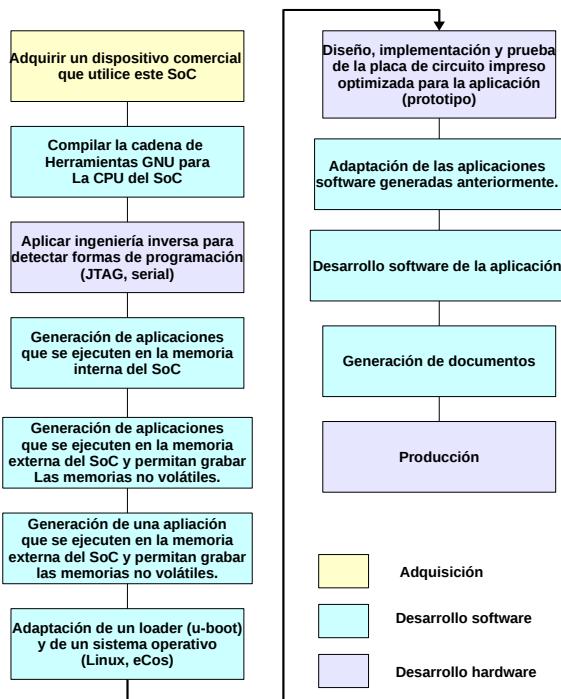


Figura 3.22.: Flujo de actividades para la apropiación de conocimiento

ya que en el país no se cuenta con empresas que suministren componentes electrónicos a precios competitivos (los impuestos locales producen un sobre-costo de cerca del 26 %). En el país se pueden construir prototipos de muy buena calidad, por lo que la mejor opción es la fabricación local del prototipo y la producción en masa en el exterior. Esto mientras se crea en el país la demanda suficiente para poder realizar el proceso completo de forma local.

3.2.5. Aplicación

En esta etapa se aplican los conocimientos adquiridos previamente a la academia y a la industria, con el fin de evaluar su impacto. En la academia se utilizó la posición del investigador principal de este trabajo como docente del departamento de ingeniería eléctrica y electrónica de la Universidad Nacional de Colombia, para modificar las asignaturas del área de electrónica digital incorporando los temas recientemente absorbidos y asimilados. En el sector industrial , se formó una empresa de base tecnológica y se brindó asesoría en la adecuación de las plataformas *hardware copyleft* para convertirlos en productos comerciales.

En la academia

Se diseñó un nuevo programa académico que sigue los lineamientos de la iniciativa CDIO; en el capítulo 4 se detallan las características de este programa, el cual fue implementado en el Departamento de ingeniería eléctrica y electrónica de la Universidad Nacional de Colombia en las asignaturas del área de electrónica digital con el fin de formar profesionales con las habilidades necesarias para desarrollar productos comercializables basados en esta tecnología, el proceso de aplicación duró dos años y actualmente es el programa oficial en dicha institución; los cambios

generados debido a su aplicación son sorprendentes y se evidencian en la calidad de los trabajos de grado donde se pasó del uso de tabletas experimentales o proto-boards, placas de prototipado o stripboards y micro-controladores de 8 bits al uso de placas de circuito impreso diseñadas específicamente para la aplicación y el uso de SoCs de 32 bits y sistemas operativos. Estudiantes encuestados⁷ sobre la pertinencia de esta nueva metodología de enseñanza manifiestan que concuerda con sus expectativas y se les proporcionan herramientas que pueden utilizar en el mundo real, motivándolos a crear nuevos productos que den solución a problemas reales ya sea como empleados en empresas ya constituidas o en nuevas empresas creadas por ellos.

En la industria

En el campo comercial, se trabajó con la empresa de base tecnológica emQbit, uno de los objetivos de trabajar con esta empresa era evaluar el impacto del uso de esta tecnología en la industria Colombiana; adicionalmente, emQbit proporcionó información sobre el estado de la industria electrónica en el país, ayudando a detectar los obstáculos que enfrentan las nuevas empresas en su ejercicio de suministrar soluciones a problemas locales. La mayor parte de sus productos utilizan como base las plataformas *hardware copyleft* diseñadas y distribuidas bajo el esquema de licencias CC BY-SA; el proceso de modificación se facilitó gracias a la disponibilidad de los archivos de diseño y al código fuente de las aplicaciones necesarias para la operación básica de las mismas. Estos productos están siendo documentados y podrán ser estudiados/utilizados/modificados por quien esté interesado. Es muy importante mencionar que (hasta donde llegan los conocimientos del autor) emQbit es la única empresa colombiana que realiza el proceso completo: diseño, construcción y programación de dispositivos electrónicos que utilizan tecnologías modernas (procesadores de 32 bits, sistema operativo Linux). En el desarrollo de sus actividades emQbit encontró las siguientes dificultades:

- Falta de personal con experiencia en desarrollo hardware y en creación de aplicaciones: Es notoria la carencia de habilidades relacionadas con el diseño y montaje de sistemas digitales en la mayoría de los profesionales entrevistados por esta empresa para ser parte de su grupo de desarrolladores; en general, desconocen el proceso de fabricación y las herramientas necesarias para diseñar una placa de circuito impreso, no son capaces de leer esquemáticos ni de buscar componentes adecuados para realizar una función determinada (lo que refleja la dependencia de la academia hacia las plataformas de desarrollo comerciales). Por otro lado, se detectan fallas de concepto o desconocimiento de partes del proceso de diseño, originado en algunos casos por el uso de herramientas propietarias que automatizan etapas de configuración de las plataformas.
- Escasa oferta de productos, bienes y servicios relacionados con la manufactura de sistemas digitales: Como se mencionó anteriormente, en Colombia solo existen 2 empresas con la capacidad de proporcionar los servicios necesarios para la producción, montaje automático y pruebas de placas de circuito impreso utilizando componentes modernos y con la calidad de las empresas asiáticas; desafortunadamente, sus costos son muy elevados en comparación con los de las industrias manufactureras asiáticas; esto se debe a la baja demanda de la industria colombiana y a los altos impuestos que deben pagar los dispositivos semiconductores y demás insumos del proceso (26 % o más). A pesar de estos inconvenientes, la fabricación de prototipos

⁷Encuestas realizadas por este estudio y la información suministrada por el sistema de evaluación de cursos de la Universidad Nacional

a nivel local es una buena opción desde el punto de vista económico, ya que la diferencia en costos no es muy grande, y es posible resolver dudas o problemas de forma ágil gracias a que se trabaja en la misma zona horaria.

- Desconfianza de la industria local por los productos del país: Los problemas descritos anteriormente generan un clima de desconfianza hacia los productos nacionales; ya que gran parte de ellos utilizan tecnologías obsoletas con procesos de fabricación de mala calidad, no cumplen con normas internacionales y no fueron sometidos a rigurosos procesos de pruebas. La desconfianza se presenta por la ausencia de productos locales, por el desconocimiento de los planes académicos de las universidades y por la falta de profesionales con la capacidad de diseñar y construir aplicaciones “a la medida” de las necesidades.
- Problemas relacionados con la financiación para desarrollo de nuevos productos: Es muy difícil encontrar entidades que financien el desarrollo de nuevos productos, la mayoría busca dispositivos terminados en fase de producción y no se encuentran interesados en invertir tiempo y dinero en nuevos desarrollos, esto es una consecuencia directa de la pérdida de capacidad de innovación en las empresas colombianas, donde no existen o fueron eliminados los departamentos de investigación y desarrollo (I+D).
- Competencia en desigualdad de condiciones con productos importados: Los impuestos a las importaciones de insumos para la fabricación de dispositivos digitales (más del 26 %), la poca oferta de bienes y servicios relacionados con la manufactura y la falta de políticas gubernamentales que protejan la industria electrónica nacional han permitido que los productos tecnológicos de los países asiáticos desplacen la producción local. Es importante que se formulen políticas que protejan los productos nacionales, sin provocar un desabastecimiento de productos que aún no se generan localmente, para esto es muy importante identificar los productos que son fabricados localmente y agrupar de forma adecuada a las empresas en la clasificación CODIGO INDUSTRIAL INTERNACIONAL UNIFORME (CIIU) en el registro de cámara y comercio.

Discusión

La adaptación de esta metodología al entorno académico reveló la deficiencia de los programas académicos del área de electrónica digital en la Universidad Nacional de Colombia, vale la pena mencionar que esta universidad ha ocupado los primeros puestos en los exámenes de estado *SABER PRO* (antes ECAES), por lo que es de esperarse que algunos programas académicos de otras universidades presenten problemas similares. La falla más notoria que se encontró en este proceso fue la ausencia de una metodología de diseño unificada durante la enseñanza de las asignaturas que componen el área; cada curso tenía su metodología y no existía una relación entre los contenidos de las asignaturas, lo que dejaba en los estudiantes la sensación de que se trataban de temas diferentes. Por otro lado, el uso de tecnologías obsoletas desviaba el enfoque del estudio a tediosas tareas manuales de minimización e implementación, restando importancia a la concepción y validación del diseño. Errores conceptuales en el diseño de los cursos hicieron que un lenguaje de descripción de hardware como el Verilog o VHDL fuera el centro de atención ignorando por completo temas relacionados con la arquitectura de computadores. Al final de estos dos cursos, los estudiantes no veían la utilidad práctica de los conocimientos adquiridos ya que nunca fueron enfrentados a la solución de problemas reales; esto, unido a la falta de herramientas de desarrollo hicieron que muchos

de ellos no consideraran la electrónica digital para dar solución de problemas. El último curso del área (micro-controladores) proporcionaba herramientas que permitían dar solución a problemas reales, por este motivo, los estudiantes opinaban que era la única útil de la línea; sin embargo, la falta de bases en arquitectura de computadores hacía que se pensara en el micro-controlador como una caja negra sin entender completamente su arquitectura, lo que limitaba el entendimiento del proceso de diseño.

Los cambios introducidos en estas asignaturas utilizan una única metodología de diseño, los problemas son atacados de la misma forma pero con herramientas cada vez más complejas pero más sencillas de utilizar. El uso de plataformas de desarrollo abiertas permite estudiar su arquitectura y entender completamente los flujos de diseño; adicionalmente, pueden ser utilizadas como punto de partida de desarrollos comerciales. Esto unido a la experiencia de diseño de circuitos impresos en 3 asignaturas, hizo cambiar completamente la mentalidad de los estudiantes sobre el uso de estas tecnologías, haciendo que realizaran búsquedas de problemas que puedan ser solucionados con estas herramientas, lo que constituye la base de la innovación. Todo esto representa una transferencia exitosa de esta investigación hacia la academia ya que se modificaron los hábitos existentes y se utilizaron los conocimientos generados por esta investigación; adicionalmente, se demostró la importancia del uso de esta tecnología.

Por otro lado, desde el punto de vista comercial, se comprobó que la universidad puede incentivar la creación de empresa, siempre y cuando se proporcionen herramientas para que puedan crear productos novedosos, y se brinde soporte técnico para su utilización. Sin embargo, los problemas mencionados anteriormente dificultan su operación, lo que permite concluir que es necesario una alianza entre este tipo de empresas, las universidades, los generadores de políticas gubernamentales y los *ángeles inversores* para financiar el desarrollo de productos comercializables. Así mismo, se comprobó que es posible que empresas de base tecnológica colombianas utilicen tecnologías modernas para competir con productos provenientes del exterior. Desde el punto de vista de la transferencia tecnológica se puede afirmar que en este caso, se realizó una transferencia exitosa hacia la empresa, ya que los conocimientos fueron absorbidos, asimilados y se generaron nuevos productos que fueron utilizados para dar solución a problemas locales.

3.2.6. Difusión y Desarrollo

La tarea más importante de esta etapa, es la creación de una comunidad que sea consciente de la importancia de la tecnología y utilice los conocimientos generados en el proceso (recurso, bien común), proporcione nuevo conocimiento que haga parte de este recurso; depure las herramientas de difusión y el contenido del mismo; y ayude a crear conciencia en la sociedad de la importancia de esta tecnología en el desarrollo tecnológico del país. Para esto, se deben proporcionar mecanismos que permitan la realización de esas tareas así como estrategias para la vinculación de nuevos miembros. En el capítulo 5 se muestra como está conformada esta comunidad, sus reglas y recursos.

Herramientas para el manejo, almacenamiento, depuración y aumento del recurso

El conocimiento absorbido y generado durante este estudio se encuentra representado en una metodología de diseño e implementación de sistemas digitales que utiliza herramientas abiertas y en una metodología para la transferencia tecnológica y de conocimiento informal basada en la libertad del conocimiento y el uso de este como un bien público. Las plataformas hardware desarrolladas en este estudio y que se encuentran disponibles bajo el esquema de licencias Creative Commons CC-BY-SA representan la aplicación del conocimiento acumulado en el diseño y fabricación de

sistemas embebidos y pueden ser utilizadas por cualquier interesado para mejorar sus habilidades en el diseño, implementación, fabricación o programación de este tipo de dispositivos. El recurso inicial esta compuesto por la información necesaria para entender, programar, reproducir, y modificar estas plataformas; este será la base de nuevos productos tecnológicos que den solución a problemas del país y ayuden a difundir el uso de esta tecnología en centros de formación y empresas de base tecnológica y de esta forma aumentar la oferta de bienes y servicios relacionados con el diseño de aplicaciones personalizadas que se ajusten exactamente a los requerimientos de la sociedad. Sin embargo, para que esto sea posible, es necesario proporcionar herramientas que permitan la administración y difusión del recurso; por este motivo, se creó el portal abierto *linuxencaja* donde se proporcionan los siguientes servicios:

- Sistema distribuido de control de versiones ⁸: El intercambio de información entre programadores usuarios que se encuentran separados geográficamente puede generar una sobrecarga de mensajes que dificultan el proceso de desarrollo; si no se cuenta con una infraestructura adecuada, se corre el riesgo de perder valiosa información por falta de sincronización entre los desarrolladores. De aquí la importancia de contar con un sistema que permita seguir el avance de un proyecto y los cambios que han sido realizados por otros miembros del equipo, los cuales, pueden estar en cualquier lugar del mundo. Para facilitar esta tarea, se cuenta con los sistemas distribuidos de control de versiones abiertos; siendo los más populares *subversion* (svn) y *git*, en este estudio se utilizó *git* ya que permite el seguimiento total del proyecto indicando el autor y un comentario descriptivo con el aporte. En la figura 3.23 se muestran las funcionalidades de esta aplicación.

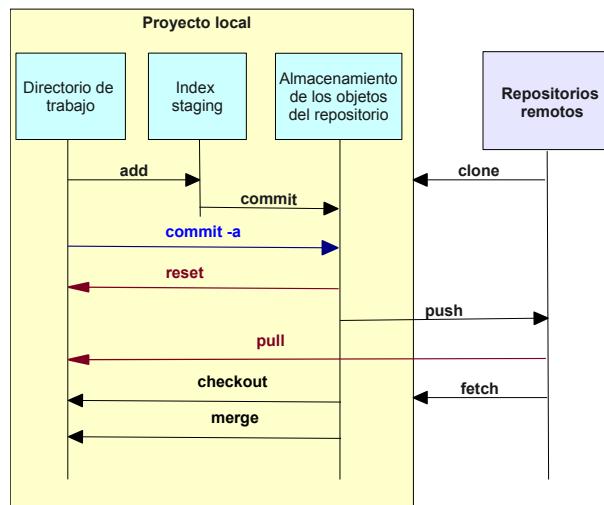


Figura 3.23.: Funcionalidad del sistema distribuido de control de versiones git

El *sitio de trabajo* es donde se realiza el proyecto y contiene archivos que serán o no tenidos en cuenta para el control de revisiones. El índice (*staging area*, se utiliza para preparar los cambios que serán publicados (*commits*) se pueden agregar (*add*) archivos o partes de archivos para al siguiente *commit*. Todos los cambios que se deseen publicar van primero al índice; la

⁸<http://projects.linuxencaja.net/>

acción *commit* aplica los cambios en el índice en el sitio de almacenamiento de los objetos. Este repositorio de objetos puede ser creado localmente y puede ser copiado a (*push*) o desde (*pull*) otro repositorio remoto. De esta forma es posible sincronizarse y realizar modificaciones sobre un repositorio remoto. Si por algún motivo, accidentalmente se borra o se daña el contenido de algunos archivos del repositorio, es posible restaurar el estado del mismo a un punto donde se reviertan los cambios que ocasionaron los daños (*reset*). Si se piensa introducir un cambio que requiera la modificación de un extenso número de objetos es posible crear y trabajar con una nueva rama del proyecto (*checkout* y conservar la rama original (*merge*)). En la actualidad *linuxencaja* cuenta con 5 repositorios: SIE, ECB_AT91, ECBOT, ECB_BF532 y AndroidStamp; cada uno de ellos reúne la información necesaria para entender y reproducir plataformas hardware que utilizan diferentes SoC. En ellos se pueden encontrar las siguientes aplicaciones:

- Archivos de diseño: Archivos necesarios para reproducir y modificar el PCB.
- Hello World: Esta aplicación implementa un parpadeo (blink) en un LED conectado a un pin de entrada/salida de propósito general, con este sencillo ejemplo se indica como se puede programar la memoria RAM interna del SoC, se proporciona un ejemplo funcional para el desarrollo de aplicaciones sin sistema operativo.
- Loader: Aplicación que permite escribir en las memorias no volátiles de la plataforma.
- Kernel: Parches que dan soporte a la plataforma en el kernel de Linux.
- File system: Sistema de archivos con las utilidades necesarias para la ejecución de aplicaciones.
- Drivers: Módulos del kernel que muestran como implementar drivers de Linux para control de periféricos

blink: La misma funcionalidad que *Hello World* pero como un driver de Linux.

irq: Ejemplo de un módulo del kernel con interrupciones y con un programa en espacio de usuario que controla este módulo.

- Examples: Proyectos académicos destacados que utilizan la plataforma y que pueden ser utilizados como referencia para desarrollo de aplicaciones.

- Wiki ⁹: Las wikis son sitios web con facilidades de edición de páginas web utilizando un lenguaje propio (wikitexto). En este estudio se utilizó el software *Mediawiki* que permite la creación instantánea de páginas web sin la necesidad de permisos especiales por parte del administrador de la máquina; Mediawiki (y demás aplicaciones similares) almacena en una base de datos el contenido de las páginas en texto plano, (este formato se suministra al usuario en el momento de la edición) y genera en tiempo real el código HyperText Markup Language (HTML) a partir de este texto plano cada vez que se solicita una página; una característica especial es que las páginas pueden ser editadas por múltiples voluntarios; adicionalmente, se dispone de un historial de cambios que le permite al administrador restaurar el contenido en un eventual daño voluntario o involuntario.

Con esta herramienta es posible que usuarios del recurso documenten nuevos proyectos o mejoren la documentación de los ya existentes; en la actualidad, esta wiki está siendo utilizada por los estudiantes del área de electrónica digital del Departamento de Ingeniería Eléctrica y

⁹<http://wiki.linuxencaja.net/>

Electrónica de la Universidad Nacional de Colombia para documentar el proceso de diseño de los proyectos de estas asignaturas; los mejores proyectos son conservados y etiquetados como proyectos de referencia y se encuentran disponibles por futuros usuarios que pueden ser industriales o estudiantes de próximos semestres; generando de esta forma un banco de proyectos que puede ser utilizado como fuente de información para mejorar habilidades o para ayuda en el proceso de diseño de nuevos proyectos/productos.

- Lista de correo¹⁰: Contar con un canal de comunicación directo entre los diseñadores y usuarios con más experiencia, permite resolver dudas en corto tiempo, estas respuestas constituyen una fuente de documentación muy importante y debe ser almacenadas para crear un banco de preguntas/respuestas que ayuden a usuarios con las mismas inquietudes. Por esta razón, se creó una cuenta de correo que permite este intercambio de información y puede ser utilizado por los miembros de un determinado proyecto para discutir temas propios del desarrollo del mismo. La metodología utilizada por un determinado grupo quedará registrada en esta lista y puede ser utilizada como referencia de ese y otros proyectos; adicionalmente, estas listas de correo permiten realizar contactos académicos y laborales entre sus miembros.

Estrategia de difusión

Esta metodología de transferencia tecnológica y de conocimientos no cuenta con el apoyo financiero necesario para realizar cursos de capacitación a empresas de base tecnológica destinados a transferir las metodologías de diseño y fabricación propuestas, por este motivo se utilizará una difusión basada en relaciones locales entre centros de formación, empresas, y entidades del estado que financien proyectos de transferencia tecnológica.

- Relaciones con grupos de investigación de la misma universidad: En la Universidad Nacional (sede Bogotá) se importaban todas las placas electrónicas necesarias para el desarrollo de proyectos de investigación en diferentes departamentos de la facultad de ingeniería, muchas de ellas con grandes sobre-costos originados por la importación. Existen áreas de trabajo que pueden ser estudiadas por grupos interdisciplinarios como robótica, telemedicina, telemetría o control. Bajo esta investigación se desarrollaron aplicaciones con los departamentos de sistemas y mecatrónica para desarrollar plataformas robóticas [45] [45] que pudieran ser utilizadas para validar los modelos computacionales y algoritmos propuestos y con el centro de telemedicina de la Universidad Nacional, con quien se diseñó y construyó un monitor remoto de signos vitales [5].
- Relaciones con centros de formación: Aprovechando la posición y reconocimiento de la Universidad Nacional de Colombia y los contactos existentes con los centros de formación que participaron en este estudio. Se difundieron los conocimientos obtenidos y generados en este estudio y se proporcionó la información necesaria para que sean incorporados en los programas académicos de las asignaturas relacionadas con el diseño digital en estos centros de formación. Con esto se busca aumentar el número de personas que utilicen y contribuyan al crecimiento del recurso público. Adicionalmente, el programa académico (ver capítulo ch:education) creado e implementado en la Universidad Nacional de Colombia se encuentra disponible para cualquier centro de educación que quiera implementarlo parcial o totalmente.

¹⁰<http://groups.google.com/group/linuxencaja>

- Relaciones con la industria: Para trabajar con empresas de base tecnológica se creó un proyecto¹¹ que busca difundir el uso de esta tecnología en la creación de nuevos productos que den solución a un problema local; este proyecto realizará un proceso para seleccionar 2 empresas de base tecnológica en ciudades donde existan centros de formación que utilicen este recurso; dichas empresas formularán un proyecto que satisfaga una necesidad local; estudiantes de los centros de formación (como trabajo de grado) ayudarán a estas empresas en el diseño y construcción de un dispositivo que implemente la solución propuesta por ellas. Todos los proyectos deben ser de tal naturaleza que puedan ser implementados en un dispositivo electrónico que utilice la tecnología utilizada en este estudio; esto con el fin de implementar todos los proyectos con una plataforma electrónica base que permita adaptarse a las diferentes aplicaciones; esta *plataforma base* será diseñada por todos los participantes y se buscará reunir la máxima funcionalidad con el menor precio posible; utilizando los recursos suministrados por el portal *linuxencaja* se coordinará el trabajo entre los diferentes grupos de trabajo; una vez diseñada esta plataforma base se fabricaran tantos prototipos como grupos de trabajo para determinar su correcto funcionamiento, una vez corregidos los posibles errores y comprobado el correcto funcionamiento de todos los componentes; se realizará la producción de 100 unidades con el fin de adquirir experiencia en la producción de estos sistemas utilizando la industria manufacturera asiática; después de realizar la pruebas necesarias, cada grupo de trabajo diseñará y fabricará una tarjeta hija que proporcione la funcionalidad requerida por cada proyecto. Todo el proceso de diseño, los criterios de desarrollo, las discusiones para determinar las especificaciones tanto de la plataforma base como de las plataformas hijas se documentará en el portal *linuxencaja* para permitir la reproducción de este curso.

¹¹En la actualidad se están buscando fuentes de financiación para su ejecución

Tabla 3.4.: Comparación de precios entre la transferencia ofrecida por Corea y el costo utilizando los conocimientos generados en este trabajo

Item	KETI-ETRI	Este trabajo
Equipos de cómputo (25) con licencias de:	115.000	50.000 ¹²
<ul style="list-style-type: none"> ▪ Windows, VMWARE,¹² ▪ Android SDK, Apache 		
Plataforma de desarrollo (15)	60.000	4.500 ¹³
<ul style="list-style-type: none"> ▪ Procesador ARM, 512MB DDR, MicroSD ▪ LCD 800 x 480, touch screen ▪ USB 2.0, WiFi, Camera. ▪ Bluetooth. Android OS. 		
Software para simulación:	132.000	0 ¹⁴
<ul style="list-style-type: none"> ▪ Qemu con soporte para OpenMoko 		
Depuración hardware:	180.000	2.000 ¹⁵
<ul style="list-style-type: none"> ▪ Trace 32 de Laterbach (15) 		
Herramientas CAD:	180.000	0 ¹⁶
<ul style="list-style-type: none"> ▪ PADS de Cadence (15) 		
Plataforma hardware (15):	120.000	50.000
<ul style="list-style-type: none"> ▪ Especificaciones de la plataforma de desarrollo ▪ Archivos para reproducir el hardware: <ul style="list-style-type: none"> • Esquemáticos. • Archivos Gerber. ▪ Código fuente: <ul style="list-style-type: none"> • bootloader: u-boot. • Kernel de Linux. • Android OS. ▪ documentación <ul style="list-style-type: none"> • Guía de usuario. • Port de Android. 	<ul style="list-style-type: none"> ▪ Archivos para Modificar la plataforma¹⁷ ▪ Código fuente: <ul style="list-style-type: none"> • u-boot • Kernel de Linux. • Android OS. ▪ Documentación: <ul style="list-style-type: none"> • Tutoriales. • Proceso completo de diseño. • Cursos en Línea. • Listas de discusión. 	

En la actualidad se está evaluando la posibilidad de realizar una transferencia tecnológica entre el estado Colombiano y dos centros de desarrollo coreanos Electronics and Telecommunications Research Institute (ETRI) y Korea Electronics Technology Institute (KETI) en el área de aplicación de televisión digital. Como parte de la transferencia, se contempla transferir conocimientos necesarios para realizar la fabricación de una plataforma de desarrollo que pueda ser utilizada en este tipo de aplicaciones. La tabla 3.4 muestra los costos de esta transferencia (en miles de dólares) y el ahorro

que se puede realizar gracias a los resultados de este trabajo.

Como podemos observar en la tabla 3.4, los costos de esta parte de la transferencia¹⁹ son muy elevados (787.000 USD), frente al costo que tendría realizar las mismas actividades con los conocimientos adquiridos en este trabajo (106.500).

La transferencia que ofrece Corea no tiene en cuenta la capacitación necesaria para que el país pueda absorber y asimilar estos conocimientos de tal forma que pueda generar sus propios productos lo que constituye el éxito de la transferencia. Si este proyecto se llegara a formalizar se crearía una dependencia del país hacia a tecnología ofrecida por Corea. El uso de herramientas comerciales en la etapa de diseño hace que los costos de este tipo de productos aumente de forma considerable, en esta comparación las herramientas de diseño de circuitos impresos, el sistema operativo y un emulador de Linux cuestan 245.000 USD, y son reemplazadas sin costo alguno por las herramientas utilizadas en este trabajo.

¹²Se eliminan las licencias de Windows y VMWARE al utilizar Linux, Android SDK y Apache son gratuitos

¹³Existen dispositivos comerciales con estas características que pueden ser utilizados en el desarrollo utilizando ingeniería inversa, como se hizo en este trabajo

¹⁴Qemu es gratuito, el proyecto del celular abierto OpenMoko distribuye de forma gratuita su simulador

¹⁵El proyecto OpenOCD permite realizar este tipo de pruebas utilizando una interfaz con el puerto JTAG de los procesadores; en este trabajo se desarrolló una interfaz USB-JTAG que permite hacer esto de forma económica.

¹⁶Este trabajo proporciona una metodología para utilizar herramientas abiertas en todo el proceso de fabricación de PCBs

¹⁷Plataforma *hardware copyleft*

¹⁹El costo total de la transferencia cuesta 2 millones de USD

4. Plan de Estudios Para la Enseñanza/Aprendizaje de Sistemas Embebidos

4.1. Introducción

La disponibilidad de personal calificado que absorba, asimile y aplique los conocimientos asociados a una tecnología es, el punto más importante en el proceso de transferencia tecnológica propuesto en este trabajo; como se mencionó anteriormente, los canales tradicionales de transferencia han demostrado ser poco efectivos para difundir estos conocimientos en el área bajo estudio; la hipótesis central que maneja esta investigación es que la causa principal del atraso de la industria electrónica (relacionada con el diseño digital) es la poca oferta de personal calificado que genere un cambio que permita pasar de importadores a generadores de soluciones que utilizan los últimos avances tecnológicos en la industria electrónica mundial.

Es importante que los estudiantes utilicen herramientas adecuadas cuando se enfrentan a la implementación física. Hasta hace poco en el Departamento de Ingeniería Eléctrica y Electrónica (DIEE) se encontraban trabajos académicos realizados por estudiantes de pregrado y posgrado que utilizaban placas de prototipos; se proporcionaban soluciones basadas en herramientas comerciales las cuales eran conseguidas de forma ilegal o se utilizaban demostraciones con funcionalidad limitada. Este tipo de soluciones están muy lejos de una solución comercial real, primero por que no proporciona una plataforma física robusta y segundo porque es necesario adquirir software muy costoso. Adicionalmente, no es ético utilizar herramientas comerciales en la enseñanza, si no se cuentan con las licencias necesarias; es muy común ignorar este tipo de violaciones a los derechos de autor cuando se trabaja en aplicaciones académicas; por esta razón, una de las premisas de esta metodología de enseñanza/aprendizaje es el uso de herramientas abiertas que produzcan resultados comparables al de las herramientas comerciales.

El plan de estudios propuesto en este capítulo adopta los conocimientos generados en el proceso de transferencia descrito en el capítulo 3, lo que lo convierte en una herramienta de difusión hacia la academia, con esto se espera que los estudiantes tomen conciencia de la importancia del uso de esta tecnología. El contenido y la metodología del plan de estudios se ajustará a la iniciativa CDIO. El Departamento de Ingeniería Eléctrica y Electrónica de la Universidad Nacional de Colombia está realizando el proceso de adaptar la Iniciativa CDIO a las asignaturas de sus dos carreras (eléctrica y electrónica); la mayoría de las asignaturas que componen estos programas no contemplan actividades que ayuden a mejorar las habilidades en la implementación de sistemas, de aquí la importancia de incluirlos como parte de esta propuesta.

4.1.1. La Iniciativa CDIO

La iniciativa CDIO¹ ha sido desarrollada por el Massachusetts Institute of Technology (MIT) con ayuda de académicos, industriales, ingenieros y estudiantes [64] como respuesta a los diferentes caminos que están tomando la educación de la ingeniería y las demandas del mundo real². Esta iniciativa ha sido adoptada por un creciente número de instituciones académicas; hacer parte de este esfuerzo mundial ayuda a mantener los planes académicos actualizados con los cambios que se realizan en países más industrializados. En este capítulo se mostrará cómo esta iniciativa se adapta perfectamente a la metodología propuesta en este trabajo ya que adiciona dos componentes importantes para la aplicación de la tecnología en la creación de nuevos productos: la implementación y la operación.

La iniciativa CDIO se basa en la suposición de que los egresados de los centros de formación en ingeniería deben ser capaces de: **Concebir, Diseñar, Implementar y Operar** sistemas funcionales en el mundo real. Como se mencionó anteriormente, en Colombia, una parte importante de los centros de formación solo tienen en cuenta la concepción y el diseño, descuidando la implementación y la operación. Lo que impide que se generen habilidades necesarias para establecer una estrecha relación con la industria, la cual, requiere productos que pueda comercializar o den soluciones a sus necesidades. La frase *en el mundo real* resalta la importancia de trabajar en la solución de problemas que pueden encontrarse en el ejercicio profesional, lo que es muy difícil de determinar cuando los docentes no tienen un contacto frecuente con él. La iniciativa CDIO se enfoca en preparar a los estudiantes con los conocimientos habilidades y aptitudes para ser ingenieros líder; y sus principales objetivos son [64]:

- Educar a los estudiantes para dominar un conocimiento más profundo de los fundamentos técnicos.
- Educar a los ingenieros para liderar la creación y operación de nuevos productos y sistemas.
- Educar futuros investigadores para que conozcan la importancia estratégica y el valor de su trabajo.

Estos objetivos se adaptan a los requerimientos que se exige a la plataforma tecnológica de un país para que pueda realizar una adecuada absorción del conocimiento transferido para la creación de nuevos productos adaptados a las necesidades locales. Las premisas que capturan la visión, objetivos y fundamentos pedagógicos de la iniciativa son:

- Es posible cumplir las necesidades propias de la profesión mientras se realiza el proceso de concebir, diseñar, implementar y operar sistemas en el contexto de los sistemas de ingeniería.
- Los resultados de la formación deben ser fijados por los sectores interesados (academia, industria, gobierno) y deben formar una secuencia de experiencias de aprendizaje, algunas de las cuales son experimentales; es decir, deben enfrentar a los estudiantes a situaciones que encontrarán en el ejercicio de su profesión.
- La adecuada construcción de esta cadena de actividades tendrán un doble impacto en la formación de los estudiantes: por un lado facilitará el aprendizaje de habilidades críticas e

¹<http://www.cdio.org>

²lo que se aplica perfectamente al estado de la industria electrónica en Colombia

inter-personales y fortalecerá las habilidades de construcción de sistemas, productos y procesos, mientras se mejora el aprendizaje de los conceptos fundamentales.

4.1.2. Estructura del Plan de Estudios CDIO

La figura 4.1 muestra los bloques constructores del plan de estudios CDIO. En el primer nivel se puede observar que todo individuo interesado en obtener habilidades técnicas posee *habilidades personales y profesionales*, las cuales son fundamentales para la práctica. Para ser capaces de desarrollar sistemas complejos en ingeniería, los estudiantes deben dominar los fundamentos del *razonamiento y conocimiento técnico*; para trabajar en un entorno moderno basado en grupos de trabajo, los estudiantes deben desarrollar *habilidades interpersonales* de comunicación y trabajo en equipo; finalmente, para ser capaz de crear y operar productos y sistemas, un estudiante debe entender el concepto de *concebir, diseñar, implementar y operar sistemas en el contexto social y empresarial* [3]



Figura 4.1.: Bloques constructores de conocimiento, habilidades y actitudes necesarias para concebir, diseñar, implementar y operar sistemas en el contexto social y empresarial fuente:[3]

Razonamiento y conocimiento técnico Los componentes del primer nivel *razonamiento y conocimiento técnico* son comunes a los planes de estudio de las ingenierías modernas y son:

- Fundamentos avanzados de ingeniería.
- Fundamentos del núcleo de ingeniería.
- Conocimiento científico.

La razón de poner este bloque constructor en el primer nivel es solo para recordar que el objetivo primordial de cualquier programa de pregrado es el desarrollo de un conocimiento profundo de fundamentos técnicos. En este trabajo no se cambiará este componente ya que para hacerlo es necesario un consenso con las demás carreras de la facultad de ingeniería de la Universidad Nacional, labor que puede tomar varios años.

Habilidades personales, profesionales e interpersonales Los niveles 2 y 3 se centran en las habilidades personales que debe poseer un individuo para que pueda cumplir con el objetivo de la iniciativa CDIO. El nivel 2 está compuesto por:

- Las habilidades profesionales que representan las tres formas de pensar más practicadas por los ingenieros: resolución de problemas; descubrimiento de conocimiento y pensamiento sistemático.

- Actitudes que incluyen integridad y comportamiento profesional así como las necesarias para planear la profesión.

Las habilidades que no hacen parte del contexto profesional ni del inter-personal son llamadas *habilidades y actitudes personales*, incluyen el carácter; iniciativa; perseverancia; formas de pensar más genéricas como pensamiento crítico, creativo; y habilidades propias como curiosidad, aprendizaje continuo y manejo del tiempo.

Las habilidades inter-personales, son un subconjunto de las habilidades personales y se dividen en dos grupos (que se traslanan) llamados: equipo de trabajo y comunicaciones. El equipo de trabajo hace referencia a las habilidades necesarias para formar, operar, fortalecer y liderar un equipo con habilidades específicas de un equipo de trabajo técnico. La comunicación se compone de habilidades para idear estrategias de comunicación y aquellas que utilizan los medios orales, escritos, electrónicos y gráficos, y, en el caso colombiano, el uso del idioma Inglés.

Habilidades CDIO Habilidades necesarias para concebir, diseñar, implementar y operar sistemas en el contexto social y empresarial; estos cuatro componentes son necesarios para que los egresados de las carreras relacionadas con la ingeniería electrónica sean capaces de absorber los conocimientos que las nuevas tecnologías proporcionan, adaptarlos a la situación tecnológica y al contexto social del país para generar productos que resuelvan necesidades locales. Para satisfacer una necesidad de la sociedad es necesario conocer la dinámica empresarial, los principios que la rigen y como se debe actuar en una empresa de cualquier tipo y tamaño.

4.2. Implementación del Plan de Estudios CDIO

La Figura 4.2 muestra los componentes que deben ser especificados para implementar el plan de estudios CDIO al currículo de las asignaturas del área de electrónica digital; en primer lugar se encuentran los resultados esperados del proceso de aprendizaje, esto es, ¿Qué deben saber y qué deben ser capaces de hacer los estudiantes al final del curso? Para contestar a esta pregunta es necesario definir las **habilidades** que serán reforzadas o desarrolladas y los *objetivos* de cada asignatura.



Figura 4.2.: Objetivos, actividades, y evaluación:

Para alcanzar los objetivos definidos en el primer paso, es necesario generar una serie de **actividades** que le permitan al estudiante retener nuevos conocimientos y habilidades y desarrollar las

competencias deseadas; las actividades deben cubrir todas las habilidades que se quieran desarrollar o reforzar. Finalmente, se deben desarrollar métodos de evaluación que permitan conocer el nivel de competencia de los estudiantes, y de esta forma ajustar las actividades para obtener los resultados esperados.

4.2.1. Definición e Identificación de las Habilidades CDIO

El primer paso en la implementación del plan de estudios CDIO es definir e identificar las habilidades requeridas en una determinada área del plan de estudios; en este caso en las asignaturas del área de electrónica digital. En el DICE de la Universidad Nacional de Colombia, el área de electrónica digital está compuesta por tres asignaturas para la carrera de ingeniería electrónica: Electrónica Digital 1, Electrónica Digital 2 y Sistemas Embebidos y por electrónica Digital 1 para ingeniería eléctrica.

Grado de competencia

Para trasladar las habilidades a objetivos de aprendizaje es necesario determinar el grado de competencia que se espera que el profesional adquiera en cada una de las asignaturas; por supuesto, algunas de estas habilidades no pueden obtenerse solo en una asignatura y es necesario que todo el plan académico contribuya a generarla, lo que requiere un consenso del personal académico. Los niveles de competencia seleccionados para indicar el grado en que debe ser apropiada una determinada habilidad son:

- *Introducir (I)*: Introduce pero no evalúa.
- *Enseñar (E)* : Enseña y evalúa.
- *Utilizar (U)*: Utiliza, puede ser evaluado o no.

Aptitudes personales, profesionales y habilidades interpersonales

La tabla 4.1 muestra las habilidades para las *aptitudes personales y profesionales* de las tres asignaturas del área de electrónica digital. En ella se puede observar que existen habilidades comunes a las tres asignaturas en lo relacionado con el planteamiento y resolución de problemas, experimentación y descubrimiento de conocimiento y habilidades y actitudes personales; todas ellas buscan que el estudiante sea capaz de identificar un problema y con base en los conocimientos adquiridos formule hipótesis y modelos que permitan darle solución. Las habilidades interpersonales son tratadas de forma gradual, en el primer curso, se guía en la formación de estas habilidades utilizando ejemplos que ellos utilizarán en los cursos posteriores.

Habilidades CDIO Sistemas en el contexto Empresarial, Social y Ambiental - Innovación

La primera columna de la tabla 4.2 muestra las habilidades C.D.I.O. Sistemas en el contexto Empresarial, Social y Ambiental - Innovación. Este grupo de habilidades son las que marcan diferencia con las otras asignaturas del plan de estudios ya que como se mencionó anteriormente, la mayoría de los cursos no contemplan la implementación de sistemas reales donde se apliquen los conocimientos que se intentan transmitir.

Competencias de las habilidades CDIO nivel 2 y 3			
APTITUDES PERSONALES Y PROFESIONALES	Nivel 1		
	E. Dig1	E. Dig2	Sist. Emb.
Planteamiento y resolución de problemas de ingeniería	EU		
1 Identificación y formulación del problema		EU	
2 Modelamiento		EU	
3 Solución y recomendación		EU	
Experimentación y descubrimiento de conocimiento	U		
4 Formulación de hipótesis		U	
5 Investigación experimental		U	
Pensamiento sistemático	EU		
6 Pensamiento global		U	
7 Surgimiento e interacciones		U	
Habilidades y actitudes personales	U		
8 Pensamiento creativo		IEU	
9 Pensamiento crítico		IEU	
10 Toma de conciencia de conocimientos propios		IEU	
11 Curiosidad y aprendizaje permanente Habilidades y actitudes profesionales	U		
12 Ética profesional, integridad, responsabilidad		U	
13 Comportamiento profesional		U	
39 Confianza y lealtad		IEU	
HABILIDADES INTERPERSONALES	Nivel 1		
	E. Dig1	E. Dig2	Sist. Emb.
Equipo de trabajo	EU		
14 Formar grupos efectivos	EU	U	U
15 Equipo de liderazgo	EU	U	U
40 Equipo Técnico y Multi-disciplinario	EU	U	U
Comunicaciones estructuradas	EU		
16 Estrategia de comunicación	EU	U	U
17 Estructura de la comunicación	EU	U	U
18 Comunicación Escrita	EU	U	U
19 Comunicación Electrónica	EU	U	U
20 Presentación Oral	EU	U	U
Comunicación en Idioma Extranjero	U		
21 Inglés	U		
Comunicaciones Informales: Relacionarse con los demás	U		
41 Preguntar, Escuchar y Dialogar	EU	U	U
42 Negociación, compromiso y resolución de conflictos	EU	U	U
43 Establecimiento de conexiones	IEU	U	U

Tabla 4.1.: Competencias para los niveles 2 y 3 CDIO

HABILIDADES CDIO			
Nivel 1			
E. Dig1	E. Dig2	Sist. Emb.	
Contexto Externo, Social, Económico y Ambiental	IEU		
22 Rol y responsabilidad de los Ingenieros		IEU	
23 Impacto sobre la sociedad y el medio ambiente		IEU	
24 Cuestiones y valores actuales	IEU		
44 Sostenibilidad y necesidad de un desarrollo sostenible	IE	IE	IE
Empresa y contexto empresarial	EU		
25 Interesados en la empresa, metas y objetivos		I	
26 Espíritu Empresarial Técnico		I	
27 Trabajo exitoso en organizaciones		I	
45 Finanzas y Economía de los Proyectos de Ingeniería	IE	IE	IE
Concepción y Administración de Sistemas en Ingeniería.	IEU		
28 Entender las necesidades y establecer las metas	IEU	EU	U
29 Definir la función, concepto y arquitectura	IEU	EU	U
Diseño	IEU		
30 Proceso de Diseño	IEU	EU	U
31 Fases del proceso de Diseño y enfoques	IEU	EU	U
32 Utilización de conocimiento científico en el diseño	IEU	EU	U
33 Diseño específico	IEU	EU	U
34 Diseño multi-disciplinario	I	E	U
Implementación	EU		
35 Proceso de fabricación Hardware	IEU	EU	U
36 Proceso de Implementación de Software	I	EU	U
37 Integración Software - Hardware	I	EU	U
38 Pruebas, verificación, validación y certificación	IE	EU	U

Tabla 4.2.: Competencias para CDIO

4.3. Integración de las Habilidades CDIO al Plan de Estudios

4.3.1. Metodología de Diseño

La metodología de diseño utilizada en la creación de este programa académico se describió en la sección 3.1.4 del capítulo 3. En la figura 4.3 se muestra como será abordada esta metodología en los diferentes cursos que componen la línea de electrónica digital.

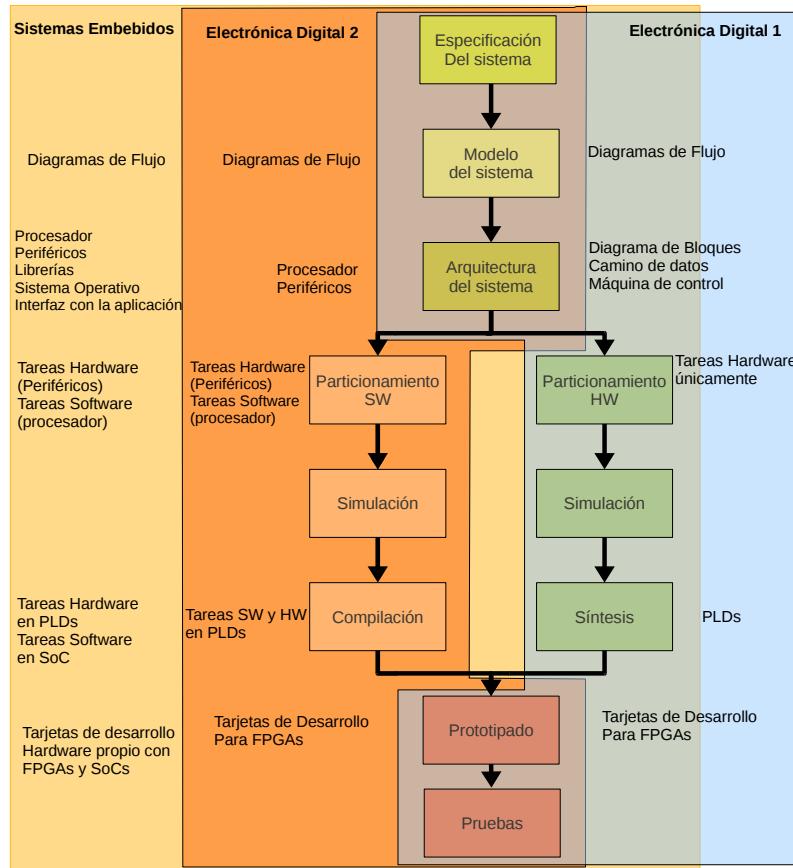


Figura 4.3.: Metodología de diseño para el área de Sistemas Digitales

4.3.2. Objetivos de Aprendizaje - Dominio Cognitivo

El dominio cognitivo involucra conocimiento y desarrollo de habilidades intelectuales. Incluye el reconocimiento de hechos específicos, procedimientos, y conceptos que ayudan en el desarrollo de habilidades y capacidades intelectuales. Bloom [65] identificó las siguientes seis categorías, las que están ordenadas desde el comportamiento más simple al más complejo. Las categorías pueden considerarse como grados de dificultad. Es decir, se deben dominar las primeras antes de poder desarrollar las otras.

1. Conocimiento: definido como la acción de recordar información aprendida anteriormente.

- Objetivos generales
 - *Identifica y reproduce* las diferentes etapas del proceso de diseño.
 - *Identifica* la estructura de un sistema digital.
 - *Selecciona* los componentes adecuados para cada problema (particionamiento hardware/software).
 - *Define* las especificaciones de un sistema digital.
 - *Lee* esquemáticos electrónicos y hojas de especificaciones de sus componentes.
 - *Escribe* documentos que *describen* el proceso de diseño de un sistema digital.
 - Electrónica digital 1
 - *Identifica* la arquitectura de un circuito lógico y secuencial.
 - *Reproduce* los pasos para sintetizar una máquina de estados algorítmica.
 - *Identifica* los componentes básicos de una máquina de estados algorítmica.
 - *Identifica* los pasos del flujo de diseño hardware.
 - Electrónica digital 2
 - *Identifica* las tareas que deben ser ejecutadas en software y en hardware.
 - *Identifica y describe* los componentes de un SoC.
 - *Identifica y describe* el funcionamiento de los componentes de una unidad de procesamiento central.
 - *Identifica* los pasos del flujo de diseño software.
 - *Describe* el comportamiento de tareas hardware y software.
 - Sistemas embebidos
 - *Describe* la integración de software en hardware electrónico.
 - *Identifica* los componentes de un sistema embebido.
2. Comprensión: relacionado con los objetivos, comportamientos o respuestas que representan el entendimiento de un mensaje contenido en una comunicación, sin referirse a otro material. Para llegar a este entendimiento el estudiante puede cambiar la comunicación en su mente para reflejar una forma alterna más significativa para él.
- Objetivos generales
 - *Entiende* las etapas de la metodología de diseño de sistemas digitales.
 - *Entiende* la diferencia entre tareas hardware y software.
 - *Analiza* las diferentes formas de implementar un sistema digital.
 - *Describe* el comportamiento de un sistema digital.
 - Electrónica digital 1
 - *Entiende* la forma de implementar máquinas de estado algorítmicas en dispositivos lógicos programables (PLD) utilizando lenguajes de descripción de hardware.
 - Electrónica digital 2

- *Entiende* el funcionamiento de un SoC.
 - *Entiende* los canales de comunicación entre un periférico y la unidad de procesamiento central.
 - *Entiende* los pasos necesarios para crear aplicaciones utilizando lenguajes de alto nivel.
 - *Entiende* el uso del conjunto de instrucciones en la elaboración de aplicaciones.
- Sistemas embebidos
 - *Entiende* la diferencia entre procesadores softcore y hardcore.
 - *Entiende* la diferencia entre aplicaciones que utilizan y las que no utilizan sistema operativo.
3. Aplicación: habilidad de utilizar información aprendida previamente en nuevas situaciones para resolver problemas con una única o mejor solución.
- Objetivos generales
 - *utiliza y aplica* la metodología de diseño de sistemas embebidos en la solución de problemas.
 - *Implementa* sistemas digitales que dan solución a una necesidad de la sociedad.
 - *Implementa* placas de circuito impreso.
 - *Utiliza* software como ayuda en el proceso de diseño.
 - *Aplica* el concepto de re-utilización de código.
 - *Utiliza* niveles de jerarquía para el diseño de sistemas digitales.
 - *Construye* sistemas digitales.
 - *Desarrolla y aplica* pruebas para comprobar el correcto funcionamiento de un sistema digital.
 - Electrónica digital 1
 - *Implementa* máquinas de estado algorítmicas.
 - *Utiliza* lenguajes de descripción de hardware.
 - *Utiliza* el flujo de diseño hardware.
 - Electrónica digital 2
 - *Implementa* tareas hardware como periféricos de un SoC.
 - *Utiliza* herramientas de compilación y el flujo de diseño software.
 - *Implementa* tareas software utilizando una unidad de procesamiento central.
 - *Desarrolla* aplicaciones que implementan tareas hardware y software.
 - *Utiliza* canales de comunicación entre tareas hardware y software
 - Sistemas embebidos
 - *Desarrolla* aplicaciones utilizando un sistema operativo.
 - *Implementa* periféricos en un PLD.

- *Desarrolla* controladores del sistema operativo para periféricos implementados en PLDs.
 - *Utiliza* librerías disponibles para el desarrollo de aplicaciones.
 - *Desarrolla* interfaces gráficas para aplicaciones embebidas.
4. Análisis: La separación de la información en sus partes componentes, examinando y entendiendo su estructura. Distinguiendo entre hechos e inferencias.

- Objetivos generales
 - *Distingue* las diferentes formas de implementación de sistemas digitales.
 - *Subdivide* un problema en componentes funcionales.
 - *Reconoce* los componentes de un sistema digital.
 - *Distingue* los dominios de descripción y los niveles de abstracción de los sistemas digitales.
 - *Reconoce* los niveles jerárquicos en un proceso de diseño.
 - *Diferencia* la implementación de tareas hardware y software.
- Electrónica digital 1
 - *Distingue* los componentes de la lógica combinatoria y secuencial.
 - *Reconoce* la arquitectura de las máquinas de estado algorítmicas.
 - *Reconoce* sistemas implementados con lenguajes de descripción de hardware.
- Electrónica digital 2
 - *Reconoce* la arquitectura de un SoC.
 - *Distingue* los componentes de una unidad de procesamiento central.
 - *Reconoce* funciones que deben ser implementadas en hardware para cumplir con restricciones de diseño.
 - *Distingue* la diferencia entre un lenguaje de alto nivel y el lenguaje ensamblador.
- Sistemas embebidos
 - *Distingue* los componentes de un sistema embebido.
 - *Reconoce* la arquitectura de sistemas digitales comerciales.
 - *Diferencia* las aplicaciones independientes (standalone) de las que usan sistemas operativos.
 - *Reconoce* las funciones de un sistema operativo.

5. Síntesis: definida como la acción de unir elementos y partes para conformar una estructura o patrón, enfatizando en la creación de un nuevo significado o estructura.

- Objetivos generales
 - *Diseña* y *genera* sistemas digitales que dan solución a problemas reales.
 - *Colabora* en un grupo de trabajo para dar solución a un problema común.
 - *Colabora* con la sociedad compartiendo el resultado de sus estudios.

- *Combina* dispositivos semiconductores para cumplir especificaciones de diseño.
- *Comunica* el resultado de sus investigaciones utilizando medios electrónicos.
- *Desarrolla* pruebas para determinar el correcto funcionamiento de un sistema.
- *Crea* soluciones a problemas reales utilizando dispositivos electrónicos.
- *Planea* actividades que darán como resultado soluciones a problemas reales.
- *Valida* el cumplimiento de restricciones de diseño.
- Electrónica digital 1
 - *Diseña* máquinas de estado algorítmicas
 - *Modifica* diseños de referencia para ajustarlos a necesidades puntuales.
- Electrónica digital 2
 - *Integra* tareas software en el diseño de soluciones.
 - *Diseña* SoC dedicados.
 - *Diseña y desarrolla* tareas hardware como periféricos de un SoC.
- Sistemas embebidos
 - *Diseña* sistemas digitales utilizando SoC comerciales que ejecutan un sistema operativo.
 - *Integra* librerías y aplicaciones existentes en el diseño.
 - *Diseña* periféricos y sus respectivos controladores para el sistema operativo utilizado.

6. Evaluación: emitir juicios personales sobre el valor de las ideas o materiales.

- Objetivos generales
 - *Compara* las ventajas y desventajas de las tareas hardware y software.
 - *Decide* la arquitectura mas adecuada para un sistema digital.
 - *Interpreta* los resultados de las simulaciones.
 - *Justifica y replantea* sus criterios de diseño.
 - *Critica* implementaciones y arquitecturas de sistemas digitales.
 - *Evalúa* las diferentes alternativas de implementación de un sistema digital.
- Electrónica digital 2
 - *Evalúa y compara* el desempeño de funciones implementadas en software y en hardware.
 - *Decide* el particionamiento hardware/software.
- Sistemas embebidos
 - *Evalúa* las ventajas y desventajas de utilizar un sistema operativo.

4.3.3. Dominio Afectivo

- Recibir: disposición a recibir información:
 - *Identifica* malas costumbres en su método de estudio que le impiden el aprendizaje.
 - *Utiliza* material disponible para aumentar su conocimiento.
 - *Localiza* vacíos conceptuales e identifica soluciones.
- Responder: reacciona y participa activamente en su propio aprendizaje.
 - *Lee* material relacionado con el tema del curso con anterioridad.
 - *Responde* dudas formuladas en clase de temas que no han sido tratados previamente.
 - *Asiste* a las clases con preguntas relacionadas con el proceso de concepción, diseño e implementación de sistemas digitales.
 - *Presenta* los ejercicios propuestos en clase.
 - *Ayuda* a sus compañeros en el proceso de aprendizaje.
 - *Practica* para adquirir habilidad en temas relacionados con el curso.
 - *Escribe* material que le ayude a entender los temas desarrollados en clase.
- Valorar: Asigna valores y expresa opiniones personales
 - *Muestra una creencia* en la importancia del auto-aprendizaje en el desarrollo de habilidades personales.
 - *Muestra una creencia* en la importancia de la creación de nuevos productos en el desarrollo del país.
 - *Comparte* el resultado de su esfuerzo con futuras generaciones.
 - *Participa* y hace aportes en grupos de discusión.
 - *Valora* el trabajo en equipo.
 - *Explica* sus criterios de diseño.
- Organizar valores: resuelve conflictos internos y desarrolla un sistema de valores.
 - *Combina* diferentes habilidades para lograr el objetivo.
 - *Compara* alternativas de solución y elige la más adecuada.
 - *Integra* resultados obtenidos en procesos anteriores a solución de nuevos problemas.
 - *Formula* estrategias para resolver problemas de forma óptima.
- Interiorizar valores: actúa de acuerdo a los valores que ha interiorizado, el comportamiento es pervasivo, consistente, predecible y característico de cada aprendiz.
 - *Propone* métodos para crear nuevos sistemas.
 - *Soluciona* problemas de la sociedad aplicando conocimiento adquirido previa y constante mente.

4.3.4. Dominio Psicomotor

- Imitación: repetir un acto que ha sido demostrado o explicado.
 - *Construir y ensamblar* placas de circuito impreso.
 - *Sigue* guías para la elaboración de layouts.
- Manipulación: práctica de una habilidad específica hasta que la acción se realice con seguridad.
 - *Manejar* herramientas para montaje de placas de circuito impreso.
 - *Manejar* herramientas para diseño de placas de circuito impreso.
- Precisión: se obtienen resultados de muy alta calidad en la ejecución de una tarea con poco esfuerzo.
 - *Mostrar* capacidades elevadas en el montaje y rework de placas de circuito impreso.
 - *Mostrar* habilidades en la elaboración de layouts.
- Articulación: las habilidades han sido desarrolladas de tal forma que pueden modificarse patrones para ajustarse a requerimientos especiales.
 - *Combinar* diferentes técnicas en la elaboración, montaje y adecuación de placas de circuito impreso.
 - *Adapta* diferentes técnicas de ruteo a necesidades específicas de la aplicación.
- Naturalización:
 - *Especifica* el método más adecuado para elaborar, montar, y modificar una placa de circuito impreso.

4.3.5. Metodología

Todas las actividades que se realizarán en estos cursos están encaminadas a generar habilidades necesarias para concebir, diseñar, e implementar sistemas digitales complejos, y están articuladas alrededor de una única metodología de diseño. Los tres cursos tienen un carácter teórico-práctico, el componente teórico tratará los diferentes temas de forma general, con el fin de no crear dependencia con las herramientas utilizadas, lo que permitirá realizar actualizaciones fácilmente. En el componente práctico, se tratarán temas específicos de manejo de las herramientas (lenguajes de descripción de hardware, lenguajes de programación y manejo de plataformas de desarrollo) y como se relacionan con la metodología de diseño utilizada.

El estudiante debe estudiar, profundizar y comprobar algunos temas tratados en clase y debe leer previamente la documentación que se encuentra disponible en el sitio web de los cursos. Adicionalmente, debe formar grupos de trabajo para realizar actividades a lo largo del semestre.

Durante el período académico se trabajará para definir las especificaciones, diseñar e implementar un dispositivo que resuelva una determinada necesidad (con la complejidad adecuada para cada curso). En la sesión teórica se tratarán aspectos relacionados con la concepción, diseño, identificación y definición de las funciones de los componentes del sistema, mientras que en el laboratorio se tratarán temas relacionados con la implementación de dichos componentes sobre PLDs o SoC. Se realizarán presentaciones del avance, indicando las razones que se tuvieron en cuenta en cada decisión y como se resolvieron los problemas encontrados, todo este proceso debe documentarse

en la wiki del portal *linuxencaja*, esto último para formar un banco de proyectos que pueda ser utilizado como referencia por quien esté interesado.

SIE: Plataforma abierta para el desarrollo de sistemas embebidos

En el mercado existe una gran variedad de plataformas que pueden ser utilizadas en el estudio de sistemas embebidos, sin embargo, no todas son adecuadas para la implementación del método que propone este trabajo, ya que la plataforma que se utilice debe proporcionar toda la información necesaria para entenderla, programarla, replicarla y modificarla. Con el fin de proporcionar al estudiante una herramienta que pueda ser utilizada en el desarrollo de nuevos productos comerciales se requiere: acceso a los esquemáticos y a los archivos de fabricación del PCB con posibilidad de modificación; acceso a la documentación completa del proceso de fabricación; acceso a la cadena de producción; utilización de herramientas abiertas para su programación; un PLD para la implementación de tareas hardware; un procesador para la implementación de tareas software; un canal de comunicación entre el procesador y el PLD; y una comunidad que desarrolle aplicaciones para dicha plataforma y que proporcione medios para el intercambio de información a través de listas de correo y wikis.

Después de una búsqueda minuciosa no se encontraron plataformas que cumplieran con estas condiciones, en especial con las relacionadas con el proceso de diseño y de producción; esto es normal, ya que la mayoría de las empresas no quieren que se fabriquen sus plataformas y los proyectos individuales no poseen la infraestructura necesaria para la producción masiva. Por este motivo, se decidió crear una plataforma que cumpliera con los requerimientos (plataforma *SIE*), pudiera ser utilizada en los tres cursos del área y se convirtiera en una herramienta más para la difusión de los conocimientos adquiridos en este estudio.

La Figura 4.4 muestra el diagrama de bloques de la plataforma SIE, en ella se observa un procesador que posee periféricos para comunicación serial (UART), memorias micro-SD, un puerto I2C, un LCD a color de 3 pulgadas, 2 entradas y salidas de audio stereo, 2 entradas análogas; una FPGA que proporciona 25 señales de entrada/salida digitales de propósito general (GPIOs) y controla un conversor análogo digital de 8 canales. Existen dos canales de comunicación entre la FPGA y el procesador: uno para controlar el puerto JTAG, lo que permite la configuración de la FPGA desde el procesador (lo que elimina la necesidad de cables de programación); y otro que proporciona el bus de datos, dirección y control para comunicarse con las tareas hardware o periféricos implementadas en la FPGA. El procesador utilizado es un *Ingenic JZ4725 (MIPS)* corriendo a 400MHz, se dispone de una memoria NAND de 2GB para almacenamiento de datos y programas, así como de una memoria SDRAM de 32 MB, lo que permite la ejecución de una gran variedad de aplicaciones Linux.

SIE proporciona un canal de comunicación y alimentación a través del puerto USB-device, y es configurado para ser utilizado como una interfaz de red (*usb0*), permitiendo la transferencia de archivos y ejecución de una consola remota utilizando el protocolo *ssh*; este canal de comunicación también se utiliza para programar la memoria NAND no volátil, por lo que para realizar la programación completa de los componentes de la plataforma solo es necesario un cable USB. SIE posee un sistema de archivos basado en el proyecto *openwrt* y dispone de una gran cantidad de aplicaciones y librerías que pueden ser compiladas en un computador tradicional, siguiendo los tutoriales de la wiki del proyecto.

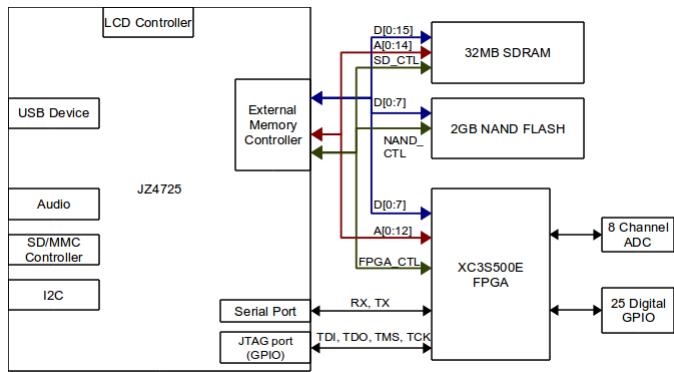


Figura 4.4.: Estructura de la plataforma de desarrollo SIE

4.3.6. Integración de SIE con los Cursos de la Línea de Electrónica Digital

En la actualidad SIE está siendo utilizada en los cursos de la línea de electrónica digital de la Universidad Nacional de Colombia (sede Bogotá). En el anexo ?? se listan las herramientas abiertas que se utilizan en el desarrollo de las actividades propuestas en este plan de estudios.

Electrónica digital 1

En el primer curso del área de diseño digital en la UNAL se realiza el estudio, diseño e implementación de máquinas de estado algorítmicas utilizando la metodología de diseño presentada anteriormente y la herramienta gratuita de Xilinx *Webpack*, los estudiantes implementarán sus diseños utilizando lenguajes de descripción de hardware (VHDL, verilog), como resultado de este proceso se obtendrán 3 archivos: uno para configurar a la FPGA con la funcionalidad deseada; uno que informa sobre la función asignada por el diseñador a todos los pines de la FPGA; y otro que contiene los resultados de la simulación del sistema³. SIE proporciona un canal de comunicación entre el procesador y el puerto JTAG de la FPGA que puede ser utilizado para:

- Configuración: Carga del archivo de configuración con la funcionalidad deseada a la FPGA; este archivo puede ser transferido al sistema de archivos de SIE utilizando el cable *USB* y el protocolo de comunicaciones *ssh*. Cuando este archivo de configuración ha sido transmitido, se utiliza la aplicación *xc3sprog* para configurar la FPGA.
- Prueba a baja frecuencia: Como se mencionó anteriormente, el protocolo JTAG permite la aplicación de vectores de prueba a un dispositivo semiconductor y la recolección de la respuesta a estos estímulos utilizando 4 señales (TDI, TDO, TMS y TCK).

Una aplicación abierta, creada por este trabajo, recibe como entrada los archivos que contienen la información de la asignación de pines y los resultados de la simulación (ver figura 4.5); extrae la información correspondiente a los vectores de prueba y los aplica al circuito implementado en la FPGA, utilizando la instrucción INTEST del protocolo JTAG; finalmente, captura la respuesta a estos estímulos⁴ los que son desplegados en el LCD de la plataforma y pueden

³Este archivo lo generan las herramientas de simulación abiertas *icarus* y *ghdl*

⁴para el control del puerto JTAG se modificó la herramienta *urjtag*

ser exportados a un archivo tipo imagen. Esta herramienta puede verse como una combinación de un analizador lógico y un generador de vectores de prueba de bajo costo.

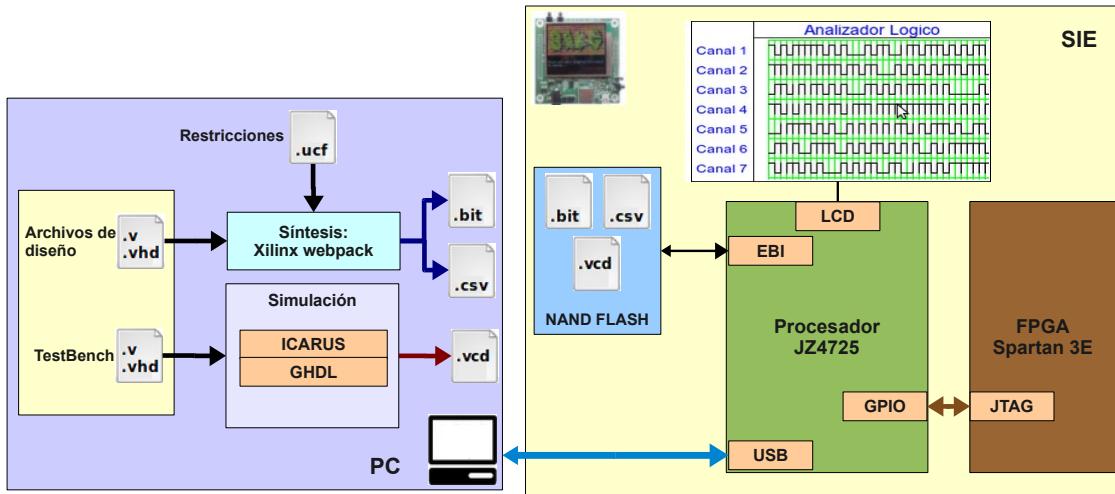


Figura 4.5.: Flujo de diseño hardware

Electrónica digital 2

Como se mencionó anteriormente, en este curso se busca que el estudiante entienda la diferencia entre tareas hardware y software, y los canales de comunicación entre ellas; para esto, se implementarán las tareas software en un procesador *softcore* que proporciona la descripción completa en un lenguaje de descripción de hardware (en la actualidad se utiliza el LM32 de Lattice). Adicionalmente, se estudiará la arquitectura de la unidad de procesamiento, su conjunto de instrucciones, manejo de interrupciones, comunicación con las memorias de datos y de instrucciones, su programación y la forma de comunicación con periféricos.

El acceso al código fuente del procesador facilita el estudio de las operaciones internas del procesador al permitir la simulación y seguimiento de señales específicas. Este tipo de simulaciones permiten entender los conceptos relacionados con el funcionamiento del sistema lo cual no es posible si se utilizan dispositivos comerciales, ya que los simuladores existentes se enfocan en los contenidos del banco de registros y de la memoria y no pueden mostrar (hasta el momento) las señales internas. Al estudiar la estructura interna de un procesador, es posible entender los mecanismos de comunicación entre la CPU y los periféricos; permite entender la diferencia conceptual entre tareas hardware y tareas software (aún cuando ambas se implementan en el mismo dispositivo). La creación de periféricos, su conexión con el procesador y el soporte que se debe dar en la aplicación software para su control; al tiempo que se proporcionan los conocimientos de la estructura de un sistema sobre silicio (SoC) moderno.

La metodología que propone este trabajo permite utilizar la cadena de herramientas GNU existente para el procesador LM32, la que permite trabajar con lenguaje ensamblador, C y C++ y proporciona un flujo de diseño software que puede ser utilizado en otros procesadores comerciales. En la figura 4.6 se muestra el flujo de diseño al utilizar un procesador *softcore*; con él, el estudiante

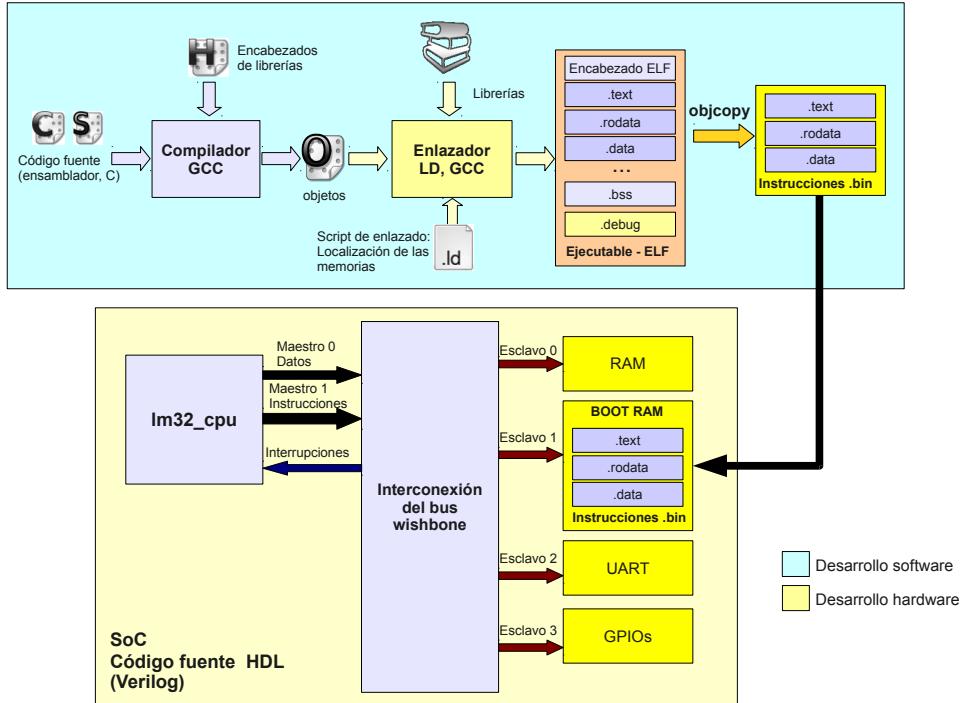


Figura 4.6.: Flujo de diseño hardware/software al utilizar un procesador *softcore*

puede entender que el flujo de diseño software se utiliza para generar el contenido de la memoria de programa del SoC y que la estructura del procesador se describe en un lenguaje de descripción de hardware al que se le aplica el mismo flujo de diseño que aplicó en la asignatura anterior.

En este curso el procesador de la plataforma SIE es utilizado como herramienta de configuración del PLD, los archivos de configuración son transferidos al sistema de archivos de SIE utilizando el protocolo *ssh* y desde allí son transferidos a la FPGA utilizando *xc3sprog* o *urjtag*. Es posible establecer una comunicación serial entre los procesadores *softcore* implementado en la FPGA y *hardcore* de la plataforma; proporcionando un canal de depuración para las aplicaciones que se ejecutan en la FPGA; con lo que se suministran todas las herramientas necesarias para la realización de las actividades previstas en este curso.

Sistemas embebidos

Una vez asimilados los conceptos de arquitectura de SoCs e implementación de tareas hardware y software se utilizará un SoC comercial, para que los estudiantes entiendan las diferencias entre los procesadores *softcore* y *hardcore* y conozcan las herramientas más utilizadas en la implementación de sistemas digitales modernos. Así mismo, se utilizará el sistema operativo Linux para ilustrar la diferencia entre las aplicaciones *standalone* y las que utilizan sistemas operativos; se utiliza el SoC de ingenic JX4725 para ejecutar tareas de visualización, comunicación, control e interfaz con el usuario, librerías gráficas de alto nivel como QT (de Nokia) para realizar la interfaz, se desarrollan módulos del kernel y programas en espacio de usuario para el control de periféricos dedicados (implementados en la FPGA). Con esto se proporciona a los estudiantes herramientas que están siendo utilizadas en la

actualidad por los grandes fabricantes de dispositivos digitales como Nokia, Dell, Hewlett Packard.

4.4. Desarrollo de Métodos de Evaluación

La naturaleza de esta metodología hace poco eficiente el método tradicional de evaluación, ya que el proceso de diseño no se puede limitar a las 2 o 4 horas que puede durar una prueba escrita, tampoco se puede aislar al estudiante de las fuentes de información, ni de la consulta con otros estudiantes. La forma de evaluación debe simular un entorno laboral, donde se trabaja en equipo y cada uno de sus miembros es responsable de una tarea; en la evaluación se tendrá en cuenta el resultado final, pero cada miembro del equipo debe sustentar de forma individual su aporte y sus conocimientos; esto con el fin de asegurar que todos los miembros del equipo realicen sus labores asignadas y estén al corriente de las actividades que realizan sus compañeros.

La aplicación de la forma de evaluación tradicional a asignaturas en las que se busca crear en el estudiante habilidades que le permitan realizar el flujo de diseño completo de un sistema digital, contemplaba una marcada división entre el componente práctico y el componente teórico. Antes de aplicar este programa académico, las prácticas de laboratorio y el tema que se trataba en clase no estaban relacionadas y las metodologías utilizadas eran diferentes, lo que originaba confusión en los estudiantes; existía un proyecto final que debía ser elaborado junto con las prácticas de laboratorio, lo que creaba una sobrecarga de trabajo que se traducía en proyectos de muy baja calidad. Por otro lado, no se puede medir la capacidad de diseño de un estudiante en un examen convencional como los que se utilizan en otras asignaturas, primero porque el tiempo requerido para entender y asimilar el problema varía dependiendo de cada persona, y limitar este tiempo favorecería a los más rápidos pero no necesariamente a las mejores soluciones.

Durante todo el semestre se realizará un proyecto que busca dar solución a un determinado problema; se realizarán tres avances para determinar su estado, las fechas y contenido de estos avances están sincronizadas con un cronograma que permite a los estudiantes aplicar los conocimientos adquiridos. De esta forma, los estudiantes estarán realizando actividades durante todo el semestre, lo que los obliga a estar revisando y aplicando constantemente la información obtenida en el componente teórico; durante las horas de práctica los estudiantes trabajarán en el mismo proyecto y darán solución a problemas de implementación con ayuda del profesor encargado del componente práctico. Esta forma de evaluación elimina la costumbre de estudiar una semana o un día antes de la prueba, buscar evaluaciones de semestres anteriores y memorizar la forma de solucionarlos. Asimismo, se proporciona al estudiante el tiempo suficiente para realizar el proceso de diseño completo lo que hace que esta experiencia se aproxime mucho a una situación que va a encontrar en el ejercicio de su profesión; adicionalmente, se pretende reducir la presión y el estrés que generan las pruebas escritas en los estudiantes.

Contenido de las entregas

En cada entrega el equipo de trabajo debe realizar una serie de actividades que le ayuden a crear o mejorar habilidades en: la generación de documentos técnicos; presentación oral y escrita de proyectos en ingeniería; aplicación de metodologías de diseño; uso de la tecnología para resolver problemas locales; trabajo en equipo y mejoramiento de técnicas de auto-aprendizaje. Por esta razón, se evaluará el contenido del informe, su presentación oral y el avance de la solución. Para la elaboración de los informes se suministran las herramientas web (*wiki*) que permiten la edición de documentos por múltiples usuarios, llevar un historial de cambios que les permita conocer los últimos

aportes y publicar de forma fácil imágenes y videos; lo que facilita la realización del documento, permitiendo que sea editado desde diferentes lugares; al tiempo que lo pone a disposición de quien esté interesado.

Sustentación individual

Con las entregas parciales se obtiene una calificación que refleja el trabajo realizado por el grupo de trabajo; sin embargo, es necesario determinar la contribución de cada miembro del equipo y evaluar la asimilación de conocimiento de forma individual; adicionalmente, se pretende evitar que miembros del equipo no realicen aportes. Para determinar este nivel de asimilación, se aplica una prueba oral a cada uno de los integrantes, con el fin de determinar si existen o no vacíos conceptuales y si se conoce el trabajo realizado en la elaboración del proyecto; de esta prueba se obtendrá un coeficiente entre **0** y **1**, el cual será aplicado a la calificación obtenida por el grupo y de esta forma obtener la calificación individual.

4.5. Actividades

A continuación se enumerarán las actividades que se desarrollan en las tres asignaturas, indicando las habilidades (ver tablas 4.1 y 4.2) que se quieren reforzar o desarrollar.

Lectura de material del curso 10, 11

Con la lectura previa de los temas, el estudiante adquiere la capacidad de absorber conocimiento (11), identificar sus preferencias, deficiencias y buscar ayuda para suplirlas (10), lo cual ayuda al mejoramiento de las habilidades para el auto-aprendizaje.

Lectura de material técnico en inglés 10, 11, 6, 30, 33, 21

La mayor parte de la documentación de los componentes electrónicos y literatura relacionada con los avances de la industria electrónica está escrita en inglés técnico; por esta razón, es necesario que el estudiante aprenda a entender este tipo de escritura y se familiarice con su estructura. Esto le permite identificar el funcionamiento de un componente del sistema (6,30), determinar que componente se adapta mejor a sus necesidades (33) y mejorar sus habilidades para comunicarse en inglés (21).

Utilización de metodologías de diseño 1, 2, 3, 6, 7, 9, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38

La metodología de diseño (30,31) de sistemas embebidos requiere identificar un problema (1, 28), plantear una solución (3,29,32) lógica (9) de alto nivel (9), modelarla (2) a nivel de sistema(6), verificar el cumplimiento de los requerimientos (33,38); proporciona métodos para determinar su arquitectura óptima y definir la función e interacción (37,7) de sus componentes software (36) y hardware (35).

Implementación de sistemas digitales sencillos 3, 14, 29, 30, 35, 36, 17, 18, 19

La realización de prácticas de laboratorio en las que grupos de trabajo (14) implementan diseños de baja o media complejidad le permite al estudiante: Formular recomendaciones (3) para que no se repitan errores en experiencias futuras y utilizar sistemas de desarrollo (30) para la implementación de tareas hardware y software a bajo nivel (36). Con el fin de mejorar la capacidad de comunicación escrita (18, 19) se deben presentar informes que refuerzen las habilidades generadas en la utilización de la metodología de diseño, los cuales deben tener la siguiente estructura (17):

- Un diagrama de caja negra que indique las entradas y salidas del sistema.
- Una descripción de alto nivel del algoritmo que implementa la solución (29).
- Un diagrama de bloques que indique el particionamiento y la interconexión entre sus componentes (30).
- Descripciones de alto nivel de cada uno de los componentes (31).
- La implementación y simulación de cada componente y del sistema completo (35), donde se muestre que el sistema cumple con las especificaciones funcionales (38).

Proyecto del curso 1,2,3, 14, 15, 30, 31, 32, 33, 34, 35, 22, 23, 24, 25, 27

Durante el semestre se trabajará para definir las especificaciones (1,2,3), diseñar (30,31,32,33,34) e implementar un dispositivo que resuelva una necesidad de la sociedad (22). En la sesión teórica se tratarán aspectos relacionados con la concepción, diseño, identificación y definición de las funciones de los componentes del sistema, mientras que en el componente práctico, los relacionados con la implementación de dichos componentes sobre PLDs o SoCs.

Los estudiantes hacen una descripción funcional de alto nivel del sistema, se organizan en grupos de trabajo (14,15), definen la función de cada uno de sus integrantes (27,14,31), establecen estrategias de comunicación (16,31), realizan y cumplen un cronograma de actividades (25,31) que permite resolver la necesidad en el tiempo especificado (22). Una de las estrategias de comunicación es la realización de presentaciones orales (20), en las que cada equipo de trabajo expondrá el estado de su proyecto, indicando las razones que se tuvieron en cuenta en cada decisión y como se resolvieron los problemas encontrados (24). Adicionalmente, todo este proceso debe documentarse en el sitio web del curso (wiki del portal *linuxencaja*) con el objetivo de crear una base de proyectos que permitan a futuros estudiantes utilizar la experiencia obtenida (23) y cuando sea el caso dar continuidad al proyecto.

El estudiante debe diseñar y construir placas de circuito impreso con los circuitos necesarios para su aplicación (35) siguiendo las normas de diseño establecidas por el fabricante (resolución, número de capas, costo) y las restricciones del circuito (capacidad de corriente, niveles de ruido, compatibilidad electromagnética, etc). Vale la pena aclarar que durante el primer curso los estudiantes no poseen la experiencia necesaria para realizar (sin asistencia) labores como la división de tareas, generación de un cronograma de actividades y fijar la estrategia de comunicación, razón por la cual el docente debe acompañarlos en este proceso.

Desarrollo del plan de innovación del Producto, 22, 23, 44, 25, 26, 45

Todo proyecto debe ser presentado como una idea innovadora; por este motivo, se debe realizar un plan de innovación donde se especifique:

- Oportunidad:
 - Concepto de la innovación
 - ¿Qué problema pretende solucionar?
 - ¿De dónde viene la idea?
 - ¿Existen productos similares?
 - Etapa de desarrollo.
- ¿Quiénes son los clientes claves de esta propuesta?
- ¿Cuál es el valor del producto para sus clientes?
- Mercado y crecimiento
 - ¿En qué mercado está inmersa la innovación?
 - ¿Qué alianzas son necesarias para terminar el desarrollo de la innovación?
- Productos/servicios
 - ¿Qué etapas de desarrollo adicionales son necesarias?
 - ¿Qué productos derivados se pueden obtener?
 - Canales de distribución.
- Inversión
 - Inversión estimada para finalizar el producto.
 - Inversión necesaria para comercializar el producto,
- Próximos pasos para concretar la innovación.

Participación en listas de discusión 21

Con el objeto de aumentar las capacidades en la comunicación en idioma extranjero, se alentará a los estudiantes a que hagan parte de listas de discusión en diferentes temas técnicos, algunos problemas que encontrarán en la realización de las diferentes prácticas deben ser consultados en estas listas para encontrar una forma de solución

4.6. Discusión

El método de evaluación propuesto ha demostrado durante su aplicación su eficacia en la generación de hábitos de continuo estudio y auto-aprendizaje; sin embargo, uno de los grandes inconvenientes para su aplicación son los malos métodos de estudio que poseen los estudiantes; por este motivo, es necesario crear actividades que motiven a los estudiantes a realizar las lecturas y actividades programadas en el curso. Esta situación se agrava debido a la aplicación de una desafortunada política gubernamental en la educación media que obligaba a los centros educativos de todo el país a promover de forma automática a todos los estudiantes, fijando un máximo del 5 % de estudiantes no promovidos, sin importar si los estudiantes poseen o no los conocimientos necesarios para la promoción. Cuando estos estudiantes, así formados, ingresan a las universidades se genera un

choque fuerte ya que no están acostumbrados a afrontar retos, no son conscientes de sus deficiencias y en muchos casos su nivel de conocimientos no son los esperados.

Encuestas realizadas a los estudiantes durante los últimos dos años muestran que ellos perciben un grado de exigencia mucho mayor comparando con otras asignaturas; pero al mismo tiempo, que la experiencia en estos cursos es muy útil para su vida profesional y que es la única asignatura que los enfrenta a problemas reales de diseño e implementación de sistemas y trabajo en equipo; entienden que es necesario dedicar tiempo por fuera de aula si se desea asimilar la información, son conscientes de que la responsabilidad de adquirir este conocimiento es de ellos; manifiestan la importancia del uso de esta tecnología en la solución de problemas locales y entienden el estado de la industria digital en el país y su papel para dar solución a sus problemas.

Con la aplicación de este plan de estudios a las asignaturas del área de electrónica digital: se elevó el nivel de los trabajos realizados en estos cursos y en trabajos de grado; la aplicación de la metodología de diseño propuesta generó hábitos que permitieron re-utilizar trabajos previos, documentar de forma adecuada el proceso de diseño, y generar dispositivos que se ajustan a las especificaciones; se eliminó el uso de placas de pruebas (protoboards) y placas universales (perfboard), siendo reemplazados por circuitos impresos diseñados por los estudiantes; se eliminó el uso de productos con tecnologías obsoletas (familias 74xx y 40xx) para dar paso a dispositivos semiconductores modernos; se creó un banco de proyectos que documenta de forma detallada el proceso de concepción, diseño e implementación de un sistema digital de función específica; se concienció a los estudiantes y profesores sobre la importancia de la creación de nuevos productos para el desarrollo tecnológico del país.

Tanto los temas como la metodología del presente plan de estudio representan una forma novedosa de enseñanza enfocada a generar en los estudiantes habilidades necesarias para crear productos innovadores, proporcionando conocimientos actualizados y metodologías de diseño modernas basadas en herramientas abiertas que permiten conocer y ejecutar el flujo de diseño completo, trabajar en equipo y compartir los resultados con quien esté interesado. Se espera que estas habilidades sean utilizadas por la industria para crear una oferta local de bienes y servicios relacionados con el diseño digital y que se genere el interés necesario en algunos estudiantes para crear empresas de diseño digital.

De la experiencia obtenida al dictar cursos de capacitación en diferentes centros de formación a lo largo del país (UDFJC, UIS, ULA, Escuela Naval Almirante Padilla (ENAP), USTA, Escuela Colombiana de Ingeniería (ECI)) y en dos empresas (SAR S.A., Microensamble S.A.), se puede decir que muchos profesionales no cuentan con las habilidades necesarias para absorber y aplicar los conocimientos necesarios para el diseño e implementación de sistemas embebidos, debido en parte al uso de metodologías de diseño obsoletas y al abandono de la implementación física por parte de los centros de formación. De lo anterior se puede concluir que al aplicar el programa académico propuesto se pueden generar en los estudiantes las habilidades necesarias para que puedan realizar procesos de auto-aprendizaje, diseño e implementación de sistemas digitales, los que pueden ser utilizados en las empresas para desarrollar sus propios productos.

Aunque en este capítulo se hizo una descripción detallada del proceso de aplicación de la iniciativa CDIO a las asignaturas de pregrado, es importante que esta sea llevada a las asignaturas de posgrado. En el DICEE de la UNAL existen dos programas de postgrado: Maestría en automatización industrial y la maestría en ingeniería eléctrica; ninguno de estos programas tiene como línea de profundización la electrónica digital, razón por la cual no se realizó una transferencia de los conocimientos adquiridos a estos programas de posgrado. Sin embargo, se creó una línea de investigación en el nuevo programa de maestría en ingeniería electrónica, este programa está en las

últimas etapas de su creación y se espera que reciba sus primeros estudiantes en el año 2012. La línea de investigación creada para esta maestría recibe el nombre de: *Desarrollo de sistemas embebidos aplicados* y las asignaturas derivadas de este estudio que serán ofrecidas son: *Temas avanzados en técnicas digitales y Diseño e implementación de sistemas embebidos*.

5. El Conocimiento Como Bien Público y su Papel en la Difusión Tecnológica

5.1. Introducción

Es indudable que el desarrollo tecnológico de un país se encuentra ligado al mejoramiento de la calidad de vida de sus habitantes. Para que en un país en vías de desarrollo se realice una transferencia tecnológica (y de conocimientos asociados a la tecnología que se transfiere) exitosa que permita desarrollar productos similares, y ajustados al contexto socio-económico local, es necesario que el país cuente con la capacidad de absorber las habilidades, técnicas, información y organización asociadas a dicha tecnología. Esta absorción de conocimientos debe ser realizada por un gran número de personas para que la transferencia tenga un impacto significativo en la sociedad. La difusión de estos conocimientos es, el punto donde fallan los canales tradicionales de transferencia de tecnología y de conocimientos, ya que restringen el acceso a un grupo específico (representantes en el país de productos extranjeros; quien pueda pagar por cursos de capacitación; quienes puedan ingresar a centros de formación consolidados). Para obtener la máxima difusión del conocimiento, es necesario considerarlo como un bien público (no rivalidad, sin posibilidad de exclusión), y como tal, el acceso a él, debe ser un derecho y por lo tanto, se deben garantizar los mecanismos de difusión para que llegue a los sectores de la sociedad interesados en él. De igual forma, es un deber de los sectores que utilizan este bien contribuir a su difusión, actualización, mejoramiento, y crecimiento.

En este estudio, el conocimiento se refiere a las ideas intangibles, información y datos de todo tipo en el que el conocimiento es expresado u obtenido; como todo tipo de entendimiento adquirido a través de la experiencia o estudio ya sea propio, científico, académico o no académico. El conocimiento puede ser visto como una mercancía y como una fuerza constitutiva de la sociedad, lo que evidencia la naturaleza compleja de este recurso; la adquisición y descubrimiento de conocimiento es un proceso social y personal.

La diferencia entre esta propuesta y las existentes promovidas por los organismos gubernamentales está en el énfasis en la transferencia y difusión gratuitas del conocimiento y del *saber hacer* en la concepción, diseño, implementación y diseño de sistemas embebidos. La Universidad Nacional de Colombia, como el centro educativo más importante de Colombia, tiene la obligación de cubrir este tipo de necesidades y hacer partícipes de los beneficios de su actividad académica e investigativa a los sectores sociales que conforman la nación colombiana; por este motivo, todos los resultados del presente trabajo estarán a disposición de quien esté interesado sin restricción alguna.

En este capítulo se hablará de las licencias *Creative Commons*, una forma de licenciamiento que permite asignar diferentes “permisos” de utilización a un determinado trabajo y pueden ser utilizadas en productos software y hardware. Se explorará el movimiento de software libre y código abierto (FOSS) para identificar como contribuye a la transferencia tecnológica y como millones de usuarios a lo largo y ancho del mundo trabajan de forma conjunta para difundir, actualizar, mejorar, y aumentar las aplicaciones disponibles. Adicionalmente, se presenta una propuesta para la creación de un movimiento que funciona con políticas similares pero centrado en el desarrollo hardware.

5.2. El Conocimiento como Bien Público

El análisis del conocimiento como bien público tiene sus bases en el estudio de recursos naturales compartidos (agua, bosques, fauna y flora). El conocimiento posee características que lo diferencian de los recursos naturales, mientras muchos bienes públicos ven comprometido su sostenimiento a medida que aumenta el número de usuarios, el objetivo del conocimiento es lograr su máxima difusión [66]. Al igual que otros bienes, es utilizado de forma conjunta y es manejado por grupos de diversos intereses y tamaños. Para que este recurso sea útil se requiere una infra-estructura que permita su difusión, actualización, mejoramiento, realimentación y crecimiento.

5.2.1. Auto-Gobierno

Ostrom [67] argumenta que la forma más eficiente de administrar un bien común es el auto-gobierno ejercido por los beneficiarios de dicho recurso. Si estos beneficiarios son conscientes de la importancia de un uso eficiente y racionado que garantice la existencia de este bien y toda acción o decisión que se haga es formulada pensando en el beneficio común se crearán normas que garantizan el beneficio de todos los usuarios y la sostenibilidad del recurso. El auto-gobierno requiere acciones colectivas que “emergen cuando se requiere el esfuerzo de dos o más individuos para cumplir con una meta” [68]. Otro aspecto importante de las acciones colectivas es que son voluntarias por parte de los individuos [69]. El auto-gobierno requiere de la acción combinada de conocimiento y voluntad unido a normas institucionales consistentes. Ostrom [70] y Baland y Platteau [71], consideran que los atributos necesarios para que un recurso tenga una probabilidad alta de crear asociaciones de auto-gobierno son:

- Posibilidad de mejoramiento: El recurso se encuentra en un punto tal que sea inútil crear una organización alrededor de él.
- Indicadores: Disponibilidad de indicadores confiables y válidos sobre la condición del recurso.
- Predictibilidad: El flujo de unidades de recurso es relativamente predecible.
- El recurso es lo suficientemente pequeño, teniendo en cuenta las tecnologías de comunicación utilizadas, para que los usuarios pueden tener un conocimiento preciso de los límites externos y de los micro-ambientes internos.

Los atributos de los usuarios del recurso son:

- Prominencia: Los usuarios dependen del recurso para una parte importante de su sostenimiento.
- Entendimiento común: Conocimiento sobre el funcionamiento del sistema y como acciones individuales afectan a los demás y al sistema.
- Confianza y reciprocidad: Los usuarios del recurso confían en que los demás mantendrán las promesas y se relacionarán entre ellos con reciprocidad.
- Autonomía: Los usuarios pueden determinar reglas sin la intervención de autoridades externas.
- Experiencia previa en organización y liderazgo local: Los usuarios poseen habilidades mínimas de organización y liderazgo a través de la participación en otras asociaciones locales u organizaciones en grupos cercanos.

De los anterior se puede decir que para que pueda emerger un auto-gobierno asociado a un recurso debe existir un excelente canal de comunicación entre sus usuarios y que todos ellos deben conocer y aceptar una serie de normas (creadas por ellos mismos) que fueron formuladas pensando en el beneficio común. Un grupo de estas normas debe establecer mecanismos de resolución de conflictos; esto es vital ya que toda la organización se basa en la confianza mutua y no puede permitirse que las relaciones entre los usuarios se deteriore. Adicionalmente, se requieren miembros que tengan claro que el éxito de sus acciones y el sostenimiento del recurso depende de sus acciones, las cuales deben realizarse teniendo en cuenta el beneficio común. El conocimiento no se ve afectado por la sustracción (uso) del mismo, todo lo contrario, el objetivo es obtener el máximo número de usuarios; sin embargo, este conocimiento puede llegar a ser obsoleto rápidamente, y por lo tanto inservible; para que esto no suceda, se requiere de su actualización para que refleje el estado actual en el área de interés.

Ecuación costo-beneficio de Ostrom

La pregunta que trata de responder Ostrom en su estudio es ¿cómo emerge el auto-gobierno? [67] Para contestar esto formula una ecuación que describe el análisis costo-beneficio que un individuo realiza para participar en un gobierno comunal:

$$D < C_1 + C_2 + C_3$$

donde D es el incentivo al cambio y compara los beneficios de utilizar las reglas tradicionales (BO) frente a los beneficios de utilizar un nuevo grupo de reglas basadas en un gobierno comunal (BN)

$$D = BN - BO$$

C_1 el costo anticipado asociado a la transición, C_2 el costo de corto plazo para adoptar y apropiar las nuevas estrategias y C_3 son los costos de largo plazo: mantenimiento, monitoreo y auto-gobierno. Para que se realice el cambio, el incentivo al cambio debe ser mayor que los costos asociados a el.

$$\begin{aligned} BN - BO &> C \\ BN &> BO + C \end{aligned}$$

En la actualidad se observa cómo grandes multinacionales, como Nokia, Motorola o Google, están participando y promoviendo el desarrollo de productos y aplicaciones [72] [73] [74] con la participación de la comunidad, lo que representa un giro de 180 grados en la políticas lideradas por multinacionales como Microsoft y Aple en las que todas sus creaciones están protegidas por licencias, acuerdos comerciales y contratos de exclusividad que restringen la participación en su desarrollo y aseguran el monopolio de los resultados de sus investigaciones. El movimiento de Software Libre [75] hizo posible este cambio, proporcionando herramientas de desarrollo y facilidades para que cualquier persona alrededor del mundo pudiera realizar sus propias aplicaciones, gracias a esto se han creado aplicaciones como el servidor web *Apache*, el explorador *Mozilla*, el sistema operativo *Linux*, aplicaciones como *gimp*, *openoffice*, librerías como *ncurses*, *Qt* y entornos de trabajo como *xfce*, *Gnome* y *KDE*. El trabajo realizado por miles de programadores para llegar a este estado ha sido enorme, y por esta razón los grandes multinacionales están poniendo sus ojos en estas iniciativas ya que les representa un gran ahorro de dinero, es decir, el costo de participar en estas iniciativas es mucho menor que el beneficio obtenido por participar en ellas.

Principios de diseño

El primer paso para gobernar un bien público es la identificación de los principios de diseño del recurso. Ostrom [66] después de dirigir un gran número de estudios empíricos sobre gobernanza de recursos de bienes comunes, encontró una serie de principios de diseño que hacen que el recurso sea sostenible a lo largo del tiempo:

- Fronteras claramente definidas
- Las reglas deben reflejar necesidades y condiciones locales.
- Los individuos que están sujetos a estas reglas pueden modificarlas y participar en su elaboración.
- Las autoridades externas deben respetar el derecho de los miembros de la comunidad a crear sus propias reglas.
- Debe establecerse un sistema de seguimiento propio.
- Disponibilidad de un sistema gradual de sanciones.
- Acceso por parte de miembros de la comunidad a mecanismos de resolución de conflictos de bajo costo.
- Las actividades de apropiación, suministro, monitoreo y sanción deben ser organizadas en estructuras anidadas con múltiples capas de actividades

Es importante mencionar que no existe una combinación de principios que garantice el éxito, es necesario probar con este grupo para identificar los principios generales que deben incluir los sistemas robustos. Estos ocho puntos se encuentran fuertemente relacionados con los atributos que requiere un recurso y un usuario de este recurso para que pueda emerger un auto-gobierno y buscan que todos los participantes tengan pleno conocimiento de las reglas que monitorean, regulan, fijan y controlan las actividades de los miembros.

Movimiento software libre

Es interesante ver como este pensamiento se aplica a proyectos de código abierto. El software no es un recurso típico, porque no existe un costo asociado a su uso; sin embargo, es un bien público, ya que es un recurso comunal que prospera o decrece gracias a la contribución de sus miembros. Los participantes de este tipo de recurso son un grupo de colaboradores, en lugar del grupo de posibles propietarios. Un aporte de un usuario a mejorar el proyecto se traduce en un beneficio colectivo. Los contribuyentes pueden ser empresas que pagan a una persona para que adicione una nueva característica, sin embargo, ellos pueden decidir no compartir estos cambios para tener una ventaja competitiva.

En los proyectos de código abierto puede utilizarse una variación modificada de la ecuación de Ostrom [76]:

$$\text{BC} > \text{BN} + \text{C}$$

donde BC es el beneficio a contribuir, y debe superar el costo de contribuir C más el beneficio de no contribuir BN . La comunidad del software libre ha creado un grupo de normas e instituciones alrededor de esta ecuación dando como resultado uno de los estructuras auto-gobernadas más exitosas. Los trabajos de la comunidad de software libre hacen que C sea cada vez más pequeño, proporcionando:

- Herramientas de programación, depuración y librerías que facilitan el desarrollo de nuevas aplicaciones o mejoras a las ya existentes.
- Listas de discusión en donde los creadores de estas herramientas responden preguntas, y buscan en conjunto soluciones a problemas específicos
- Sistemas de control de versiones y protocolos de comunicación.
- Bases de datos de solución de problemas asociadas a las listas de discusión.
- Tutoriales, documentación, libros disponibles en línea.

Para colaboradores no comerciales el costo de no contribuir es pequeño, pero, para los comerciales BN puede resultar muy grande, ya que pagar una nómina de programadores para que diseñen un sistema propio desde cero es mucho más costoso que pagar a pocos programadores para utilizar y modificar proyectos ya existentes.

5.3. Movimiento Software Libre y Código Abierto (FOSS)

Como se mencionó anteriormente, el movimiento FOSS es la estructura auto-gobernada más exitosa y será tomada como referencia en el desarrollo de la iniciativa *hardware copyleft*. Richard Stallman [77] a mediados de los 80 inició un proyecto para desarrollar un sistema operativo abierto/libre basado en *unix* llamado GNU; su principal innovación radica en un nuevo esquema de licencias unido a herramientas de colaboración basadas en Internet, lo que se convirtió en una nueva forma de bien público, donde los miembros de forma colectiva generan un recurso público: el *software*. El desafío de Free and Open-Source Software (FOSS) es realizar *acciones colectivas para crear y mantener este bien público*. A diferencia de la creencia popular, en este movimiento existen derechos de autor (*copyright*) y propiedad; sin embargo, algunos individuos involucrados en el proyecto, poseen derechos legales sobre el código (recurso), tienen control sobre las nuevas versiones del software y pueden excluir a otros que aportan código a las nuevas distribuciones.

Los miembros del movimiento FOSS usualmente son programadores (y usuarios finales de software) que contribuyen (ya sea voluntariamente o que se les pague para hacerlo) para producir un software bajo la filosofía FOSS. La “situación de acción” que estos programadores encaran es, si en algún momento, vale la pena contribuir al desarrollo de este software. La interacción de programadores trabajando de forma conjunta en Internet puede ser visto como un resultado que puede cambiar en el tiempo. Schweik y Semenov [78] identificaron tres estados de este tipo de bien una fase inicial; seguida por un estado de apertura y un estado más maduro de gran crecimiento (en términos de usuarios y participación); estabilización, donde el número de participantes (normalmente pequeño) no varía, o el proyecto se estanca y muere (sin participantes). Para que un proyecto sea exitoso, no es necesaria la participación de un gran número de programadores. La clave del éxito está en la disponibilidad de un programador para contribuir al esfuerzo colectivo de por lo menos un pequeño

grupo de actores que producen y mantienen el software. En la Figura 5.1 se ilustra el marco definido por Ostrom y Hess [4] para analizar el conocimiento como bien común, aplicado al movimiento FOSS.

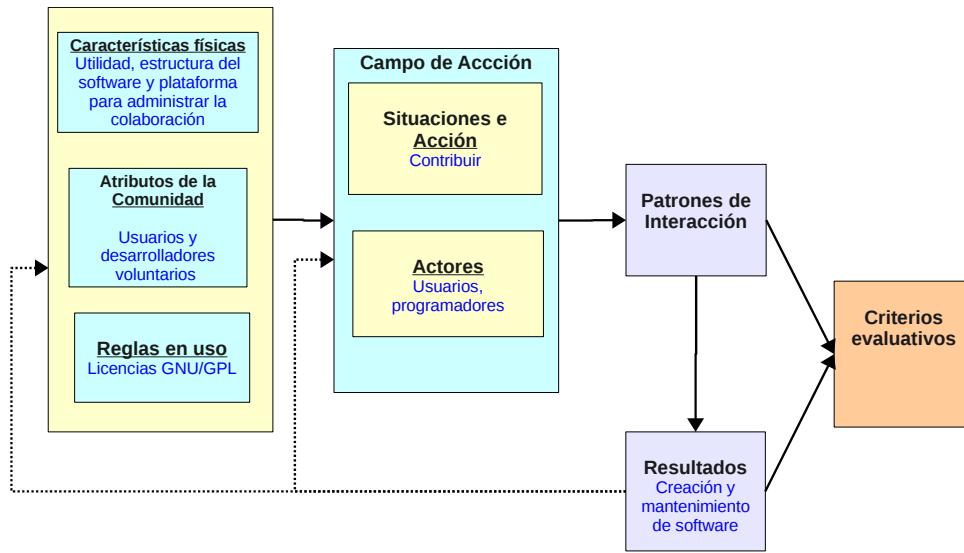


Figura 5.1.: Marco para analizar el conocimiento como bien común aplicado al movimiento FOSS.

Fuente: [4] pág 46

Reglas

Stallman [38] sostiene que las propiedades digitales del software hacen posible considerarlo como un bien público, proporcionando a sus usuarios *la libertad* de utilizarlo, distribuirlo y modificarlo. Esta filosofía se basa en cuatro libertades que encarnan el principio *copyleft*:

- Libertad de ejecutar el software para cualquier propósito.
- Libertad de estudiar como funciona el programa, y cambiarlo para hacer lo que se desee. El acceso al código fuente es una condición para esto.
- Libertad para re-distribuir copias.
- Libertad para distribuir copias de versiones modificadas. Lo que permite que la comunidad se beneficie de estos cambios. El acceso al código fuente es una condición para esto.

Esta filosofía permite que los conocimientos y habilidades que el programador posee puedan ser transferidas a programadores, empresas, instituciones académicas y sociedades; ya sea en forma de producto, o como herramienta de enseñanza. Esta actividad representa un proceso de transferencia tecnológica, en donde, el que suministra la tecnología proporciona todos los medios para que el receptor pueda absorberla y transformarla para satisfacer necesidades en su entorno social; sin embargo, para que la transferencia sea exitosa deben existir personas con la disposición y capacidad de asimilar los conocimientos asociados. Adicionalmente, Stallman ideó una forma de trabajar con las leyes de derechos de autor que proporciona una alternativa al licenciamiento tradicional. El software

con licencia *copyleft* estipula que cualquier modificación que se le realice adquiere los principios de licenciamiento del software original; lo que se conoce con el nombre de licencia tipo viral y garantiza que las mejoras que se realicen al código inicial pueden ser utilizadas por la comunidad. La licencia General Public License (GPL)¹ (GNU General Public License) fue creada para implementar estos principios.

Atributos de la comunidad

La comunidad FOSS está compuesta por usuarios y programadores voluntarios motivados por intereses tecnológicos, socio-políticos, económicos y académicos. Desde el punto de vista tecnológico, resulta más efectivo en tiempo y en dinero, participar en un proyecto conjunto para la realización de una aplicación que satisface una determinada necesidad. La principal motivación socio-política es la creencia por parte del programador en la filosofía del movimiento, en la lucha contra el monopolio del software; adicionalmente, para que el proyecto permanezca durante mucho tiempo, es necesario que sea atractivo para vincular el mayor número de participantes y de esta forma sobrevivir en el futuro. Las motivaciones económicas y académicas pueden ser consideradas en forma conjunta ya que al participar en estos proyectos, los miembros adquieren o mejoran sus habilidades de programación ya sea revisando, leyendo y entendiendo el código existente o sometiéndolo a la revisión de los demás miembros sus contribuciones. Por otro lado, esta participación puede verse como una vitrina en donde el programador muestra sus capacidades y puede ser contactado por empresas para establecer relaciones comerciales. En los dos casos la participación es una forma de inversión y una forma de ganar experiencia y reputación [79].

En proyectos con un gran número de personas asociadas, solo un pequeño porcentaje de ellos realizan la mayor parte del trabajo. En la actualidad muchas empresas están participando de forma activa en estos proyectos, aportando programadores pagados para que contribuyan con el desarrollo y mejoramiento del software. Un estudio realizado a 25 firmas participantes en el proyecto FOSS *Linux Operating System* [80] indica que la tercera parte de los programadores encuestados son pagados por sus empleadores para participar, y que su motivación a participar es el interés propio en estandarización, disminución de costos, estrategias para debilitar a la competencia y esfuerzos para hacer sus productos compatibles con los productos derivados del movimiento FOSS.

El estudio en [81] indica que las motivaciones de los que participan en este tipo de proyectos movidos por la necesidad (need-driven) y la de los aficionados (hobbyists) que participan por diversión o como pasatiempo son muy diferentes. Las preguntas, sugerencias y contribuciones de los participantes movidos por necesidades, alientan a los aficionados a realizar búsquedas de retos útiles e interesantes. Los aficionados proporcionan asistencia y soporte a los que participan por necesidad, respondiendo preguntas, creando características requeridas, integrando contribuciones al código fuente, y proporcionando servicios de mantenimiento como, por ejemplo, preservar o mejorar la arquitectura, modularidad y compatibilidad del código. Esta interacción, es crítica para permitir que los proyectos de código abierto funcionen sin un sistema formal de asignación e identificación de tareas.

Atributos físicos

Existen tres categorías [1] en el movimiento FOSS que pueden ser consideradas como atributos físicos:

¹<http://www.gnu.org/licenses/licenses.html>

1. La utilidad del software: Como se mencionó anteriormente, para que una persona participe en una actividad relacionada con el bien público, el costo por no hacerlo debe ser mayor que el beneficio por participar, es decir, el software debe representar una utilidad grande para que valga la pena participar en el proyecto.
2. El diseño o la estructura del software. Un código limpio (optimizado, documentado y óptimo) y modular, permite el trabajo en paralelo, lo que reduce el tiempo de depuración y el tiempo requerido para implementar nuevas características.
3. La infraestructura que hace posible coordinar y administrar la colaboración y la producción: El uso de Internet para soportar herramientas de control de versiones (como *svn*, *cvs* y *git*) y la comunicación vía listas de discusión hace posible coordinar de forma eficiente los esfuerzos de cooperación; permitiendo:
 - Almacenar versiones de software.
 - Descargar módulos.
 - Adicionar nuevas contribuciones y proteger el código existente.
 - Generar un historial donde se documentan los cambios realizados a lo largo del tiempo.
 - Analizar funciones para identificar diferencias entre versiones.
 - Informar a los miembros del equipo cambios en los módulos del proyecto.

Esta infraestructura trabaja junto con las reglas establecidas para proporcionar procesos que direccionen nuevos trabajos; un sistema para recibir y revisar contribuciones de nuevos módulos que pueden ser incluidos en futuras versiones. En conclusión, los proyectos FOSS evolucionan en el tiempo como resultado de la configuración de sus reglas, atributos de la comunidad y atributos físicos relacionados con la estructura del software para hacer fácil la colaboración y herramientas efectivas para la coordinación de equipos y manejo de contenido.

5.4. Hardware Copyleft

El concepto del *hardware copyleft* nació de un trabajo conjunto realizado con el proyecto *qi-hardware*² cuyo objetivo era crear un movimiento similar al FOSS pero aplicado al diseño hardware; en la actualidad, no existe un movimiento aceptado mundialmente que defina las propiedades que debe tener un proyecto hardware para ser considerado *abierto*; existen una serie de proyectos dispersos con diferentes características y grados de libertad ³ y cada uno de ellos interpreta a su manera lo que es un proyecto hardware abierto.

La iniciativa *Open Source Hardware* Open Source Hardware (OSH), busca aplicar principios generales de los ideales del hardware de código abierto con el fin de crear un estándar con el cual evaluar las licencias para diseños hardware. Esta iniciativa es apoyada por proyectos de hardware abiertos en su mayoría usuarios de la plataforma *Arduino* y por sitios encargados a hackear (aplicar ingeniería inversa) dispositivos comerciales, pero no existen muchos desarrolladores hardware que respalden esta iniciativa.

²<http://en.qi-hardware.com/wiki/Qi:About>

³http://en.wikipedia.org/wiki/List_of_open_source_hardware_projects

Esta propuesta presenta una visión diferente al proyecto iniciado con Qi-hardware, al incluir a la academia y buscar un grupo donde existan desarrolladores hardware y software que contribuyan a la creación de plataformas físicas con su respectivo software de aplicación.

La figura 5.2 ilustra el marco definido por Ostrom y Hess [4] para analizar el conocimiento como bien común, aplicado al movimiento *hardware copyleft*.

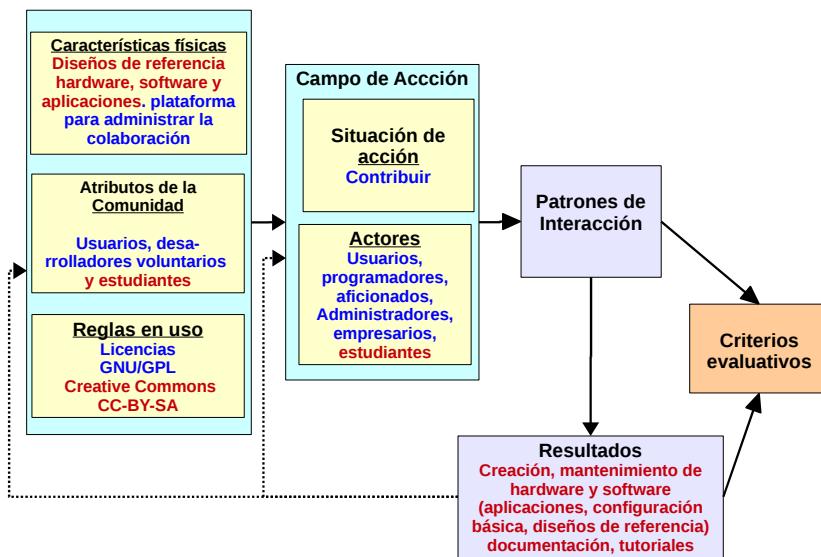


Figura 5.2.: Marco para analizar el conocimiento como bien común aplicado al movimiento hardware copyleft. Adaptación al hardware copyleft de: [4]

5.4.1. Reglas

Creative Commons⁴ es una organización no gubernamental sin ánimo de lucro que ha desarrollado una serie de licencias basadas en principios similares a los del movimiento de software libre y que pueden ser aplicadas a trabajos realizados en música, arte, video, texto y notas de clase. Para reproducir una plataforma hardware, es necesario suministrar los archivos de las herramientas CAD de diseño y fabricación (PCB, Lista de materiales), por lo que esta iniciativa resulta adecuada para distribuir este tipo de proyectos. Estas licencias permiten que el autor de un trabajo conserve la propiedad intelectual (los derechos asociados a esta) pero posibilitando su copia y distribución, con la única condición de dar créditos al autor del trabajo original y que el trabajo derivado posea la misma licencia.

Se pueden elegir diferentes permisos dependiendo de los deseos del autor; para definir dichos permisos Creative Commons proporciona una serie de preguntas que buscan determinar los derechos que se desean conservar y los que desean liberar. Las preguntas claves son:

1. ¿Los usuarios pueden copiar y distribuir su trabajo libremente?
2. ¿Es permitido a los usuarios crear trabajos derivados del contenido digital? Si es permitido,

⁴<http://creativecommons.org/>

¿Estas modificaciones deben tener la misma licencia que el trabajo original (esquema de licencia viral), o deben ser distribuidos bajo un esquema de licencias diferente?

3. ¿Es necesario atribuir el trabajo al autor original?

Estas preguntas son la base para definir un esquema de licencias modular. Existen cuatro bloques constructivos para las licencias Creative Commons:

1. *Atribución (BY)*. Se permite la distribución pero se debe dar crédito al autor.
2. *No comercial (NC)*. Se permite la distribución del trabajo sin fines comerciales, si se desea utilizar este trabajo para obtener dinero es necesaria una autorización del autor.
3. *No a trabajos derivados (ND)*. Permite la copia y la distribución del trabajo original sin modificaciones.
4. *Compartir de la misma forma (SA)*. Exige que todo trabajo derivado del uso de proyectos con este esquema de licencias deben tener la misma licencia de los trabajos originales.

Licencia CC-BY-SA

La licencia *creative commons CC-BY-SA* permite compartir, copiar, distribuir, ejecutar, comunicar públicamente la obra y hacer obras derivadas, con la condición de reconocer los créditos de la obra original de la forma indicada por el autor sin que esto implique que se cuenta con su apoyo o que apoya el uso que se hace de su obra; si altera o transforma esta obra, o genera una obra derivada, debe distribuir la obra generada bajo la misma licencia. Este esquema de licencias ha sido elegido para los productos de este estudio.

Al elegir que el resultado sea de acceso libre, conservando los derechos de autor; se proporciona un bien público que puede ser utilizado por cualquier sector de la sociedad para suplir necesidades propias; su naturaleza viral asegura que se agregarán nuevos proyectos a este bien; la labor de la comunidad será definir que proyectos cumplen con los requerimientos de calidad para ser incluidos en los repositorios públicos.

Licencia GPL

Debido a que los proyectos hardware requieren la ejecución de un software que implemente la funcionalidad requerida, es necesario diferenciar los componentes de dichos sistemas; casi la totalidad de las aplicaciones software tomadas como referencia para adaptar las plataformas desarrolladas poseen la licencia GPL, por esta razón se conservarán las libertades de este tipo de licencias para el componente software:

- Libertad de ejecutar el software para cualquier propósito.
- Libertad de estudiar como funciona el programa, y cambiarlo para hacer lo que se deseé.
- Libertad para re-distribuir copias.
- Libertad para distribuir copias de versiones modificadas.

Características

Para que un proyecto sea considerado *hardware copyleft* es necesario proporcionar los archivos necesarios para cumplir con las libertades de las licencias GPL y CC-BY-SA. A modo de ver del autor de este trabajo, la clave para garantizar dichas libertades está en suministrar la información necesaria para modificar los proyectos; para lo que es necesario poseer los conocimientos necesarios y tener acceso a los archivos de diseño.

Es posible reproducir un diseño realizado en cualquier herramienta Computer-Aided Design (CAD); sin embargo, existen dos factores que lo dificultan: el uso de herramientas propietarias y la complejidad del proyecto original. El uso de herramientas propietarias implica el pago de fuertes sumas de dinero para adquirir licencias y la complejidad del diseño puede incrementar de forma considerable el costo del prototipo. Es necesario reducir al máximo los costos para hacer atractiva la fabricación de productos propios⁵; Por lo que es recomendable utilizar herramientas abiertas en el desarrollo de los proyectos *hardware copyleft*, en el capítulo 4 se presentó una metodología de diseño que utiliza software abierto únicamente y será utilizada como referencia para el desarrollo de nuevos productos.

Archivos necesarios para modificar un proyecto existente

Para modificar o reproducir un proyecto existente es necesario suministrar:

- Componente hardware
 - *Esquemáticos*: Diagramas eléctricos que representen el diseño completo.
 - *Netlist*: Archivo generado a partir del esquemático, contiene los elementos del circuito (señales y componentes), sus interconexiones, y asociaciones con representaciones físicas (dimensiones de líneas, footprints).
 - *Layout*: Representación física de la placa de circuito impreso, utiliza el netlist como entrada,
 - *Archivos de fabricación*: Archivos que son utilizados para la fabricación de las placas de circuito impreso, son útiles para los usuarios que solo deseen reproducir el proyecto.
 - *Archivos CAD* utilizados en la creación de formas especiales en las placas de circuito impreso o fabricación de la carcasa.
 - *Lista de componentes* Listado de los componentes utilizados en la placa de circuito impreso.
- Componente software: Código fuente de:
 - *Aplicación para grabar las memorias no volátiles*
 - *Bootloader*
 - *Kernel*
 - *Sistema de archivos*
 - *Aplicaciones*
- Archivos para dispositivos lógicos programables

⁵Esto no implica que solo los proyectos más baratos hagan parte de la definición de un proyecto

- *Aplicaciones*

5.4.2. Recursos

Uno de los objetivos del presente trabajo es crear una comunidad que utilice los frutos del presente trabajo, para adquirir conocimientos relacionados con el diseño de sistemas embebidos y crear nuevos productos que den solución a necesidades locales. Para facilitar las tareas de vinculación de usuarios, comunicación entre usuarios, administración y depuración del recurso, se creó el portal público *linuxencaja*⁶; donde se proporcionan facilidades para: reunir la documentación existente; proporcionar facilidades para la creación de nuevos proyectos; almacenar los archivos generados por nuevos proyectos y mantenimiento de repositorios de código fuente y archivos de diseño de proyectos existentes. El acceso a la información generada en el desarrollo de los proyectos existentes y a los servicios de este portal no tiene restricciones de ningún tipo. A continuación se realiza una descripción de los recursos con los que cuenta la comunidad *linuxencaja*.

Herramientas

- Lista de discusión en donde participan desarrolladores hardware/software, estudiantes y miembros de la comunidad para coordinar el desarrollo de nuevos proyectos y resolver dudas sobre conceptos relacionados con el diseño de sistemas embebidos.
- Base de datos de solución de problemas asociadas a las listas de discusión.
- Sistema de control de versiones *git* que permite el trabajo multi-usuario y la creación de repositorios para los proyectos.
- wiki que permite la elaboración de tutoriales, documentación sobre temas relacionados con el diseño de sistemas digitales o reportes del proceso de diseño de nuevos proyectos.

Diseños de referencia

Se cuenta con 7 tarjetas de desarrollo (ver descripción en el capítulo 3) que cumplen con los requerimientos de los dispositivos *hardware copyleft*; estos diseños de referencia ya han sido construidos y probados en diferentes aplicaciones por lo que su funcionalidad es garantizada; todos ellos utilizan componentes que son fáciles de conseguir y no requieren grandes costos para su reproducción. En los repositorios del portal *linuxencaja* se encuentra la información necesaria para reproducir entender y modificar estas plataformas; así como el código fuente de las siguientes aplicaciones (que pueden ser utilizadas para estudiar su funcionamiento):

- *loader* y *blink*: Aplicación que permite cargar binarios en la memoria interna del SoC (*loader*) y una aplicación que despliega un mensaje por el puerto serial de la plataforma y realiza un *blink* en un LED.
- *Drivers de dispositivos*: Ejemplos para aprender a escribir controladores sencillos para el sistema operativo Linux.

⁶<http://www.linuxencaja.com>

- *Aplicaciones en espacio de usuario:* Ejemplos que permiten comunicarse con los drivers desde espacio de usuario, y ejemplos de aplicaciones sencillas utilizando las facilidades del sistema operativo Linux.
- *Aplicaciones para configuración:* Algunas plataformas de desarrollo poseen micro-controladores de 8 bits o FPGAs que pueden ser programados o configurados desde la misma plataforma. Se suministra el código fuente de estas aplicaciones.

Banco de proyectos académicos

Cada semestre se realizan proyectos que utilizan alguna de las plataformas *hardware copyleft*; cada uno de ellos realiza la documentación completa del proceso de diseño y suministra los archivos necesarios para reproducir o continuar el proyecto. Este banco de proyectos representa un recurso donde se pueden encontrar soluciones a problemas comunes de diseño y su realización ayuda a la formación de habilidades en diseño y escritura de informes técnicos.

Documentación

Linuxencaja proporciona documentación y tutoriales para:

- Artículos sobre la metodología de diseño y el programa académico generados por este trabajo.
- Proceso para proporcionar soporte a una plataforma embebida en el kernel de Linux.
- Comunicación entre tareas hardware (implementadas en PLDs) y software (implementadas en un SoC)
- Inicialización de SoCs.
- Proceso de fabricación y montaje (manual y automático) de placas de circuito impreso.
- Proceso de diseño que utiliza la metodología de diseño y el programa académico propuestos en este trabajo.
- Material para los cursos basados en la metodología propuesta (en construcción).
- Tutoriales sobre temas de interés relacionados con diseño software y hardware de sistemas digitales.

Miembros

Linuxencaja cuenta en la actualidad con diferentes tipos de usuarios: desarrolladores software, desarrolladores hardware, aficionados, estudiantes, industriales y administradores del servidor. La figura 5.3 muestra los tipos de miembros que hacen parte de la comunidad *linuxencaja*

Los desarrolladores hardware y software se encargan de generar las plataformas de desarrollo y las aplicaciones software que las controlan e implementan su funcionalidad; el número actual de desarrolladores es pequeño⁷, pero va en aumento gracias a los aportes de los estudiantes. Los

⁷En la actualidad los profesores Carlos Camargo, Jorge Sofrony y Jonatan Gómez de la Universidad Nacional; Jorge Ramón de la Universidad Industrial de Santander; Cesar Pedraza de la Universidad Santo Tomás; Fredy Segura de la Universidad de los Andes; Nelson Castillo de Google; Andrés alderón de emQbit y Rafael Zurita de la Universidad Nacional de Camahue en Argentina



Figura 5.3.: Miembros de la comunidad linuxencaja.

aficionados, son personas a las que les gusta el diseño de sistemas embebidos, el desarrollo software, y conocer el estado de la tecnología; su papel es muy importante ya que por lo general, ellos se encargan de labores de mantenimiento, depuración, participan en los foros resolviendo preguntas y ayudan a mejorar el material existente.

Los estudiantes, son miembros temporales que pueden o no continuar siendo parte de la comunidad después de cursar las asignaturas, sus aportes deben ser revisados con cuidado para verificar la calidad de su trabajo (muchos proyectos no se finalizan) y solo los mejores proyectos serán parte de los repositorios de la comunidad⁸. Gracias a la vinculación de los estudiantes a la comunidad como parte de la metodología para la enseñanza de sistemas digitales presentada en el capítulo 4 se han creado 2 nuevos proyectos que entran a ser parte de los recursos; lo que muestra el éxito de esta modalidad de vinculación. Los estudiantes están vinculados a diferentes universidades, la mayoría son de la Universidad Nacional de Colombia sede Bogotá. Por otro lado, gracias a los cursos de actualización que se dictaron en la universidades Distrital e Industrial de Santander y al curso de sistemas embebidos que se dictó en la Universidad de los Andes, existen proyectos de grado asociados a las plataformas hardware copyleft diseñadas en esta investigación; adicionalmente, la Universidad Industrial de Santander utilizará estas plataformas como herramientas didácticas en los cursos del área de electrónica digital desde este año, con lo que se espera un aumento considerable en el número de estudiantes vinculados a la comunidad linuxencaja.

La universidad de los Andes está creando un proyecto basado en los conocimientos adquiridos en esta investigación; utilizando las técnicas de fabricación y las aplicaciones software generadas en la realización de la familia de plataformas ECB_AT91, el Centro de Microelectrónica de la Universidad de los Andes (Centro de Microelectrónica de la Universidad de los Andes (CMUA)), liderado por el PhD Fredy Segura está diseñando y construyendo una Single Board Computer (SBC) con el fin de adquirir experiencia en la fabricación de sistemas digitales; lo que representa un gran cambio en la política de investigación de este grupo, caracterizado por su interés en la micro y nano-electrónica.

⁸Durante los dos períodos académicos en los que se han vinculado estudiantes a la comunidad como parte de su proyecto de curso, se han desarrollado 5 nuevas aplicaciones aplicaciones

Los industriales miembros de la comunidad se pueden dividir en dos grupos: los que hacen pequeños aportes en las listas de discusión, y no han creado proyectos nuevos, y los que participan activamente y aportan su trabajo a la comunidad. El primer grupo está conformado por empresas consolidadas desde hace varios años (a excepción de la empresa emQbit) y su comportamiento es entendible debido a su naturaleza privada. El segundo grupo esta conformado por micro-empresas que han sido creadas por estudiantes que generaron nuevos productos con los recursos suministrados por la comunidad y utilizan los resultados de los proyectos académicos.

Los administradores (2 personas Carlos Camargo y Andrés Calderón) son personas que cuentan con los conocimientos necesarios para configurar servidores y realizar mantenimiento a los servicios suministrados por el portal.

5.5. Flujo de Diseño Utilizando *Hardware Copyleft*

Para comprobar la utilidad de las plataformas de referencia *hardware copyleft* en el desarrollo de nuevos dispositivos se utilizó la plataforma *ECB_AT91* [59] como punto de partida para la construcción de un monitor de signos vitales [5] y para la creación de un sistema de medición de la calidad del servicio eléctrico [82]. En el primer caso, un estudiante de posgrado agregó un módulo de comunicaciones para comunicarse con las tarjetas de adquisición de señales médicas y suministró puertos USB host para conectar módulos de comunicación Tecnología de redes inalámbricas (WiFi), Bluetooth y 3rd generation mobile telecommunications (3G)/Global System for Mobile communication (GSM); en este caso, le tomó 6 meses fabricar la nueva plataforma y escribir el código fuente de la aplicación. La figura 5.4 muestra la plataforma original y la modificación realizada para adaptarse a la aplicación.

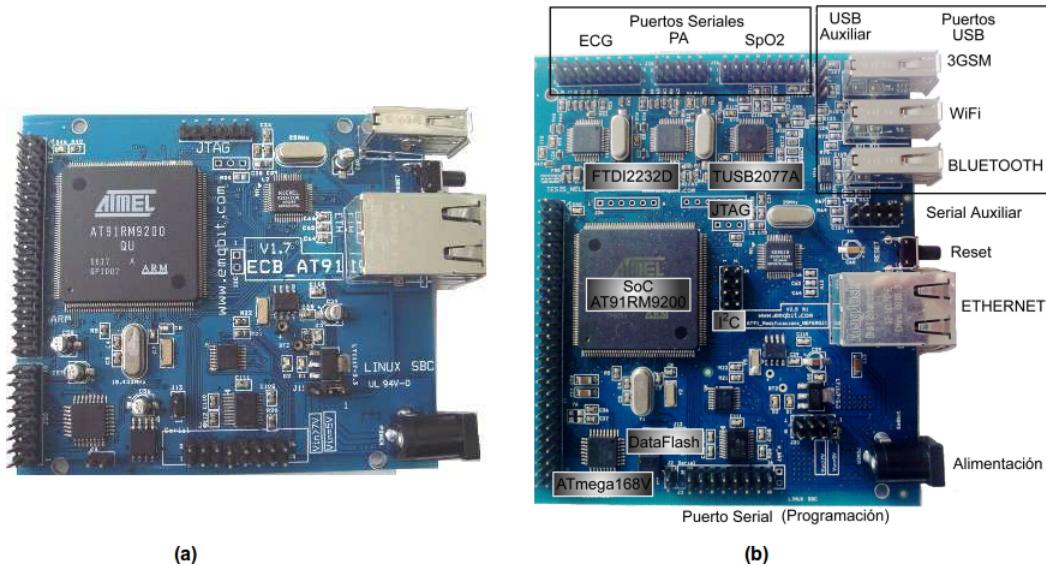


Figura 5.4.: Adaptación de la plataforma de desarrollo ECB_AT91 (a) a un monitor de signos vitales
(b) Fuente: [5]

En el segundo caso, la empresa emQbit, agregó una etapa de adquisición de señales análoga, un sistema de comunicación y un sistema de referencia geográfica basada en un Global Positioning

System (GPS), para crear un sistema que permite medir la calidad del servicio de energía eléctrica [82], empleando 6 meses en ello.

Los casos anteriores muestran una disminución drástica del tiempo necesario para la producción de nuevos productos comparados con el tiempo que demoró el desarrollo de la placa utilizada como referencia. Esto es posible, gracias a la posibilidad de utilizar los archivos de diseño de la plataforma los cuales, han pasado por un proceso de depuración y optimización, y el código fuente de las aplicaciones necesarias para la ejecución de aplicaciones Linux (Board Support Package Board Support Package (BSP)). Lo anterior representa un ahorro de tiempo considerable ya que el proceso de concepción, diseño, puesta en funcionamiento y generación de aplicaciones para esta plataforma duró 18 meses.

La figura 5.5 muestra una comparación del flujo de diseño cuando se utilizan o no plataformas *hardware copyleft*

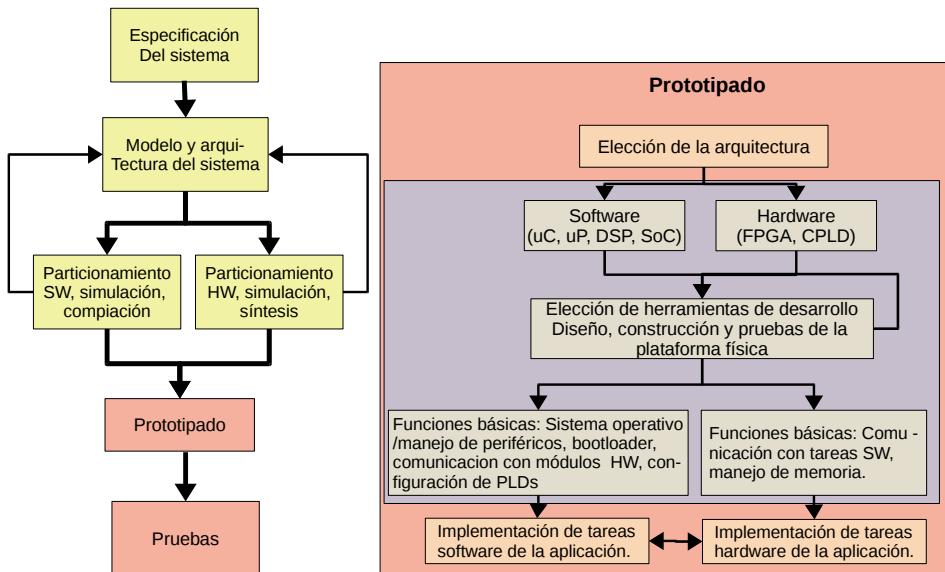


Figura 5.5.: Flujo de diseño al utilizar hardware copyleft

Como puede apreciarse en la figura 5.5 la utilización de plataformas *hardware copyleft* afecta únicamente la etapa de implementación en el proceso de diseño, esto debido a que las primeras etapas no deben tener ninguna restricción tecnológica. Cuando se utilizan las plataformas *hardware copyleft* suministradas por este trabajo es posible reducir de forma considerable las tareas necesarias (actividades a realizar en el prototipado) para la fabricación de un prototipo funcional; ya que al proporcionar los archivos necesarios para la modificación de las plataformas, se reduce de forma considerable el tiempo requerido para diseñar, construir, probar la placa de circuito impreso e implementar las funciones básicas software y hardware. Por otro lado, se suministra la información de los componentes utilizados, lo que facilita su producción, al utilizar dispositivos que no requieren firmas de contratos o la compra de un número muy grande de unidades. La ventaja de esta alternativa frente al uso de plataformas de desarrollo comerciales está en la posibilidad de ajustarse a las necesidades, lo que permite optimizar el número de componentes y reducir el costo final.

El proceso de selección de la arquitectura se limita a elegir dentro de la gama de procesadores

disponibles⁹ el más indicado (económica y funcionalmente) para la aplicación. En la tabla 5.1 se resumen las características de cada plataforma, indicando el costo de los componentes (costo de 50 unidades).

Tabla 5.1.: Características de las plataformas de desarrollo *hardware copyleft* proporcionadas por el presente estudio

Plataforma	CPU MHz	CPU \$USD	USB host	USB device	LCD	SPI/I2C/UART/	Mem Volátil \$USD	Mem no Volátil \$USD	FPGA \$USD	Mínima config. \$USD	Sistema Operativo
ECB_ARM7	33	3.5		X		X	0 - 6.5	0	NA	7.5 - 14.5	RT - eCos
ECB_AT91_V1	180	18	X	X		X	6	9	NA	33	RT - Linux
ECB_AT91_V2	180	18	X	X		X	6	9	20	53	RT - Linux
ECBOT	180	18	X	X		X	6	9	10	43	RT - Linux
ECB_BF532	400	17				X	6	9	0	32	uClinux
SIE	300	3		X	X	X	6	12	10 - 23	31 - 44	Linux
AndroidStamp	454	8	X	X	X	X	6	7	NA	21	Android

La opción más económica (De acuerdo con la tabla 5.1 es el uso de una plataforma basada en los SoC SAM7, los cuales incluyen en el mismo sustrato las memorias volátiles y no volátiles; la cantidad de memoria disponible es de máximo 256 kBytes para la memoria no volátil y 64 kBytes de memoria RAM, lo que puede ser suficiente para muchas aplicaciones; sin embargo, las aplicaciones actuales que requieren capacidades multimedia, conexión a diferentes redes y manejo de sistemas de archivos no pueden implementarse en esta arquitectura. El puerto de comunicaciones *USB host* permite la conexión de una gran variedad de periféricos (audio, comunicaciones y medios de almacenamiento) disponibles a bajos precios; reduciendo de forma considerable los costos finales. Las FPGAs son muy costosas y elevan de forma considerable el consumo de potencia del dispositivo, por lo que deben ser utilizadas en aplicaciones donde sea estrictamente necesario, ya sea para cumplir con restricciones temporales o para cumplir con funcionalidades especiales.

5.6. Discusión

El uso de plataformas *hardware copyleft* permite ahorrar tiempo y dinero en el desarrollo de aplicaciones proporcionando diseños de referencia modificables que pueden ser utilizados como base para futuros desarrollos.

Las plataformas suministradas como parte del recurso de la comunidad *linuxencaja* cubren un amplio rango de posibles aplicaciones y pueden ser utilizadas por cualquiera que este interesado ya sea para aumentar los conocimientos en el diseño e implementación de sistemas digitales, o para crear productos novedosos que satisfagan necesidades específicas de la sociedad.

⁹Es necesario aclarar que los procesadores utilizados en este trabajo son solo una muestra de la gran cantidad de referencias disponibles, por lo que es posible encontrar otros procesadores más económicos que implementen la funcionalidad deseada, las plataformas suministradas por este trabajo permiten ahorrar tiempo y dinero en la elaboración del prototipo y no necesariamente en el costo de fabricación en grandes volúmenes

La creación de este tipo de comunidades ayuda a difundir el conocimiento generado en la academia permitiendo el acceso a quienes estén interesados, lo que contrasta fuertemente con el modelo actual de capacitación que requiere el pago de elevadas sumas de dinero para obtener este tipo de conocimientos. La incorporación de centros de formación e industriales a esta comunidad posibilita identificar necesidades de la industria y ofertas de la academia contribuyendo a la creación de un vínculo entre ambos sectores.

Los vínculos entre universidad y empresa son muy difíciles de crear; durante la realización de este trabajo se pudieron establecer relaciones fuertes con 4 empresas tres de ellas creadas por egresados de la Universidad Nacional (emQbit, e-modust, Importex, em-electronics) y con dos empresas externas (SAR S.A y Microensamble S.A.), de estas relaciones se han generado pasantías, prestación de servicios a bajo costo y apoyo financiero para adquisición de equipo y manufactura de sistemas de desarrollo digitales. Se intentó establecer vínculos con grandes empresas como CODENSA y TES AMERICA, pero este tipo de empresas no están interesadas en el desarrollo de productos sino en productos finales; este conocimiento es valioso ya que para poder establecer contacto con estas compañías es necesario tener desarrollos en una etapa que permita su producción masiva.

De lo anterior se puede concluir que la mejor forma de crear vínculos entre la universidad y la empresa es haciendo que los estudiantes sean los actores principales de procesos de investigación e innovación orientados a la construcción de nuevas ideas de empresa (Spin Off Universitarias); sin embargo, el objetivo final no debe ser únicamente la creación de la empresa, sino brindar soporte para que esta pueda generar utilidades que puedan suministrar regalías a las universidades o la posibilidad de proporcionar bienes y servicios útiles en el ejercicio académico.

La experiencia con la empresa *emQbit* muestra que: es posible motivar en los estudiantes la creación de empresa; generar en ellos las habilidades necesarias para la producción de productos novedosos; y que con un acompañamiento adecuado por parte de la universidad es posible que la empresa se mantenga actualizada y sobreviva en el tiempo. La figura 5.6 muestra el modelo de transferencia que se obtuvo de la interacción con la empresa *emQbit*.

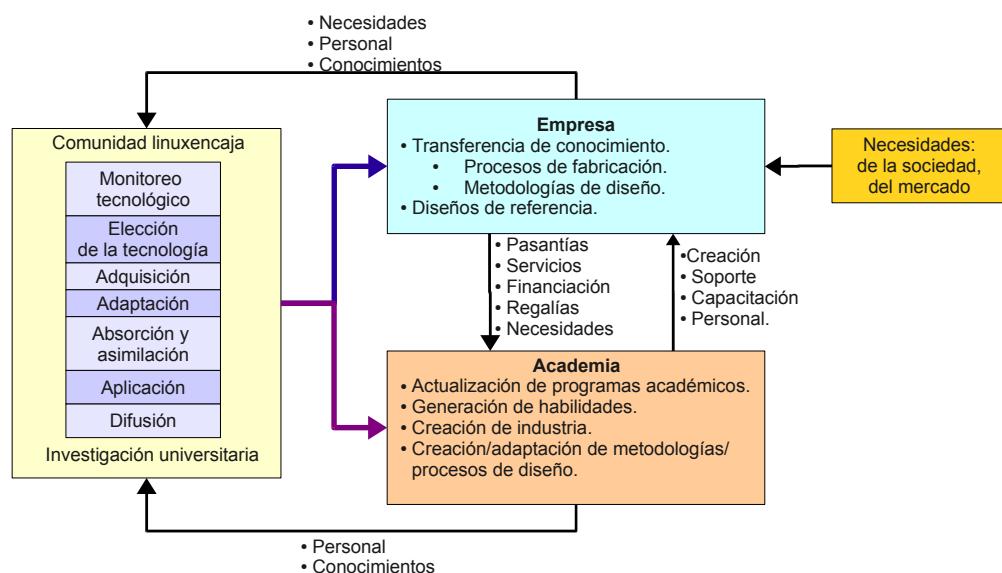


Figura 5.6.: Modelo de transferencia tecnológica aplicado a la empresa emQbit y a la academia

La figura 5.6 resume el proceso de transferencia tecnológica y de conocimientos desarrollado en este trabajo; una metodología para la adquisición/adaptación/absorción/asimilación/aplicación/difusión de la tecnología y el conocimiento asociado al diseño de sistemas digitales modernos, ejecutada inicialmente por el autor de este trabajo, y posteriormente por los integrantes de una comunidad; transfiere los conocimientos necesarios para realizar productos novedosos hacia: la empresa, proporcionando metodologías de diseño basadas en herramientas software abiertas y plataformas *hardware copyleft*, procesos de fabricación y contactos con empresas manufactureras locales y asiáticas; y hacia la academia, actualizando los contenidos de los programas académicos de tal forma que se generen las habilidades requeridas por la industria para cumplir con los requerimientos de la sociedad; utilizando metodologías de diseño y procesos de fabricación resultado del proceso de transferencia. La universidad y la academia interactúan en un intercambio de necesidades y soluciones, la empresa requiere personal calificado para cumplir con sus actividades, y la academia responde ajustando sus programas académicos para formar dicho personal, suministrando soporte técnico y capacitación; a cambio de esto, la empresa proporciona servicios a la academia para facilitar su actividad académica aportando dinero para la realización de proyectos conjuntos. Tanto la academia como la empresa aportan a la comunidad personal y conocimientos para que puedan adaptarse a las nuevas necesidades de la sociedad.

6. Conclusiones y trabajo Futuro

En este trabajo se propuso una metodología para la transferencia tecnológica y de conocimientos en el área de diseño de sistemas embebidos, la cual se centró en la educación y entrenamiento como canal para la transferencia tecnológica; una de las razones para esta elección es la dificultad que se presenta en Colombia para acceder a información especializada que ayude en el proceso de transferencia de conocimientos. Este trabajo suministra: una serie de conocimientos que pueden usarse como punto de partida para el desarrollo de aplicaciones comerciales o como material para la capacitación de personal; un canal de comunicación entre los diseñadores locales y la industria manufacturera asiática (de dispositivos digitales); una metodología de diseño basada en el uso de herramientas abiertas lo cual reduce de forma considerable la inversión de capital en herramientas de desarrollo.

La falta de políticas de estado que tracen normas encaminadas a incentivar la inversión en investigación y desarrollo, defina líneas y campos de investigación, proteja los productos locales de la invasión de productos extranjeros, regule la oferta laboral y los programas académicos, crea las condiciones adecuadas para que el país dependa casi en su totalidad de productos importados. La adopción de productos provenientes del mercado asiático; es una respuesta de la industria a la escasez de profesionales con las habilidades necesarias para crear productos locales que compitan con los importados.

Ante la situación de la industria electrónica nacional, es necesario cuestionar el papel que han tenido las facultades con programas relacionados en el desarrollo de habilidades en los actuales profesionales y preguntarse por la función que deben cumplir para dar solución a este problema. Una parte importante de las universidades utilizan tecnologías y metodologías de diseño obsoletas en los cursos relacionados con el área de electrónica digital; lo que al modo de ver del autor de este trabajo es una de las principales causas del atraso de la industria electrónica del país.

Las técnicas de evaluación tradicionales han demostrado ser ineficientes para medir las habilidades en diseño. Es necesario replantear el método de evaluación y la metodología de enseñanza en las asignaturas relacionadas con diseño en general, el método propuesto en este trabajo ha demostrado su eficacia en la generación de hábitos de continuo estudio y auto-aprendizaje; sin embargo, uno de los grandes inconvenientes para su aplicación son los malos métodos de estudio que poseen los estudiantes; por este motivo, es necesario crear actividades que motiven a los estudiantes a realizar las lecturas y actividades programadas en el curso. Enfrentar al estudiante a un problema durante todo el período académico permite la aplicación de todo el proceso de diseño en la solución de un problema real, haciendo que los conceptos sean asimilados en diferentes tiempos, dependiendo de las habilidades individuales; lo que no se puede obtener cuando se realiza una única evaluación. El plan de estudios propuesto en este trabajo proporciona una solución novedosa a los problemas mencionados anteriormente, brindándole al estudiante la posibilidad de adquirir habilidades que pueden ser utilizadas en el ejercicio de su profesión al simular el entorno laboral y proporcionar herramientas útiles con las que se pueden resolver problemas reales.

Las asignaturas del área de electrónica digital son de vital importancia en el proceso de formación de los ingenieros electrónicos, porque su campo de aplicación cubre todas las actividades

de los profesionales en electrónica (control, automatización, comunicaciones, procesamiento de señales y de imágenes, etc) y porque es la asignatura más adecuada para enfrentar a los estudiantes al proceso de implementación de sistemas. El plan de estudios propuesto en este trabajo puede ser tomado como referencia para adaptar su metodología a otras asignaturas.

La aplicación de la metodología propuesta al entorno académico reveló la deficiencia de los programas académicos del área de electrónica digital en las universidades colombianas. Difundir los resultados de este estudio a todos los centros de formación permite nivelar los contenidos curriculares de los diferentes centros de formación lo que se traduciría en una mejor capacitación de los profesionales de las carreras relacionadas. Por este, debe existir una evaluación continua de los planes académicos, para que se adapten a las necesidades del sistema productivo local y a los últimos avances de la tecnología. La Universidad Nacional de Colombia, como principal centro de formación del país debe liderar estos procesos de difusión y evaluación para garantizar la creación de estas habilidades en todos los profesionales del sector.

Es una realidad que a corto plazo Colombia no puede competir con la industria manufacturera asiática, pero si podemos utilizarla para construir productos diseñados en el país que satisfacen necesidades locales. En el país se pueden construir prototipos de muy buena calidad, por lo que la mejor opción es la fabricación local del prototipo y la producción en masa en el exterior. Se debe crear conciencia en la industria de las ventajas de tener productos diseñados localmente, resaltando los servicios adicionales que pueden generarse al personalizar estos productos y proporcionar servicios derivados de su uso. Los centros académicos deben trabajar en problemas del entorno local que, aunque no tienen mucho reconocimiento a nivel internacional si refleja un grado de compromiso con el entorno social en donde ellas operan.

Utilizar la ingeniería inversa sobre productos comerciales de consumo masivo es la opción más económica de obtener conocimientos sobre: como están construidos; como funcionan y como se programan dispositivos comerciales. Su utilización (de las plataformas comerciales), facilitó el aprendizaje del funcionamiento de la tecnología, permitiendo el estudio de metodologías de diseño y la evaluación de herramientas de desarrollo; reduciendo el tiempo y el costo de desarrollo de plataformas propias diseñadas para dicho fin. En este trabajo se realizó un estudio gradual de las características de un sistemas digital, comenzando con su arquitectura, siguiendo con su programación y finalizando con las técnicas de fabricación y producción masiva.

Este trabajo demostró que es posible que la academia motive la creación de empresas de base tecnológica; sin embargo, lo realmente importante es hacer un acompañamiento que permita su consolidación. Adicionalmente, es necesario dar a conocer la forma de realizar planes de innovación y planes de negocio que permita la búsqueda de financiación para el desarrollo de nuevos productos; para esto es necesario incluir en el programa de estudios asignaturas en donde se enseñen estos temas, pero que estén articulados con la naturaleza de la profesión; es importante concienciar a los estudiantes de la importancia de estas asignaturas en su formación integral.

Toda actividad financiada por el gobierno que busque transferir, absorber y asimilar conocimientos debe proporcionar la documentación necesaria para que sea utilizada de forma gratuita por cualquier sector de la sociedad. Los centros de formación públicos deben proporcionar cursos gratuitos y no limitar la difusión de conocimiento a las personas que pueden pagar por ellos. Al permitir la utilización de los resultados de esta investigación por quien este interesado y considerar este conocimiento como un bien público, se aumentan las posibilidades de difusión. La clave de todo proceso de transferencia es la formación de personal que permita su asimilación y adaptación al entorno local. La Universidad Nacional de Colombia es el centro de formación superior más importante de Colombia; aquellos planes que inicie serán vistos y tomados como modelo en todo

el país. Esta situación privilegiada trae consigo la obligación de hacer partícipe a la sociedad de los resultados de su actividad académica; por esta razón, es un deber hacer que los contenidos de esta investigación estén disponibles a: otros centros de formación para que puedan realizar la actualización de sus planes de estudio, profesionales que deseen mejorar sus habilidades en el diseño de sistemas digitales y a empresas que deseen mejorar las habilidades de sus empleados.

La metodología presentada en este trabajo puede ser utilizada como referencia para trabajos futuros, ya que permite dar solución a problemas que no pueden ser resueltos por trabajos de doctorado realizados en el exterior. Este trabajo proporciona un buen punto de partida para actualizar los planes de estudio relacionados con la electrónica digital y con una adecuada difusión es posible que en un tiempo relativamente corto sea aceptado por un gran número de centros de formación.

6.1. Trabajo Futuro

Es necesario hacer que los diferentes sectores de la sociedad sean conscientes de la importancia de esta tecnología y la capacidad que tiene el país para desarrollar dispositivos comerciales con ella. También es muy importante que la sociedad prefiera y proteja los productos realizados internamente; para esto se debe continuar con la difusión de los resultados de este estudio mostrando que es posible el desarrollo de productos tecnológicos que usan tecnología de punta.

Hasta el momento, el proceso de difusión se ha realizado con algunas de las universidades bogotanas más reconocidas y con la Universidad Industrial de Santander. Sin embargo, es necesario hacer llegar este conocimiento a todos los centros de formación con programas relacionados con la electrónica digital; especialmente a los ofrecidos por centros de educación poco consolidados. Para facilitar este proceso, se está trabajando en la creación de cursos en línea en el área de diseño digital de tal forma que cualquier centro de formación los puedan utilizar. La creación de estos cursos será realizada por el autor de esta investigación con ayuda de profesores y estudiantes de las universidades que utilicen este material en sus cursos.

Es necesario realizar una difusión más amplia de los resultados de este trabajo hacia la industria, aunque al experiencia con la empresa emQbit fue muy enriquecedora, es importante realizar una búsqueda de empresas similares y transferir los conocimientos generados en este trabajo para suministrarles herramientas para que puedan competir con los productos importados.

Utilizar herramientas tecnológicas para la enseñanza en centros de educación básica y media hace que desde niños los estudiantes se familiaricen con estas tecnologías y entiendan su importancia. Se escogió la robótica como medio de difusión, debido a que su utilización en la educación básica y media ha venido en aumento en los últimos años, y su uso adecuado permite el desarrollo de habilidades asociadas al constructivismo (aprender haciendo), y al aprendizaje significativo (aprendizaje relacionado con necesidades, intereses propios) haciendo que el estudiante aprenda el valor de la nueva tecnología y como esta puede ser utilizada para dar solución a problemas comunes o para desarrollar habilidades en áreas de interés. Existen dos formas de utilizar la robótica en la educación : La robótica como objeto de aprendizaje: enfocado a aspectos relacionados con el robot como construcción, programación e inteligencia artificial y la robótica como herramienta de aprendizaje: visto como proyecto interdisciplinario que involucra ciencias, matemáticas, tecnologías de la información y comunicaciones. Linuxencaja cuenta con 2 plataformas robóticas diseñadas para ser utilizadas en esta estrategia de difusión. Y se desea utilizarlas como material de enseñanza en las secretarías de educación de todo el país para reforzar en los estudiantes su pensamiento sistemático y las habilidades necesarias para la creación de algoritmos; a través del uso del entorno

de programación gráfica *SIE Blocks* y la plataforma robótica SIEBOT. *SIE Blocks* está basado en el proyecto openblocks [83] desarrollado en el MIT, el cual a su vez sigue el trabajo en entornos de programación gráficos Starlogo [84]; el que busca reducir las barreras que presentan los lenguajes de programación tradicionales al ser utilizados por principiantes.

Como parte de la aplicación de la metodología propuesta se creó una línea de investigación y dos asignaturas para la nueva maestría en ingeniería electrónica; un trabajo importante es la formulación de proyectos de posgrado que ayuden a la consolidación de esta línea de investigación y a la aplicación de los conocimientos generados al desarrollo de proyectos que vinculen a empresas locales y estudiantes de maestría.

A. Relaciones con la industria y la Academia.

A.1. Introducción

En este anexo se enumeraran las actividades realizadas con centros académicos y con empresas electrónicas del país; y de esta forma mostrar como este estudio modificó las conductas de los profesionales relacionados con el área de diseño electrónico.

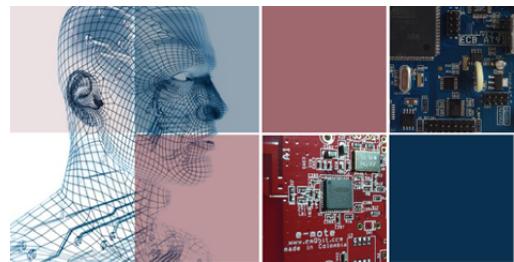
A.2. Relaciones con la industria

A.2.1. emQbit Itda.



emQbit

Linux Powered Devices Company



EmQbit Ltda es una empresa colombiana líder en investigación y desarrollo de sistemas electrónicos en la región. Combinando creatividad y experticia técnica emQbit ofrece a la industria y a la academia soluciones que están al nivel del estado del arte en el desarrollo de dispositivos electrónicos. EmQbit basa su modelo en el uso de tecnologías y métodos de fabricación de vanguardia, en la innovación y en una constante apropiación de conocimiento y su posterior difusión en la sociedad colombiana.

EmQbit es una idea que actualmente crece como el esfuerzo de gente que cree que las oportunidades no aparecen por si solas sino que se deben generar. EmQbit crea soluciones de Hardware y Software a través de desarrollos basados en Linux como sistema operativo sobre sistemas embebidos.

EmQbit busca integrar tecnologías en sistemas embebidos para generar aplicaciones que impacten varios sectores del mercado potencial. Para esto se tienen varias estrategias diseñadas:

- Una constante investigación por parte de todos los miembros de la compañía sobre de las tendencias actuales del mercado.
- Canales de comunicación con nuestros clientes actuales y con los potenciales, para mejorar las soluciones existentes y para introducir nuevos desarrollos.
- Proponer tecnologías de punta para los proyectos de nuestros clientes. Nuestra experiencia en el área y las tecnologías modernas nos permiten entregar a nuestros clientes soluciones de la más alta calidad

EmQbit quiere establecer un entorno de trabajo agradable, creando una empresa llamativa para los profesionales colombianos, donde la gente se sienta a gusto trabajando, desarrollando, generando ideas.

Existe un gran mercado en el área y sabemos que somos competitivos. A futuro emQbit dominará el diseño y la producción de sistemas embebidos en Latinoamérica, y será un participante importante en la producción de soluciones embebidas a nivel mundial.

Mercados

Los mercados en los que emQbit tiene un espacio son:

- Comodities: Computadoras de una board (Single Board Computers).
- High-end appliances: cámaras, medidores de calidad del servicio de suministro eléctrico, telemedicina.
- Servicios: Desarrollo de hardware y software, consultorías.
- Video Analítico para aplicaciones en seguridad, tráfico (control, vigilancia, amonestaciones)
- Academia: starter kits, robótica educativa, capacitación.

Productos de emQbit



1. Computadoras en una tarjeta (SBC)

EmQbit desarrolla computadoras en una sola tarjeta (SBC por sus siglas en inglés) basadas en procesadores de 32 bits, con soporte a herramientas de desarrollo de última generación y Sistemas Operativos de alto nivel, como Linux. Las SBCs de emQbit integran en un área inferior a 10x10cm, procesador, memoria RAM, almacenamiento no volátil, interfaces USB, Ethernet, interfaces análogas, entre otras.

EmQbit es la única empresa local con experiencia en el desarrollo de este tipo de dispositivos, creó la primera SBC colombiana de 32 bits capaz de correr el sistema operativo Linux y posteriormente desarrolló la primera SBC colombiana para codiseño de hardware y software basada en FPGA. Actualmente se encuentran en desarrollo nuevas SBC con procesadores más potentes y con capacidades gráficas.

Los SBC de emQbit pueden ser integrados a gran cantidad de aplicaciones como sistemas de control y monitoreo Industriales, sistemas de seguridad electrónica, puntos de venta, telefonía IP, etc.

ECB_AT91 V1

La V1 está basada en un procesador (de 32bits) ARM9 a 180MHz de Atmel. El bajo número de componentes y la simpleza del circuito impreso la hacen confiable y económica. Ha sido diseñada para ejecutar Linux como sistema operativo y para aplicaciones orientadas con conectividad a Internet. Cuenta con interfaz Ethernet, compatibilidad con adaptadores USB/WiFi, incluye soporte a módem de telefonía celular 3G/GSM. El puerto USB permite el uso de memorias, cámaras y casi cualquier otro dispositivo USB. A través de un microcontrolador AVR de 8bits es posible realizar adquisición y procesamiento digital de señales.



La V1 puede ser usada en varios sectores industriales, en generación de energía, control de procesos de manufactura, administración de tráfico, administración de sistemas de impresión, infraestructura de comunicación, hosting de sitios web, recolección de datos,

prueba de equipos de laboratorio, sistemas de seguimiento, sistemas de interconexión entre redes heterogéneas, etc. Estas con las especificaciones de la ECB_AT91 V1:

- Procesador ARM9 (Atmel AT91RM9200) @ 180 MHz
- Soporte para Linux embebido
- hasta 64MBytes de SDRAM (Soporte para 8M/16M/32M/64M)
- 1 slot para SD/MM
- 1 10/100 Interfaz Ethernet
- USB 2.0 host
- Múltiples puertos seriales
- Puertos de expansión
- Reloj respaldado por batería
- Soporte JTAG

ECB_AT91 V2

Evolución de la V1, la V2 agrega al diseño anterior un dispositivo de hardware reconfigurable del tipo FPGA para permitir la ejecución de algoritmos de procesamiento de señales y de imágenes en tiempo real, encripción por hardware, implementación de coprocesadores, codecs de audio y video, etc.



El uso del FPGA permite la creación de periféricos adicionales, tales como controladores de vídeo, múltiples puertos seriales y MACs (Ethernet, USB o CAN). La V2 también tiene como propósito servir como herramienta de estudio de técnicas de co-diseño Hardware/Software, en donde es posible explotar simultáneamente las capacidades de un sistema operativo como Linux y el alto nivel de paralelismo y velocidad de una FPGA.

2. Desarrollos Basados en SBCs de emQbit

Las V1 y V2 pueden ser usado en un rango muy grande de aplicaciones. emQbit ha usado sus SBCs como base de algunos dispositivos. Dentro de los desarrollos más importantes podemos presentar:

- Gateway

Este sistema permite tener la conexión entre dos redes heterogéneas: la red celular GSM/GPRS y la red Ethernet.



- Módem GSM/GPRS de 4 bandas
- Ethernet 10/100
- Linux embebido
- Procesador ARM 9
- Memoria RAM de 32MB
- Memoria no volátil de 1GB
- 3 puertos USB
- Soporte para Bluetooth
- Soporte para WiFi

- Sistema de Seguimiento GPS-GSM

Sistema de seguimiento vehicular GPS que utiliza la red celular GSM/GPRS para el envío de datos.



- Módulo GPS
- Módem GSM/GPRS de 4 bandas
- Linux Embebido
- Procesador ARM9
- Memoria RAM de 32MB
- Memoria volátil de 256MB

- Switch de RadioFrecuencia

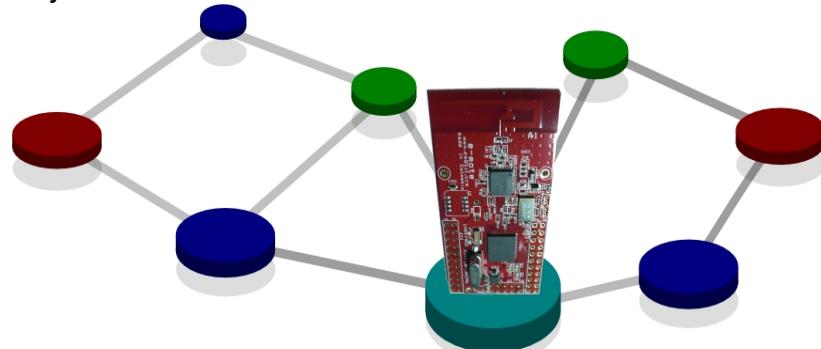
Este sistema permite tener control sobre un switch de radiofrecuencia de 4 canales a través de una red ethernet, utilizando una interfaz web o comandos SNMP.



- Mini-LCD de texto
- Switch de RadioFrecuencia de 4 canales
- Controlado por los protocolos SNMP y HTTP

3. Redes Inalámbricas de Sensores (WSN)

Una red de sensores inalámbricos (WSN) está compuesta por dispositivos terminales o nodos sensores que colectan la información del entorno. La red WSN puede conectarse a otras redes a través de un gateway para realizar monitoreo y gestión remota. Al gateway llega la información colectada por toda la red. En muchas ocasiones estos gateways deben ser dispositivos computacionalmente más poderosos que los nodos, siendo implementables con sistemas como la ECB_AT91 V1. Este tipo de redes están orientadas a la autoconfiguración y son robustas debido a la gran cantidad de sensores que mantienen conexiones seguras y datos redundantes en la red.



Las aplicaciones de la redes WSN se pueden clasificar en tres grupos: recolección de datos de un entorno, monitoreo de seguridad y seguimiento de nodos. emQbit a desarrollado un nodo para este tipo de redes, orientando su aplicación especialmente al monitoreo de variables ambientales, tales como, temperatura, humedad relativa y luminosidad, en pro de encontrar un nicho en el mercado de la agricultura, o en la integración con los sistemas de monitoreo en edificios inteligentes.

Estas son las especificaciones del nodo e-mote:

- Transceiver CC240 a 2.4GHz
- Antena integrada F-inverted en tecnología de microcintas
- Microcontrolador AVR Atmega128L

- Integración con redes Ethernet, WiFi, Bluetooth, etc.
- Implementa el estándar IEEE 802.15.4 con soporte para aplicaciones Zigbee
- Compatible con nodos MicaZ
- Operación a 2.5V-4.0V

4. Sistemas para Interfaz con Humanos (HMI)

EmQbit está desarrollando dispositivos que puedan ser utilizados como sistemas de interfaz con humanos (o computadoras de panel). El objeto de este tipo de dispositivos es el de proporcionar una interfaz gráfica para interactuar con el usuario, como ejemplos de este tipo de sistemas podemos nombrar los monitores de señales biológicas, Controladores y supervisores de procesos industriales, Puntos de Venta, Sistemas de Seguridad.

La principal característica de estos dispositivos es el bajo costo, en comparación a la utilización de computadores de escritorio, ya que estos dispositivos solo realizan una tarea específica para la que fueron creados, lo cual los hace muy eficientes.



Las especificaciones técnicas de estos dispositivos se listan a continuación:

- Linux Embebido
- Pantalla de toque LCD a todo color de 320x340 pixeles
- Procesador ARM9
- Memoria RAM 32MB
- 2 puertos USB
- Soporte para Bluetooth
- Soporte para Wifi

5. Kits de Robótica

EmQbit cree necesario que la empresa privada colombiana se comprometa en la reducción de la brecha tecnológica en el País. Como parte de su labor, emQbit ha creado kits de robótica educativa que pueden ser usados dentro de gran variedad de programas

académicos, los cuales pueden ir desde cursos básicos para estudiantes de educación media hasta cursos de pregrado y postgrado en ingenierías Eléctrica, Electrónica, Sistemas y Mecatrónica.



ECBOT es el sistema de robótica más avanzado desarrollado por emQbit, basado en procesadores ARM9 de 32 bits y FPGAs; capaz de ejecutar Linux embebido, posee un sensor de imagen, comunicación WiFi, 8 sensores de proximidad, 7 entradas analógicas, y la capacidad de controlar 10 servomotores.

Para introducir a los estudiantes de primaria y bachillerato al mundo de la robótica y la electrónica emQbit diseñó un kit de desarrollo basado en procesadores ARM7 de 32 bits. El cual es capaz de correr Sistemas Operativos de Tiempo Real (RTOS por sus siglas en inglés). Incluye además un enlace inalámbrico a 2.4GHz, controladores para 4 motores DC, control para 8 servomotores, interfaz infrarroja y 6 entradas analógicas.

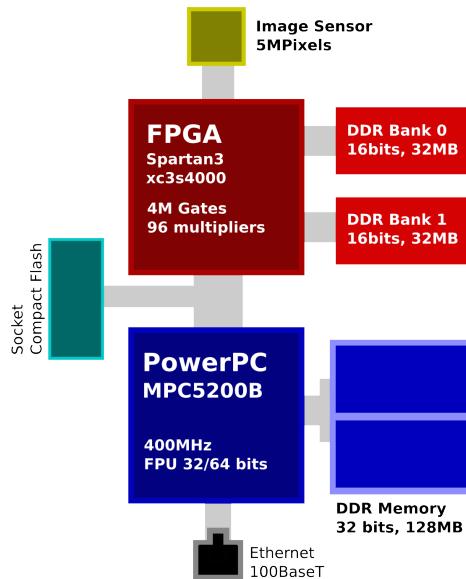
6. emVision

emQbit está desarrollando una cámara inteligente de red programable, en donde es posible implementar algoritmos de procesamiento de imágenes y video usando la flexibilidad y poder computacional que puede dar un procesador con unidad de punto flotante combinado con un FPGA.

Gracias a la FPGA usted podrá realizar procesamiento en tiempo real de imágenes y video en esta cámara. La programación in-situ le permite cambiar la funcionalidad de la cámara remotamente utilizando la red, sin necesidad de estar haciendo reprogramación completa.

Debido a que la cámara corre Linux como sistema operativo, usted no necesita comprar kits de desarrollo. Nosotros la entregamos lista y con las herramientas que usted necesitará para usarla y programarla. Una cámara con estas características tiene mercados objetivo muy interesantes, desde la detección y reconocimiento de rostros, pasando por el control automático de parqueaderos, hasta la detección automática de elementos peligrosos en espacios públicos.

Características Técnicas:



Procesador

- PowerPC de 400MHz con unidad de punto flotante de 32/64 bits
- Cachés independientes de datos e instrucciones (16K/16K)
- 128 MBytes DDR RAM
- Ethernet 100BaseT
- USB Host

FPGA

- Xilinx's Spartan3 FPGA (XC3S4000) - con 4 millones de compuertas
- 96 multiplicadores (300MHz)
- 96 bloques de SRAM de doble puerto (2KBytes/bloque)
- 2 o 3 bancos de memoria DDR de 16 bits a 133MHz

Optica

- Sensor de 5 MegaPixels, 1/2.25"
- Compatibilidad con lentes para montaje tipo C

7. Servicios

EmQbit cuenta con profesionales expertos en sistemas operativos, con conocimientos profundos en programación, electrónica digital, arquitectura de computadores, sistema operativo Linux y sistemas embebidos que podrán apoyar de manera efectiva sus proyectos

desarrollando tanto el Software como el Hardware y la interconexion entre ellos. emQbit utiliza las herramientas de desarrollo Software y Hardware más utilizadas por los desarrolladores en países industrializados y hace parte activa de las listas de discusión aportando soporte y haciendo contribuciones a estas herramientas.

Desarrollamos hardware a la medida para sistemas embebidos y personalizamos soluciones basadas en nuestros productos. Dada la flexibilidad de nuestros sistemas, podemos ofrecer soluciones de hardware en tiempos cortos, que le permitirán aprovechar oportunidades de negocio.

EmQbit está en la capacidad de proveer los ingenieros especializados en desarrollo hardware y software para sistemas embebidos, para que trabajen en proyectos de impacto en la industria.

NUESTROS CLIENTES

En la industria

• ***Voting Solutions Colombia S.A.***

NIT 900146241-0

Avenida el Dorado # 68C-61 Of 531 Bogotá

Tel: +57(1) 4273433

Bogotá, Colombia

• ***SAR: SISTEMAS ASESORIAS Y REDES S.A.***

Contacto: Ing. Luis Fernando Samaca

NIT: 860.512.079-9

CARRERA 15A No. 124-84

Tel: +57(1) 6298920

Bogotá, Colombia

• ***TEMPERATURE@LERT***

Contacto: Harry Schechter

139A Charles St., #345

Boston, MA 02114

• ***Onderzoeksgroep Embedded Systems***

Contacto: Patrick Pelgrims

Hogeschool voor Wetenschap & Kunst

Campus de NAYER Instituut

Jan De Nayerlaan 5

B-2860 Sint-Katelijne-Waver
Belgium

•CEAROBOT

Contacto: Stefano Parusso
S.S. 352 Udine-Grado Km 3,6
33050 Lauzacco
Pavia di Udine - Italy
Tel +39 0432 675768
Fax +39 0432 655127

•TES America Andina

Contacto: Orlando Saavedra
NIT 830-053-662-4
CLL 99 No 11B-20
Tel: +57 300 567 6078
BOGOTA

En la Academia

•UNIVERSIDAD INDUSTRIAL DE SANTANDER

Contacto: Jorge Ramón
Tel: +57(300)3672764
BUCARAMANGA, SANTANDER, COLOMBIA

•UNIVERSIDAD DISTRITAL DE COLOMBIA

Contacto: Giovanni Bermudez
Director del Centro de investigaciones
gbermudez@udistrital.edu.co

•ESCUELA NAVAL ALMIRANTE PADILLA

Contacto: Alejandro Mejía Ayala Base Naval ARC Bolívar
Tel: +57 315 752 0662

•UNIVERSIDAD SANTO TOMAS DE AQUINO

Contacto: Jaime Vitola
Prof. Facultad de Ingeniería Electrónica
Tel: +571 3164686707

•UNIVERSIDAD NACIONAL DE COLOMBIA

Contactos: Carlos Camamrgo Bareño Dpto de Ing. Eléctrica y Electrónica ext 11126
Jonatan Gómez Dpto de Ing. Sistemas e Industria
Jorge Sofrony Dpto de Ing. Mecánica y Mecatrónica
Eduardo Romero Facultad de Medicina ext 15025

A quien corresponda

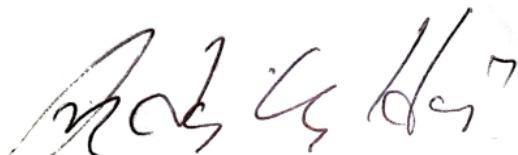
Como gerente general de la empresa emQbit certifico que el profesor Carlos Iván Camargo Bareño ha colaborado en diversas tareas de diseño, soporte y capacitación, los productos realizados en conjunto con el profesor Camargo poseen la licencia Creative Commons BY-SA la que permite su uso, reproducción y modificación aún para uso comercial, con el único requisito de dar crédito al autor del trabajo original y publicar los trabajos derivados bajo la misma licencia.

Esta filosofía de hardware abierto es el eje de nuestra empresa, de tal forma que la mayor parte de nuestros productos poseen la licencia CC-BY-SA; esto debido a que por un lado, gran parte de ellos se deriva de tarjetas de desarrollo elaboradas por el profesor Camargo y por otro lado, uno de los objetivos de la creación de emQbit es ayudar a la difusión del conocimiento necesario para diseñar e implementar sistemas digitales que utilizan tecnología de punta.

Adicionalmente, el profesor Camargo participó en la creación de emQbit proporcionando las plataformas diseñadas por él para que fueran utilizadas como diseños de referencia para la creación de nuevos productos, de igual forma, proporcionó soporte técnico para la implementación de la funcionalidad requerida.

Como parte del proceso de transferencia tecnológica entre el estudio realizado por el profesor Camargo y emQbit se realizaron 2 pasantías que permitieron capacitar a estudiantes en el diseño de sistemas digitales.

Finalmente, gracias al apoyo del profesor Camargo, nuestra empresa ha podido posicionararse como la única en Colombia que realiza el proceso de diseño completo desde la concepción hasta la implementación de sistemas digitales que utilizan herramientas abiertas y tecnología de punta. Por esta razón, emQbit colabora en los procesos de importación de materiales y suministros necesarios para la realización de proyectos de investigación de los departamentos de eléctrica y electrónica, mecatrónica y sistemas de la Universidad Nacional, estos servicios son gratuitos. Adicionalmente, la empresa ha financiado parte de los desarrollos del profesor Camargo, los cuales han sido liberados bajo la licencia CC-BY-SA.



Andrés Felipe Calderón de Restrepo
Gerente General emQbit LTDA.
C.C. 79.684.619 de Bogotá

A.2.2. e-modust

e-modust, es una empresa creada por los egresados de la carrera de Ingeniería electrónica Edison Olarte e Israel Santoyo y su principal campo de aplicación es la creación de aplicaciones para la medición de campos eléctricos y magnéticos asociados a tormentas eléctricas [85]. Su producto comercial registrado en Colombia es un molino de campo eléctrico [86]; en el primer prototipo se utilizó la plataforma *ECB_AT91* (desarrollada en este estudio) como núcleo del sistema, encargado de la comunicación con el sensor de campo, procesamiento de esta información y despliegue de la misma; la comunicación con el usuario se realizó a través de un servidor Hypertext Transfer Protocol (HTTP) utilizando conexión inalámbrica y cableada. La utilización de la plataforma *ECB_AT91* redujo de forma considerable el tiempo necesario para el diseño de esta aplicación, ya que al utilizar los archivos de diseño y los tutoriales sobre su manejo, esta empresa pudo terminar un prototipo en cerca de 6 meses; este prototipo permitió que la empresa CODENSA invirtiera en un proyecto conjunto con la Universidad Nacional para instalar molinos de campo en zonas donde las descargas eléctricas deben ser monitoreadas de forma constante.

A.2.3. Importex

Importex, es una empresa creada por egresados de la Universidad Industrial de Santander y en la actualidad es el encargado de distribuir las plataformas de desarrollo SIE, utilizadas en los cursos de la línea de electrónica digital de la Universidad Nacional de Colombia sede Bogotá y en la UIS. Esta empresa descargó los archivos de diseño de dicha plataforma del portal *linuxencaja* y fabricó un pequeño lote de 100 unidades en Taiwan, el contacto con las empresas manufactureras en este país fue proporcionado por *linuxencaja* y gracias a esta asociación los estudiantes de ambas universidades pueden utilizar estas plataformas en sus cursos.

A.2.4. em-electronics

em-electronics fué una empresa creada por los ingenieros electrónicos de la Universidad Nacional Ernesto y Alejandro Poveda; esta empresa utilizó plataformas de desarrollo abiertas (no están en la actualidad en la página de *linuxencaja*) diseñadas como parte del estudio que finalizó en el desarrollo del presente estudio, estas plataformas ayudaron a la actualización tecnológica en la carrera de ingeniería electrónica de la Universidad Nacional pasando del uso de componentes discretos (familias 74 y 40) al uso de CPLDs y FPGAs, desafortunadamente esta empresa no siguió con el suministro de este tipo de plataformas.

A.2.5. Microensamble

Microensamble SAS es una empresa Bogotana fundada por Jorge Dominguez dedicada a la fabricación y verificación de placas de circuito impreso y montaje automático de componentes electrónicos, en Colombia solo existen dos empresas con la capacidad de realizar fabricación y montaje automático de PCBs con la calidad de las empresas asiáticas, micro-ensamble es una de ellas. La alianza con microensamble ayudó a esta empresa a ajustar los procesos de fabricación de PCBs, este proyecto proporcionó diseños de referencia que utilizaban medidas adecuadas para este proceso.

Por otro lado, este proyecto realizó la capacitación a empleados de microensamble en el diseño de sistemas digitales utilizando Linux, como resultado de esta capacitación se generó un producto

que permite la comunicación segura utilizando mensajes de texto a través de la red celular.

Gracias a esta alianza, los estudiantes de la Universidad Nacional pueden realizar circuitos impresos a bajos costos utilizando equipos adecuados para esto, lo que elimina la costumbre anterior de realizar los PCBs de forma artesanal lo que complicaba la implementación.

A.2.6. SAR SA

Las relaciones con SAR SA. se realizaron de forma indirecta a través de la empresa emQbit; SAR S.A. junto con su cliente y aliado Angelcom S.A, es el concesionario de la Fase I del sistema de recaudo TransMilenio S.A en la ciudad de Bogotá D.C. Aportan la plataforma tecnológica de recaudo para 68 estaciones actualmente en funcionamiento en 4 vías troncales (Calle 80, Av. Caracas, Avenida Jiménez, Autopista Norte y Calle 13). Adicionalmente son parte de la Unión Temporal Fase II, concesionario de recaudo para la fase II del sistema TransMilenio, responsable de la operación de 47 estaciones, distribuidas en tres troncales en la ciudad de Bogotá (NQS, Av. De las Américas y Av. Suba).

emQbit LTDA. ayudó en la puesta en marcha de este sistema de recaudo, aportando la programación de los dispositivos que controlan el acceso a las estaciones de TransMilenio; la ayuda de emQbit fue de vital importancia en este proyecto ya que es la única empresa en Colombia que trabaja con Linux en sistemas embebidos; adicionalmente, la unidad de procesamiento de la plataforma de recaudo de SAR S.A posee el mismo procesador utilizado por la familia de plataformas ECB_AT91, y muchas pruebas de funcionamiento fueron implementadas primero en las plataformas hardware copyleft desarrolladas en esta investigación. A continuación se muestra la carta que califica el trabajo realizado por emQbit como *muy confiable*.



Bogotá, 31 de diciembre de 2010
GG-DC-535-2010

Señores
EMQ BIT
Atn. Andrés Felipe Calderón
Calle 44 No. 8-50 int. 202
Tel. 3002753666
Ciudad

Asunto: Resultados calificación Proveedores 2010

Respetados señores:

De acuerdo al procedimiento de calidad y teniendo en cuenta los parámetros estipulados en el formato de calificación de proveedores, la cual se realizó en diciembre remitimos los resultados obtenidos en el periodo correspondiente al año 2010.

Queremos felicitarlos por el nivel de calificación muy confiable y haber cumplido con la meta establecida de 90 puntos sin embargo, los invitamos a mejorar en cuanto a la implementación de Sistema de Gestión de Calidad y en la parte comercial – crédito- puntos débiles que se reflejan en la calificación que estamos anexando.

Cordialmente,

JORGE IGNACIO LOPEZ CARDOZO
Representante Legal

Copia: archivo
Consecutivo



Código N° SC 1384-1

Integración y desarrollo de soluciones de tecnología de la información para la optimización y manejo del suministro de recursos, para la mejora continua de la eficiencia de los procesos de trabajo y control de acceso en lugares con riesgos sanitarios, vehículos, sistemas para la identificación y autorización de clientes y proveedores de servicios, desarrollo de software, instalación, mantenimiento de sistemas informáticos, consultoría y soporte.

NTC ISO 9001:2000

A.3. Relaciones con la academia

Como parte de la estrategia de difusión se realizaron las siguientes actividades con grupos de investigación en diferentes universidades del país.

Universidad Distrital Francisco José de Caldas

Curso de capacitación en el diseño de sistemas embebidos utilizando Linux y las plataformas de desarrollo de la familia ECB_AT91, al grupo de investigación en Robótica móvil Autónoma “ROMA”¹, liderado por el Ingeniero Giovanni Bermúdez.

Universidad de Los Andes

Se dictó el curso de sistemas embebidos, el cual se convirtió en una electiva del Departamento de Ingeniería Electrónica, como resultado se dirigieron las tesis de pregrado: [87] y se asesoró la tesis de maestría [52].

En la actualidad el Centro de Microelectrónica de la Universidad de Los Andes a través del profesor Fredy Segura está repitiendo el proceso de diseño y construcción de una plataforma embebida² basándose en los diseños existentes de la familia de plataformas ECB_AT91 y en la metodología de diseño y fabricación presentadas en este informe.

Escuela Naval Almirante Padilla

Curso de operación e Implementación Utilizando Linux como Herramienta de Diseño para Sistemas Embebidos al Departamento de Armas y Electrónica (DARET) liderado por el Ingeniero Alejandro Mejía.

Universidad Industrial de Santander

Se dictaron los siguientes cursos: Curso de extensión Sistemas Embebidos: Teoría y Metodologías de Diseño, Curso de Actualización en Diseño de Sistemas Embebidos Utilizando Linux. Adicionalmente, se realizaron varias tesis de pregrado en conjunto y se introdujo el uso de la plataforma de desarrollo SIE como herramienta didáctica, todo con la colaboración del profesor Jorge Ramón.

Universidad Santo Tomás

Creación del grupo de estudio en Linux *linuxencaja*, de estas reuniones en la Universidad Santo Tomás nació la comunidad *linuxencaja*.

¹<http://www.youtube.com/watch?v=HKuY198f09U>

²http://157.253.195.173/mirror2/index.php/Condor_Board

A. Artículos

A.1. Aplicaciones

A.1.1. En Hardware Evolutivo

Las plataformas desarrolladas en este trabajo fueron utilizadas como una forma novedosa y económica de implementar algoritmos de hardware evolutivo, la posibilidad de tener una FPGA y un procesador en la misma placa permitió la implementación de algoritmos genéticos de forma intrínseca, es decir, verificando el nivel de ajuste de cada individuo a la solución requerida dentro del mismo dispositivo. En esta área se generaron dos artículos:

- **Intrinsic Evolvable Hardware for Combinatorial Synthesis Based on SoC+FPGA and GPU Platforms** Será publicado en la conferencia *Genetic and Evolutionary Computation Conference GECCO 2011*; clasificada 7ma entre 711 conferencias en inteligencia artificial (AI conference ranking) y es publicado por ACM (Association for Computing Machinery¹)
- **Low Cost Platform for Evolvable-Based Boolean Synthesis** Publicado en el 2nd IEEE *Latin American Symposium on Circuits and Systems LASCAS 2011*.

A.1.2. Aplicaciones en Robótica

- **CONTROL DE SISTEMAS PARALELOS INSPIRADO EN LA NATURALEZA** 3th IEEE Colombian Workshop on Circuits and Systems.
- **PLATAFORMAS ABIERTAS HARDWARE/SOFTWARE PARA APLICACIONES EN ROBOTICA**, V Congreso Internacional de Ingeniería Mecánica y III de Ingeniería Mecatrónica.

A.2. Sobre la Metodología

- **Método de Enseñanza/Aprendizaje en Sistemas Embebidos Utilizando Hardware Copyleft** A publicar en: IEEE Latin America Transactions
- **Hardware copyleft como Herramienta para la Enseñanza de Sistemas Embebidos** Congreso Argentino de Sistemas Embebidos CASE 2011, Buenos Aires Argentina ISBN 978-987-9374-69-6
- **Metodología Para la Transferencia Tecnológica en la Industria Electrónica Basada en Software Libre y Hardware Copyleft**. VI Congreso Internacional de la Red de Investigación Y Docencia en Innovación Tecnológica RIDIT “Innovación, Empresa Y Región”.

¹ ACM, es la sociedad educativa y científica informática más grande del mundo.

- **ECBOT y ECB_AT91 Plataformas Abiertas para el Diseño de Sistemas Embebidos y Co-Diseño HW/SW**, VIII Jornadas de Computación Reconfigurable y Aplicaciones, Madrid España.
- **Metodología Para la Transferencia Tecnológica en la Industria Electrónica Basada en Software Libre y Hardware Copyleft** XVII Workshop de Iberchip 2011.
- **SIE: Plataforma Hardware copyleft para la Enseñanza de Sistemas Digitales** XVII Workshop de Iberchip 2011.
- **Metodología para la Transferencia de Conocimientos en el Diseño de Sistemas Digitales**
A publicar en la reunión nacional ACOFI 2011.

Bibliografía

- [1] Charles M. Schweik. *Understanding Knowledge as a Commons: From Theory to Practice*, chapter Free/Open-Source Software as a Framework for Establishing Commons in Science. The MIT Press, 2006.
- [2] Luis Alejandro Cortés. *Verification and Scheduling Techniques for Real-Time Embedded Systems*. PhD thesis, Linköpings universitet Institute of Technology, 2005.
- [3] Edward F. Crawley. The CDIO Syllabus A Statement of Goals for Undergraduate Engineering Education. *URL:* <http://www.cdio.org>, 2001.
- [4] Charlotte Hess and Elinor Ostrom, editors. *Understanding Knowledge as a Commons: From Theory to Practice*. The MIT Press, December 2006.
- [5] N. Rosas. Diseño e Implementación de un Sistema Embebido para la Adquisición y Transmisión de Señales Biomédicas a Tráves de la Red Celular. Master's thesis, 2011.
- [6] Mark Weiser. The Computer for the 21st Century. <http://www.ubiq.com/hyertext/weiser/Sci-AmDraft3.html>.
- [7] M. Weiser. Some computer science issues in ubiquitous computing. *Commun. ACM*, 1993.
- [8] D. Servant. Combining amorphous computing and reactive agent-based systems: a paradigm for pervasive intelligence? In *First international joint conference on Autonomous agents and muktiagent systems: part 1*, 2002.
- [9] Héctor Martínez. Apropiación de conocimiento en Colombia. El caso de los contratos de importación de tecnología. *Revista Cuadernos de Economía*, 2004.
- [10] Kenneth Arrow. *Economic Welfare and the Allocation of Resources for Invention*. Princeton University Press, 1962.
- [11] European Technology Platform on Smart Sistem Integration (EPoSS). European Technology Platform on Smart Systems Integration. Strategic Research Agenda 2009. 2009.
- [12] M. Tovar and R. Rodríguez. PROSPECTIVA Y VIGILANCIA TECNOLÓGICA DE LA ELECTRÓNICA EN COLOMBIA. Master's thesis, Universidad Nacional de Colombia, 2007.
- [13] M. Duque and A. Gauthier. Formación de Ingenieros para la Innovación y el Desarrollo Tecnológico en Colombia. *Revista de la Facultad de Minas - Universidad Nacional de Colombia*, December 1999.
- [14] D Zuluaga, S Campos, M Tovar, R Rodríguez, J Sánchez, A Aguilera, L Landínez, and J Medina. Informe de Vigilancia Tecnológica: Aplicaciones de la Electrónica en el Sector Agrícola. Technical report, COLCIENCIAS, 2007.

- [15] M. Odedra. *Information Technology Transfer to Developing Countries: Case studies from Kenya, Zambia and Zimbabwe*. PhD thesis, London School of Economics, 1990.
- [16] Innovation Associates Inc. Technology Transfer and Commercialization Partnerships Executive Summary.
- [17] Creative Commons. Attribution-ShareAlike 2.0. URL: <http://creativecommons.org/licenses/by-sa/2.0/>, February 2010.
- [18] V. Gigch, editor. *Applied General System Theory*. Harper & Row, 1978.
- [19] F. Bar, F. Pisani, and M. Weber. Mobile technology appropriation in a distant mirror: baroque infiltration, creolization and cannibalism. *Seminario sobre Desarrollo Económico, Desarrollo Social y Comunicaciones Móviles en América Latina*. Buenos Aires, April 2007.
- [20] Goel Cohen. *Technology transfer: strategic management in developing countries*. Sage Publications inc, 2004.
- [21] K. Goel and Sayers B. Modelling Global-Oriented Energy Tecnology Transfer to DCs. *Sixth Global Warning International Conference, San Francisco*, 1995.
- [22] M. odedra-straub. The Myths and Illusions of Technology Transfer. *IFIP World Congress Proceedings*, August 1994.
- [23] James A. Jolly. The Technology Transfer Process: Concepts, Framework and Methodology. *The Journal of Technology Transfer*. Springer, 1977.
- [24] E. Mansfield. International technological transfer: Forms, Resource Requirements, and Policies. *American Economic Review, Paper and Proceedings*, 1975.
- [25] A. Erdilek and A. Rapoport. *Conceptual and measurement problems in international technology transfer: a critical analysis*”. Quarum Books, Westport, 1985.
- [26] M. Odedra. Information Technology Transfer to Developing Countries Is it really taking place? *The 4th IFIF.TC9 International Conference on Human Choice and Computers, North Holland, Amsterdam, Netherlands, HCC 4 held jointly with the CEC FAST Programme.*, 1991.
- [27] C. Forero and H. Jaramillo. The access of researchers from developing countries to international science and technology. *International Social Science Journal, Volume 54, Issue 171*, 2002.
- [28] K. Al-Mabrouk and J. Soar. Building a Framework for Understanding and Improving Information Technology Transfer Process in the Arab Countries. *9th IBIMA Conference: Information Management in Modern Organisations - Trends & Challenges*, 2008.
- [29] D. Wood and A. Weigel. International Collaboration on Satellite-Enabled Projects in Developing Countries. *CP1103, Space, Propulsion & Energy Sciences International Forum - SPESIF*, 2009.
- [30] K. Al-Mabrouk and J. Soar. Identification of key issues for successful technology transfer in the Arab countries: a Delphi study. *International Journal of Technology Transfer and Commercialisation*, 2009.

- [31] R. Janssen. Exploring the impact of culture Technology transfer to five African countries. Master's thesis, University of Twente, the Netherlands, 2010.
- [32] W. Knox. Systems for technological information transfer. *Science*, 1973.
- [33] Colciencias. Plan Estratégico del Programa Nacional de Desarrollo Tecnológico Industrial y Calidad 2005 - 2015. Technical report, Colciencias, 2005.
- [34] M. Smith. *Application specific integrated circuits*. Addison-Wesley, 199.
- [35] Jon Hall. POR GRANDES QUE SEAN...: ASEGURE EL FUTURO DE SU NEGOCIO. *Linux magazine*, ISSN 1576-4079(58):92, 2009.
- [36] A. Grove. How America Can Create Jobs. http://www.businessweek.com/magazine/content/10_28/b4186048358596.htm, May 2010.
- [37] Linux Foundation. Estimating the Total Development Cost of a Linux Distribution. URL: <http://www.linuxfoundation.org/sites/main/files/publications/estimatinglinux.html>, 2008.
- [38] R. M. Stallman. *The GNU Operating System and the Free Software Movement Voices from the Open Source Revolution*. O'Reilly and Associates, 1999.
- [39] S. Banguero and M. Erazo. *Plataforma de Desarrollo para Sistemas Embebidos Basada en el GameBoy Advance*. Universidad Industrial de Santander, 2007.
- [40] C. Camargo. ECB_AT91 y ECBOT Plataformas Abiertas para el desarrollo de Sistemas Embebidos. URL: <http://wiki.emqbit.com/free-ecb-at91>.
- [41] C. Camargo. First Colombian Linux SBC runs Debian. URL: <http://www.linuxfordevices.com/c/a/News/First-Colombian-Linux-SBC-runs-Debian/>, 2006.
- [42] C. Camargo. ECBOT: Arquitectura Abierta para Robots Móviles. *VII conferencia Iberoamericana en Sistemas, Cibernetica e Informatica*, 2008.
- [43] C. Camargo. ECBOT y ECB_AT91 Plataformas Abiertas para el Diseño de Sistemas Embebidos y Co-Diseño HW/SW. *VIII Jornadas de Computación Reconfigurable y Aplicaciones, Madrid España*, September 2008.
- [44] C. Camargo. ECBOT: Arquitectura Abierta para Robots Móviles. *IEEE Colombian Workshop on Circuits and Systems*, 2007.
- [45] C. Camargo. PLATAFORMAS ABIERTAS HARDWARE/SOFTWARE PARA APLICACIONES EN ROBOTICA. *V Congreso Internacional de Ingeniería Mecánica y III de Ingeniería Mecatrónica (por publicar)*, 2011.
- [46] C. Camargo. SIE: Plataforma Hardware copyleft para la Enseñanza de Sistemas Digitales. *XVII Workshop de Iberchip, Bogotá Colombia*, February 2011.
- [47] D. Ovalle and D. Mendez. *Diseño e Implementación de equipo Básico de Laboratorio Utilizando Sistemas Embebidos y FPAs*. Universidad Nacional de Colombia, 2004.

- [48] O. Orjuela. Sistema Embebido Para la Reproducción de Video en pantalla dde LEDs RGB. Master's thesis, Universidad Nacional de Colombia. Facultad de ingeniería, 2011.
- [49] C. Gavilan. Diseño Hardware-Software Para Seguimiento de un objeto Mediante Visión Artificial. Master's thesis, Universidad Nacional de Colombia. Facultad de Ingeniería, 2010.
- [50] J. Duque. Voz IP Sobre un Sistema Embebido Linux. Master's thesis, Universidad Nacional de Colombia. Facultad de Ingeniería, 2011.
- [51] D. Arango. Diseño e Implementación de una Plataforma de Desarrollo Para Aplicaciones que Requieran Co-Diseño Hardware-Software Utilizando un Procesador de 32 bits. Master's thesis, Universidad Nacional de Colombia. Facultad de Ingeniería, 2009.
- [52] D. Mendez. Desarrollo de una plataforma flexible para el prototipaje de aplicaciones WSN (Wireless Network Sensors). Master's thesis, Universidad de los Andes, 2007.
- [53] C. Camargo. Implementación de Sistemas Digitales Complejos Utilizando Sistemas Embebidos. *Memorias del XI Workshop de Iberchip ISBN 959-261-105-X*, 2005.
- [54] C. Camargo. Control de Sistemas Paralelos Inspirado en la Naturaleza. *Colombian Workshop on Circuits and Systems*, 2009.
- [55] C. Camargo. Linux como herramienta de Desarrollo de Sistemas Embebidos. *XII Workshop de Iberchip, San Jose*, 2006.
- [56] C. Camargo and O. Sanchez. Linux embebido como herramienta para realizar reconfiguración parcial. *XII Workshop Iberchip*, 2006.
- [57] I. Castillo, C. Camargo, and C. Perez. Automatización de un puente grúa a escala, mediante una plataforma embebida la cual soporta multiprogramación. *XII Workshop Iberchip*, 2006.
- [58] J. Espinosa, F. Segura, and C. Camargo. Evolución de un Arreglo de Células Utilizando Algoritmos Genéticos. *Memorias del XI Workshop de Iberchip ISBN 959-261-105-X*, 2005.
- [59] C. Camargo. ECBOT y ECB_AT91 Plataformas Abiertas para el Diseño de Sistemas Embebidos y Co-diseño HW-SW. *VIII Jornadas de Computación Reconfigurable y Aplicaciones*, 2008.
- [60] C. Camargo. Hardware copyleft como Herramienta para la Enseñanza de Sistemas Embebidos. *Simposio Argentino de Sistemas Embebidos*, 2011.
- [61] C. Camargo. Hardware copyleft como Herramienta para la Enseñanza de Sistemas Embebidos. *Congreso Argentino de Sistemas Embebidos CASE 2011, Buenos Aires Argentina*, March 2011.
- [62] C. Camargo. Metodología Para la Transferencia Tecnológica en la Industria Electrónica Basada en Software Libre y Hardware Copyleft. *XVII Workshop de Iberchip, Bogotá Colombia*, February 2011.
- [63] C. Camargo. Low Cost Platform for Evolvable-Based Boolean Synthesis. *IEEE Latin American Symposium on Circuits and Systems (LASCAS 2011), Bogotá Colombia*, February 2011.
- [64] Worldwide CDIO Initiative. "Benefits of CDIO" URL:<http://www.cdio.org/benefits-cdio> on November, 2009.

- [65] Bloom B S, editor. *Taxonomy of Educational Objectives, the classification of educational goals*. Mckay.
- [66] Elinor Ostrom. *Governing the Commons: The Evolution of Institutions for Collective Action (Political Economy of Institutions and Decisions) (Paperback)*. Cambridge University Press, November 1990.
- [67] Elinor Ostrom. Reformulating the Commons. *Swiss Political Science Review, Volume 6, Number 1*, 2000.
- [68] Sandler Todd. Collective Action: Theory and Applications. *Ann Arbor: University of Michigan Press*, 1992.
- [69] Ruth Meinzen-Dick, Monica Di Gregorio, and Nancy McCarthy. Methods for Studying Collective Action in Rural Development. *CAPRI Working Paper, no. 33. International Food Policy Research Institute, 2033 K Street, N.W., Washington, DC 20006*. <http://www.capri.cgiar.org/pdf/capriwp33.pdf>, 2004.
- [70] E. OSTROM. The Rudiments of a theory of the Origins, Survival, and Performance of Common-Property Institutions. *BROMLEY, D.W. et al. (eds.) Making the Commons Work: Theory, Practice, and Policy. San Francisco, CA: ICS Press*, 1992.
- [71] J .M. BALAND and J.P. PLATTEAU. Halting Degradation of Natural Resources. Is There a Role for Rural Communities? *Oxford: Clarendon Press*, 1996.
- [72] Motorola. MOTODEV Project. URL: <http://developer.motorola.com/platforms/mobile-linux/>.
- [73] Google. Android Project. URL: <http://www.android.com/>.
- [74] Nokia. Maemo Project. URL: <http://www.maemo.org/>.
- [75] Free Software Foundation. GNU General Public License. URL: <http://www.gnu.org/copyleft/gpl.html>.
- [76] A. Cortesi. Elinor Ostrom, the commons problem and Open Source. URL: <http://corte.si/posts/opensource/ostrom/index.html>, 2009.
- [77] Wikipedia. Richard Stallman. URL: http://es.wikipedia.org/wiki/Richard_Stallman.
- [78] C. M Schweik and A. Semenov. The Institutional Design of 'Open Source' Programming: Implications for Addressing Complex Public Policy and Management Problems. URL: http://www.firstmonday.org/issues/issue8_1/schweik/, 2003.
- [79] Van Wendel, R. Joode, J. A. de Bruijn, and M. J. G. van Eeten. *Protecting the Virtual Commons: Self-Organizing Open Source and Free Software Communities and Innovative Intellectual Property Regimes*. Asser Press, 2003.
- [80] R. A. Ghosh, G. Robles, and R. Glott. Free/Libre and Open Source Software: Survey and Study. Technical report University of Maastricht, The Netherlands: International Institute of Infonomics. URL: <http://www.infonomics.nl/FLOSS/report/index.htm>, 2002.

- [81] S. Shah. Motivation, Governance, and the Viality of Hybrid Forms in Open Source Software Development. *Management Science*, July 2006.
- [82] E. Rodriguez, A. Calderón, D. Mendez, and C. Camargo. Sistema Integrado de Medición de Energía, Calidad del Servicio y Operación Remota. *MEMORIAS : VI Jornada de Distribución de Energia Eléctrica - ASOCODIS*, 2009.
- [83] R. Vallarta. OpenBlocks : an extendable framework for graphical block programming systems. *MIT*, 2007.
- [84] M. Resnick. Starlogo: An Environment for Decentralized Modeling and Decentralized Thinking. *Conference Companion on Human Factors in Computing Systems. New York, NY, USA: ACM Press.*, 1996.
- [85] D. Aranguren, H. Torres, E. Olarte, I. Santoyo, F. Amórtegui, and E. Perez. Sistema de Medición de Tormentas Eléctricas. In *Simposio Internacional sobre Calidad de la Energía Eléctrica SICEL-2007*, 2007.
- [86] D. Aranguren, H. Torres, E. Olarte, and I. Santoyo. Molino de campo eléctrico. Registro comercial, Colombia, 2004.
- [87] C. Perez. Automatización de un puente grúa a escala mediante una plataforma embebida la cual soporta multiprogramación. Tesis de pregrado Departamento de Ingeniería Electrónica Universidad de los Andes., 2005.