Universidad de la República
Facultad de Ingeniería

Proyecto de fin de carrera

Recarga Fácil por Radio Frecuencia

RF^2

Daniel Aicardi, Melina Rabinovich, Edgardo Vaz

Tutores: Ing. Juan Pablo Oliver, Ing. Andrés Aguirre

Montevideo, Uruguay
Julio 2011

Recarga Fácil por Radio Frecuencia

Resumen

El presente documento describe el prototipo Recarga Fácil por Radio Frecuencia, RF^2. El mismo consiste en un sistema para recarga y consulta de tarjetas RFID como las que se utilizan hoy día en el sistema de transporte metropolitano (STM).

El prototipo es rápido y posee una interfaz de comunicación con el usuario para que el mismo comprenda sin problemas que es lo que está sucedediendo.

El hardware, consiste en una single board computer (BeagleBoard), un lector de tarjetas RFID, un lector de tarjetas de contacto (SAM), y una interfaz de usuario. Salvo la single board computer, el resto del hardware fue enteramente diseñado por el grupo de trabajo. El lector escritor de tarjetas RFID es el único pcb en el prototipo final que fue fabricado por el grupo, el resto fueron fabricados únicamente en etapas de prueba.

Agradecimientos

En primer lugar queremos agradecer a nuestras familias y amigos. Agradecemos a Leonardo Steinfeld, Nicolás Barabino, Francisco Lanzari, María Eugenia Corti, Christian Gutierrez, Andrés Bergeret, Gonzalo Tabares, Klaus Rotzinger, Marcelo Fiori, Pablo Cancela, Ana y Claudia Rabino. Y a todos los que de alguna u otra forma colaboraron con nosotros.

para cuchu,
chuchuchu,
y pirimpirim jajaja

Prefacio

"El ciudadano Línea saca su billetera, extrae su tarjeta y la introduce en la máquina registradora; una serie de gestos automáticos. Unas mandíbulas de aluminio se cierran sobre ella, unos dientes de cobre buscan la clave magnética, y una lengua electrónica saborea la vida del ciudadano Línea. Lugar y fecha de nacimiento. Padres. Raza. Religión. Historial educativo, militar y de servicios civiles. Estado. Hijos. Ocupaciones, desde el comienzo hasta el presente. Asociaciones. Medidas físicas, huellas digitales, retínales, grupo sanguíneo. Grupo psíquico básico. Porcentaje de lealtad, índice de lealtad en función del tiempo hasta el momento del último análisis... ... El ciudadano Línea se encuentra en la ciudad donde, la noche anterior, dijo que estaría, así que no ha tenido que hacer una corrección. Los nuevos informes se añaden al historial del ciudadano Línea. Toda su vida regresa al banco de datos. Desaparece de la unidad exploradora y la unidad comparativa, para que éstas atiendan la próxima llegada. La máquina ha tragado y digerido otro día. Está satisfecha."

Sam Hall (1953), Poul Anderson

La narración anterior es parte de un cuento de ciencia ficción llamado "Sam Hall", escrita por Poul Anderson en 1953. En esta historia el autor describe un mundo donde cada persona tiene asignada una tarjeta conteniendo datos que la caracterizan, y puede ser controlado su accionar a través de una super computadora que almacena y procesa los datos de toda la humanidad. En nuestros días este cuento de ciencia ficción no está tan alejado de la realidad, las tarjetas "inteligentes" (smart cards) son cada vez más usadas en múltiples aplicaciones como ser, pasaporte electrónico, pago electrónico, sistemas de transporte, controles de acceso y sistemas de seguridad, entre otros. El siguiente proyecto se desarrolla con la intención de aprender las bases del mundo de las tarjetas "inteligentes" y que sirva como punto de partida para que otros entiendan su funcionamiento. No es intención de los autores que se use el contenido de este documento con fines como los que se indicaban en la narrativa de ciencia ficción, muy por el contrario, el empleo de esta tecnología debe estar en favor de las personas y no en su contra.

Índice general

Índice de figuras

Índice de cuadros

Parte I

Introducción

Capítulo 1

Descripción del proyecto

1.1. Definición

A partir de la puesta en marcha del STM, surge la necesidad de consultar y recargar tarjetas RFID (utilizadas en dicho sistema) en línea (con conexión a un servidor de la Intendencia de Montevideo), de forma rápida, segura y auto-gestionada por parte del usuario, en diversos puntos de Montevideo.

1.2. Antecedentes

- AFE: Prototipo de sistema embebido capaz de cargar y consultar tarjetas RFID como las utilizadas en el STM. El mismo se compone de varios módulos: una SBC (single board computer), un lector-escritor de tarjetas RFID, un lector de tarjetas de contacto, un módem 3G/GPRS y una interfaz con el usuario que consta de un display,leds y buzzer.

- OpenPCD: Diseño de hardware libre para dispositivos de proximidad de acoplamiento (PCD) basado en comunicación RF de 13,56MHz. Este dispositivo es capaz de desplegar información desde Tarjetas de proximidad de Circuito Integrado (PICC) que se ajusten a las normas de proveedores independientes, tales como ISO 14443, ISO 15693, así como los protocolos propietarios como Mifare Classic.

Capítulo 2

Objetivo general del proyecto

2.1. ¿Qué y para qué?

En principio, el objetivo del proyecto era realizar un prototipo de sistema embebido mediante el cual se pudiera interactuar con tarjetas RFID (basadas en las normas ISO 14443) como las usadas en el STM. Mejorar la arquitectura actual del dispositivo AFE rompiendo dependencias tecnológicas con el actual lector-escritor de tarjetas Mifare.

Para lograrlo se partiría de la base de un dispositivo open-hardware y open-firmware (OpenPCD). Generar conocimiento dentro de la División Tecnología de la Información de la Intendencia de Montevideo.

Luego se dejó de lado el partir del dispositivo OpenPCD, para lograr un diseño propio de lector-escritor de tarjetas RFID.

2.2. ¿Por qué cambiar la arquitectura actual?

Como arquitectura precedente existe la del prototipo AFE (Artefacto Feo de Exhibir), realizada por el grupo de electrónica de la IM. La misma consiste en una SBC, que se fabrica con otro propósito y es utilizada en esta aplicación puesto que es la única forma de adquirir este tipo de hardware en plaza. A ésta se conectan a través de puertos USB, un lector-escritor de tarjetas Mifare, un lector de tarjetas de contacto, un modem 3G y un dispositivo diseñado a partir de un microcontrolador PIC, llamado USB4ALL, el cual es open-hardware y open-firmware, en el que se pueden conectar otro tipo de

dispositivos cuya interfaz nativa no sea USB, como ser display, buzzer, leds, sensores, etc, los cuales no pueden ser conectados directamente a la SBC porque la misma no cuenta con los puertos de expansión necesarios.

Surge entonces la necesidad de cambiar la configuración de dicha arquitectura. Se hace necesario romper dependencias tecnológicas con el lector-escritor de tarjetas Mifare, ya que dejó de ser soportado por la librería pcsclite (a pedido del fabricante); y con la SBC, que es empleada en una aplicación específica y puede dejar de fabricarse o sufrir cambios drásticos que ya no permitan su uso.

2.3. Alcance

- Hardware: Se fabricará un módulo donde se insertará la tarjeta de contacto (SAM). Se agregará un display LCD 16x2, leds y buzzer como interfaz para el usuario. Se fabricará un módulo de lectura-escritura RFID. Se estudiará la forma de conectar los periféricos a la placa de la SBC.

- Software: Se hará lo necesario para que el lector-escritor RFID funcione como un dispositivo soportado por la librería pcsclite empleada en sistemas operativos Linux para interactuar con lectores de tarjetas. Con esto se logrará compatibilidad hacia atrás, de modo de poder reutilizar las aplicaciones desarrolladas para el dispositivo AFE.

2.4. Especificación funcional

El prototipo final deberá ser capaz de interactuar con tarjetas RFID a través de la antena del dispositivo lector-escritor RFID, y con una tarjeta de contacto. Luego de los controles correspondientes y autenticación de la tarjeta (con datos encriptados), comenzará la interacción con el usuario mediante un display LCD16x2 que será la interfaz de comunicación con el mismo. El display informará al usuario de las tareas que se estén realizando, mensajes cortos y descriptivos. Los tiempos de recarga y consulta deberán ser menores a un minuto.

2.5. Criterios de éxito

- Lograr recargar y consultar tarjetas RFID mediante el dispositivo embebido, el cual estará en comunicación directa con la infraestructura STM.
- Los tiempos de recarga y consulta deberán ser menores a un minuto.

Parte II

Diseño

Capítulo 3

Funcionamiento del prototipo

3.1. Requerimientos

Los requerimientos que deben cumplirse son los mismos que fueron definidos para el dispositivo AFE. Es necesario poder leer y escribir tarjetas RFID, comunicarse con servidores STM, leer tarjetas de contacto (módulo de seguridad), informar al usuario de lo que ocurre a través de una interfaz simple.

3.2. Funcionamiento general del prototipo

Normalmente el prototipo está en espera de la llegada de un usuario, desplegando los mensajes “Sistema de recarga STM”, “Aproxime su tarjeta”, en forma alternada con un pequeño intervalo de tiempo de espera entre cada mensaje. Cuando llega un usuario, éste acerca su tarjeta RFID y el dispositivo desplegará el mensaje “No retire su tarjeta”. Si el usuario retirara la tarjeta, de modo de que el dispositivo no pueda leerla, el mismo desplegará el mensaje “Mantenga cerca su tarjeta”. Si no es posible leer la tarjeta el prototipo lo indicará desplegando el mensaje ”Lectura inválida. Vuelva a intentarlo.”. Si el servidor STM indica que hubo algún error con la tarjeta, el prototipo desplegará en el display los mensajes “Operación incompleta. Vuelva a intentarlo.”, “Disculpe las molestias.”. Si la tarjeta cuenta con saldo para cargar, lo indicará con un mensaje del tipo “Transfiriendo 50 pesos” o “Transfiriendo 10 viajes” en caso de ser una tarjeta de estudiante. Si al intentar escribir la tarjeta, el usuario la retira, se le indicará “No mueva su

tarjeta.".

Si no se pudiera escribir en la tarjeta, "No se pudo grabar su tarjeta. Vuelva a intentarlo.". Al finalizar la transacción, indica al usuario el nuevo saldo con un mensaje del tipo "Su saldo es de 50 pesos" o "Su saldo es de 10 viajes". Si algún imprevisto ocurriera, el prototipo desplegará los mensajes "Operación incompleta. Vuelva a intentarlo.", "Disculpe las molestias.". Si el dispositivo pierde conexión con el servidor STM intercala los mensajes "Temporalmente sin servicio", "Disculpe las molestias.".

Capítulo 4

Hardware

4.1. Arquitecturas estudiadas

Se definieron varias alternativas como posible solución, y se fueron descartando a medida que se encontraron limitantes o que no se cumplieran los requerimientos exigidos.

A continuación se describen algunas de las posibles arquitecturas:

1 - SBC + OpenPCD + microcontrolador + lector de tarjetas de contacto + display + buzzer + leds Tanto el OpenPCD como el microcontrolador se conectan directamente por USB a la SBC. El microcontrolador maneja el resto de los dispositivos (lector de tarjetas de contacto, display, buzzer y leds).

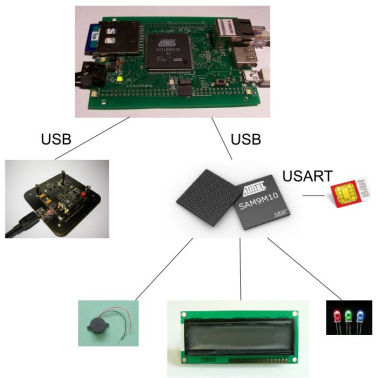


Figura 4.1: Solución posible 1

Esta arquitectura tiene como ventaja el uso de la SBC que permite instalar un sistema operativo, reutilizar código ya implementado, posee varios puertos de E/S (I2C, USART, SPI, USB, GPIO, etc.), tiene gran capacidad de procesamiento, maneja memoria externa y brinda facilidad para realizar prototipos. Otra ventaja es el uso del microcontrolador que actúa como co-procesador, manejando el resto de los periféricos.

2 - SBC + OpenPCD + lector de tarjetas de contacto + display + buzzer + leds El OpenPCD se conecta por USB a la SBC. La SBC maneja el resto de los dispositivos (lector de tarjetas de contacto, display, buzzer y leds) a través de sus interfaces nativas.

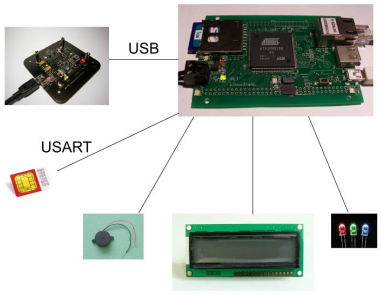


Figura 4.2: Solución posible 2

3 - SBC + lectores de tarjetas + display + buzzer + leds Todos los periféricos (lectores de tarjetas, display, buzzer y leds) se conectan a la SBC a través de sus interfaces nativas, también el integrado CL RC632 de Philips que maneja la comunicación con las tarjetas sin contacto. Se diseña la antena para propagar RF.

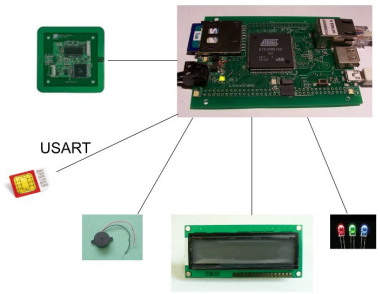


Figura 4.3: Solución posible 3

4 - microcontrolador + lectores de tarjetas + display + buzzer + leds Consta de un único PCB, que posee un microcontrolador como sistema central al cual se conectan el resto de los dispositivos. Dicho PCB tiene incorporada la antena para la propagación de RF. Se prevee el agregado de un modem 3G.

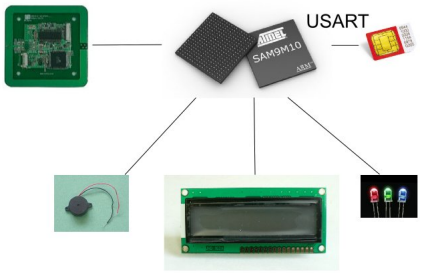


Figura 4.4: Solución posible 4

4.2. Arquitectura seleccionada

Luego de estudiar ventajas y desventajas de las arquitecturas planteadas, y discutirlo con los tutores, se eligieron dos de las posibilidades:

- SBC + OpenPCD + lector de tarjetas de contacto + display + buzzer + leds
- SBC + lectores de tarjetas + display + buzzer + leds

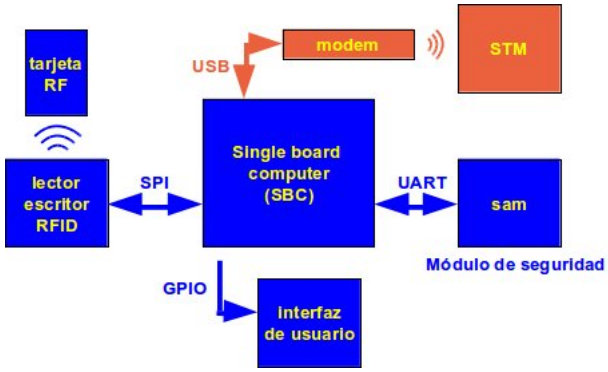


Figura 4.5: Diagrama de bloques de la arquitectura seleccionada

En principio se pensó en diseñar la primera opción como paso intermedio para testear el hardware diseñado, pensando en luego migrar a la segunda. De modo que, se migraría de la primer configuración a la segunda al diseñar e implementar un lector-escritor RFID que sustituya al OpenPCD. Luego simplemente se diseñó la segunda, aunque se hicieron pruebas con el OpenPCD nunca se llegó a implementar por completo la primer configuración.

En una primera instancia se pretendía usar únicamente el dispositivo OpenPCD, ya que el mismo cuenta con un microcontrolador de la familia ARM, el AT91SAM7S128, una vez estudiado se llegó a la conclusión de que no permitía instalarle un kernel de Linux. Otra desventaja encontrada fue que sólo tiene un puerto I2C como forma de conectar periféricos.

Surge entonces la necesidad de usar una SBC como dispositivo capaz de correr un sistema operativo y ejecutar las aplicaciones necesarias para que el dispositivo cumpla con los requerimientos exigidos. El dispositivo OpenPCD pasaría entonces a cumplir

la función de lector-escritor de tarjetas RFID, conectado a la SBC a través de su puerto USB, mientras que para el resto de los periféricos se diseñaría un pcb que fuera capaz de ser conectado a la SBC a través de sus interfaces nativas. Esta arquitectura fue descartada por el incremento en el costo del proyecto.

Fue necesario entonces descartar el uso del dispositivo OpenPCD y dar lugar a un diseño propio del lector-escritor de tarjetas RFID, usando para esto el integrado RC632 de Philips.

La última opción y la más ambiciosa, plantea el diseño completo de un pcb conteniendo un microcontrolador y memoria capaz de correr un sistema operativo, los lectores de tarjetas, tanto de contacto como RFID, un modem 3G y el resto de los periféricos (display, leds, buzzer). Esta opción fue dejada de lado por entender que excedería los plazos de tiempo del proyecto.

4.3. Elección de hardware, módulos

4.3.1. SBC

En primera instancia se confeccionó una lista con posibles candidatas de SBC disponibles en el mercado internacional, teniendo en cuenta factores como: precio, puertos de E/S, memoria RAM, memoria Flash, puertos USB, Linux embebido, entre otros. Se definieron una serie de requisitos mínimos necesarios para seleccionar de la lista la SBC que más se adecuara a la arquitectura definida. Para la comunicación con el resto de los módulos será necesario: una interfaz UART para el módulo de seguridad (SAM); una interfaz SPI para el módulo lector-escritor RFID (RC632 de Philips); 20 GPIO para display, leds, buzzer, otros; 2 USB host, uno para la conexión de un módem 3G (intercambio de datos con un servidor central – STM) y otro para aplicaciones futuras. En cuanto a la memoria disponible debe ser de 32Mb de RAM y 8Mb de flash para el uso de un sistema operativo embebido. Es conveniente, pensando a futuro, que el procesador trabaje a una frecuencia no menor a 200MHz. Dado el presupuesto estimado para el proyecto, el precio no debe superar los 150 dólares en origen. Como requisito adicional se exigió que existiera un foro actualizado y soporte técnico que permita evacuar dudas.

Aplicados los requisitos mínimos a la lista previamente confeccionada de SBC can-

didatas, optamos por dos: GESBC-9G20 y HAWKBOARD. En cuanto a la primera opción, GESBC-9G20, los fabricantes no respondieron consultas, por tanto se descartó. Se optó entonces por la segunda opción, HAWKBOARD, puesto que respondieron a las consultas en tiempos razonables y se logró evacuar dudas desde el foro.

Finalmente, la SBC seleccionada para trabajar fue la BeagleBoard. Luego de comprar dos HAWKBOARD, ambas resultaron defectuosas, para intentar cumplir los plazos se optó por usar lo que se tenía a la mano (INCO) y justo cumplía los requisitos mínimos aunque se tuvo que diseñar otro módulo por problemas de voltajes.

4.3.2. VLT - Conversor de Voltajes

Este módulo no fue tenido en cuenta en la primera etapa del diseño de la arquitectura hardware, sino que surge como necesidad debido al cambio de SBC que nos vimos obligados a tomar. Como consecuencia de lo anterior vimos la ventaja de incorporar una placa que permite la conexión entre la SBC y el resto del hardware, el cual puede permanecer inalterado por más que no ocurra lo mismo con la SBC, ya que ésta puede cambiar de versión o dejar de fabricarse en un breve lapso de tiempo. El único elemento a cambiar sería entonces la placa VLT, que es más simple y barata de fabricar que las restantes partes. La placa de circuito impreso VLT consta básicamente de dos conectores, uno de ellos permite la conexión con la Beagleboard y el otro la conexión con el restante hardware el cual se encuentra intergrado en un PCB llamdo SCUI. Ambos conectores no se encuentran directamente interconectados entre sí a través de pistas, pues para el caso particular de Beagleboard fue necesario incorporar conversores de tensión que permitieran el traslado del nivel de tensión desde 1,8 Volts que usa esta SBC, a las tensiones con las que operan los periféricos, ya sea 3,3 o 5 Volts. El último elemento, no menos importante, es un regulador de tensión LDO que permite generar 3,3 Volts a partir de la fuente de tensión de 5 Volts de la propia Beagleboard.

4.3.3. SCUI - Lector de tarjetas de contacto e Interfaz de Usuario

La SAM es una tarjeta de contacto en la cual corre un sistema operativo y una máquina virtual de java, son también conocidas como java card y se utilizan en aplicaciones donde es necesario generar transacciones seguras. Este módulo permite en-

criptar datos y generar sesiones seguras con los servidores usados en la infraestructura del STM. Se precisa entonces un lector de tarjetas de contacto (SAM).

La interfaz de usuario se pensó como algo muy básico, que no genere confusiones al usuario durante una transacción y le indique que operaciones se están realizando sobre su tarjeta, como ser: consulta de saldo, incremento de saldo, etc. Para cumplir con lo anterior se optó por emplear un display LCD16x2, tres leds y un buzzer.

El módulo SCUI puede dividirse en dos partes, una de ellas es un lector de tarjetas de contacto basadas en la norma ISO7816, y la otra es una simple interfaz para el usuario. El lector de tarjetas de contacto(smart cards), está compuesto por un conversor full duplex a half duplex el cual se encuentra conectado a uno de los puertos UART de la SBC a través del módulo VLT, que se describió en el punto anterior. Este conversor permite la transmisión de datos directamente entre la tarjeta y la SBC, sin necesidad de intercalar un ASIC para el manejo de tarjetas del tipo ISO7816. Cuenta también con un oscilador para alimentar la entrada de reloj de las tarjetas. La entrada de control(OE) del oscilador operada desde la SBC permite poner la salida de reloj en tercer estado, cosa muy útil a la hora de cumplir con la secuencia de inicialización de las tarjetas descritas en el estandar. El lector permite operar con tarjetas clase A (alimentadas a 5 Volts) y clase B(alimentadas a 3,3 Volts) haciendo uso de un jumper que permite intercambiar la tensión de alimentación suministrada a la tarjeta. Se cuenta con un zócalo para insertar la tarjeta de contacto. Por otra parte, la intefaz de usuario está compuesta por tres leds(verde, amarillo y rojo), buzzer y un display LCD16x2 donde son desplegados los mensajes que indican al usuario la operación que se efectúa sobre su tarjeta mifare. El último elemento a describir aquí es un conector receptáculo 5x2(100mils) en el que se conecta el módulo lector/escritor RFID que opera con las tarjetas RFID mifare.

4.3.4. Lector-Escritor RFID

Si bien se descartó el uso del dispositivo OpenPCD, se toma como ejemplo para el diseño del lector-escritor de tarjetas RF ya que el mismo se apega a las reglas de diseño que establece el fabricante del integrado RC632. Para detalles a tener en cuenta en la realización de una antena RF, ver Anexo I.

Este módulo es el encargado de la comunicación con las tarjetas RFID que cumplen con la norma ISO14443[iso]. Consta básicamente de 4 secciones entre las que se encuentran: el integrado CL RC632[pdf]; el filtro EMC, el circuito de adaptación de impedancia (matching); y el inductor de la antena. El ASIC CL RC632 permite, por un lado la comunicación digital con un microprocesador a través de su puerto de datos y por el otro lado la transmisión de datos hacia la antena que emitirá la señal RF para la comunicación con las tarjetas ISO14443. Lo que llamamos propiamente antena RF está conformada por el filtro EMC, la adaptación de impedancia (matching) y por el inductor, que propaga el campo magnético para lograr el acoplamiento necesario entre lector y tarjeta, de aquí la sigla PCD (Proximity Coupling Divice).

Los principios básicos de funcionamiento de la antena se detallan en el anexo [*].

4.4. Funcionamiento de módulos

4.4.1. SBC

La SBC contiene un microcontrolador y memoria que son capaces de correr un sistema operativo, en este caso se usará la distribución Angstrom de Linux, la cual está orientada a desarrollar sistemas embebidos. Sobre el sistema operativo se instalan los módulos y librerías necesarias para hacer uso del hardware que contiene la SBC. En nuestra aplicación usaremos uno de sus puertos SPI para la comunicación con el lector-escritor de tarjetas RF, un puerto UART para la comunicación de datos con el lector de tarjetas de contacto y varias salidas GPIO para el control de la interfaz de usuario. Por último se usará uno de los puertos USB para agregar un modem 3G que permita la conexión inalámbrica con un servidor para el intercambio de datos.

4.4.2. VLT - Conversor de Voltajes

El corazón de esta placa son los integrados TXB0108[pdf] que permiten la interconexión de dispositivos que operan en distintos niveles de tensión. Básicamente el integrado está constituído por dos puertos, puerto A y puerto B cada uno de 8 bits. El

puerto A opera con la tensión de 1,8 Volts que permite ser conectado a la Beagleboard, el puerto B opera con la tensión de 3,3 Volts en el caso que se encuentra conectado al IC RC632, y de 5 Volts para los restantes periféricos. Cada I/O de un puerto es sensible a los flancos de subida o bajada, trasladando estos cambios a la I/O correspondiente del puerto opuesto. Este integrado posee también un entrada de control para poner los puertos en estado de alta impedancia. Una ventaja es que no poseen entrada de control de dirección de flujo de datos, ahorrandonos pines de control que no tenemos disponibles en la Beagleboard. En la figura debajo podemos observar como está constituído cada una de las entradas/salidas del integrado. Otra pieza que compone esta placa es el regulador de tensión LDO implementado a partir del integrado LM1117[pdf], éste se usa para convertir la entrada de tensión de 5 Volts en una salida de tensión de 3,3 Volts y así poder alimentar el periférico correspondiente.

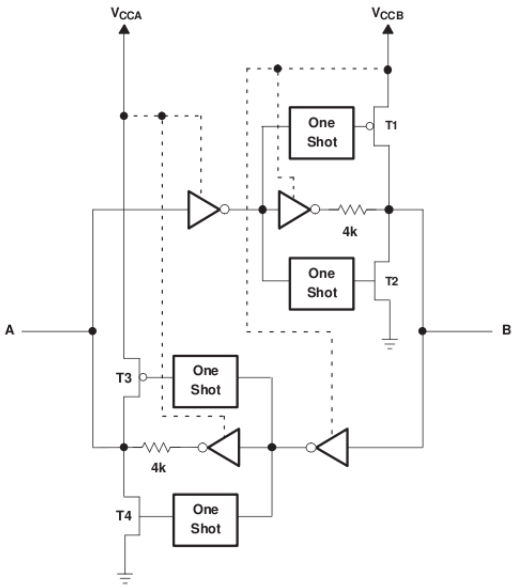


Figura 4.6: Arquitectura de una celda I/O del TXB0108

4.4.3. SCUI - Lector de tarjetas de contacto e Interfaz de Usuario

Las tarjetas de contacto que son empleadas para usos en telefonía celular (SIM) así como también las empleadas en aplicaciones de seguridad (SAM) están formadas por un microcontrolador y la comunicación con el mundo exterior la hacen a través de

un puerto UART half duplex. En nuestro diseño está contemplada la conversión de full duplex a half duplex para poder interconectar la tarjeta de contacto a la SBC cuyo puerto UART es full duplex. Este tipo de tarjetas se operan mediante el uso de comandos APDU, donde la misma responde a cada comando que es enviado desde el lado del terminal. La tarjeta sólo envía una respuesta de manera espontánea, llamada ATR (Answer To Reset), para indicar que ya está inicializada y lista para operar. En esta respuesta envía parámetros al terminal como ser la tasa en bps a la que puede transferir los datos, etc. La tarjeta nunca inicia la comunicación, actúa siempre como esclavo del terminal. La tarjeta que emplearemos en esta aplicación es del tipo SAM, los comandos APDU para operar con la tarjeta son específicos para la aplicación del STM. Mediante estos comandos es posible abrir sesiones seguras con los servidores del STM, generar las claves de lectura y escritura que luego son usadas para la comunicación con las tarjetas RF, encriptar datos a ser enviados a los servidores, etc. Las tarjetas de contacto (SAM) son suministradas por la Intendencia de Montevideo y sólo sirven para operar con tarjetas RF de testing.

El display LCD16x2 está basado en un controlador de Hitachi, el HD44780, que permite enviar comandos de control, o instrucciones de lectura-escritura a través de su puerto de entrada paralelo. Una vez que es enviada la secuencia de inicialización del display el mismo queda listo para operar, luego es posible enviar comandos para ubicar el cursor en el lugar deseado o enviar los caracteres ASCII que formarán los mensajes a ser desplegados para el usuario. Los leds, al igual que el buzzer, tienen la función de complementar los mensajes mostrados por el display. El led verde indica que la transacción se efectuó correctamente, a la vez que se emite un pulso sonoro por el buzzer. El led rojo indica que ocurrió un error, a la vez que el buzzer emite varios pulsos sonoros. El led amarillo permanece encendido mientras no haya conexión con el servidor y por tanto no se podrá operara con el dispositivo.

Lector de tarjetas de contacto ISO7816

Es un lector muy simple de implentar, su construcción se basa en un conversor full a half duplex construido a partir de un circuito transistorizado trabajando en zona de corte y saturación. Los transistores empleados son el NPN 2N3904[pdf] y el PNP

2N3906[pdf] los cuales fueron seleccionados en base a su rápida característica de conmutación que es del orden de alugnas decenas de nanosegundos. Dada la característica del circuito, es posible recibir el eco de la transmisión de datos generados por la SBC. Un elemento fundamental que compone el circuito del lector es el oscilador de frecuencia 3,579545 Mhz, este valor no es antojadizo sino que permite generar la base de tiempo adecuada para la transmisión de datos entre la tarjeta y la SBC. Otras frecuencias de reloj fueron empleadas, como ser 4 Mhz y 5 Mhz, con resultados inciertos en la recepción de los datos, aún cuando sería posible usar estos valores según la referencia [Smart Cards Handbook] para los parámetros obtenidos desde el ATR de la tarjeta. El circuito cuenta también con protección de descarga ESDA6V1W5[pdf] para los contactos de la smart card.

Interfaz de usuario

El elemento a destacar es un display LCD16x2 que basa su funcionamiento en el controlador Hitachi HD44780[pdf]. La transferencia de datos hacia el display se hace a través de un puerto con 4/8 bits de datos y 3 bits de control. Debido a no contar con la cantidad de pines disponibles en la Beagleboard para operar en el modo de 8 bits, se empleó en su lugar el modo 4 bits del display. El bit de control RS indica si el byte a enviar por el puerto de datos es una palabra de control o un caracter ASCII a ser almacenado en la memoria interna. El bit R/W por su parte indica si se efectuará una lectura o una escritura de la memoria interna del display. Por último en el bit E se indica mediante flanco de bajada que se ejecute la operación indicada con los anteriores dos bits de control, previo a este flanco la señal en el puerto de datos deben permanecer fijas. El display cuenta también con una entrada para calibrar el contraste del LCD, la calibración se realiza a partir de un divisor resistivo implementado con resistencias y un preset. El backlight del display es accionado desde uno de los pines de la SBC a partir de un circuito transistorizado que opera en zona de corte/saturación. Los restantes elementos que componene la interfaz de usuario son leds y buzzer que son accionados directamente desde los pines del puerto de expansión de la SBC.

4.4.4. Lector-Escritor RFID

En el corazón del lector/escritor de tarjetas RFID, se encuentra el chip CL RC632 que forma parte de una familia de integrados empleados para la comunicación con tarjetas sin contacto, pertenecientes a la norma ISO14443 las cuales operan a la frecuencia 13,56 Mhz. El CL RC632 soporta todas las capas del esquema de comunicación que se establecen en la mencionada norma, incluyendo el algoritmo de seguridad (CRYPTO1) para autenticar las tarjetas Mifare Classic. En lo que sigue describiremos algunas de las características principales del integrado.

Interfaz

Los comandos, bits de configuración y las banderas se acceden a través de la interfaz con un μ-procesador. El puerto elegido para la comunicación desde la SBC es el SPI, aunque es posible la comunicación a través de su puerto paralelo.

Registros

La configuración del chip se lleva a cabo a partir de un mapa de registros de control que se encuentra dividido en 8 páginas con 8 registros cada una. La manera de alcanzar estos registros es mediante el intercambio de página, mecanismo que puede ser deshabilitado mediante escritura de un "1" en el bit 7 del registro 0 en la página 0, logrando direccionamiento plano. La función de cada uno de sus registros puede ser observada en la hoja de datos del integrado[RC632.pdf].

Memoria EEPROM

La memoria está dividida en 32 bloques con 16 bytes cada bloque. El contenido de memoria EEPROM en los bloques 1 y 2 (dirección 10hex a 2Fhex) se utilizan para configurar los registros del CL RC632 durante la fase de inicialización, de forma automática. La configuración por defecto soporta la comunicación MIFARE ISO 14443 A, aunque los usuarios pueden específicar la inicialización para I-Code1, ISO 15693 o ISO 14443 B, mediante los bloques de memoria 3 al 7. Se reservan 384 bytes para almacenar las claves CRYPTO1 que son usadas para la autenticación con las tarjetas. El formato de una de estas claves puede verse en [RC632.pdf] y tiene una longitud de 12 bytes, por tanto es posible almacenar en memoria las 32 claves que posee una tarjeta.

Buffer FIFO

El integrado contiene un buffer FIFO de 64 bytes para flujo de datos con un μ-procesador. La entrada y salida del buffer de datos está conectado con el registro FI-

FOData. Escribir en este registro almacena un byte en el buffer e incrementa el puntero de escritura del buffer. La lectura de este registro muestra el contenido del buffer e incrementa el puntero de lectura. La distancia entre el puntero de escritura y lectura se puede obtener mediante la lectura del registro FIFOLength, indicando así la cantidad de bytes que se llevan almacenados. Es posible observar y controlar el estado del buffer mediante varios registros, para evitar que se produzcan errores de comuncicación con el μ-Processor.

Interrupciones

El RC632 indica ciertos eventos estableciendo el bit IRQ en el registro PrimaryStatus, y además, por la activación del pin IRQ. La señal en el pin IRQ se puede utilizar para interrumpir un μ-procesador. Las posibles fuentes de interrupción son:

- Timer, a través de su bandera TimerIRq
- Transmisor, coprocesador CRC y memoria E2PROM, a través de su bandera TxIRq
- Receptor, a través de su bandera RxIRq
- Registro de comando, a través de su bandera IdleIRq
- Buffer FIFO, a través de sus banderas HiAlertIRq y LoAlertIRq

El RC632 informa al μ-procesador sobre el origen de una interrupción mediante el establecimiento del bit adecuado en el registro InterruptRq. La relevancia de cada bit de petición de interrupción como fuente de una interrupción puede ser enmascarada con el bit de habilitación de interrupciones en el registro InterruptEn. Si alguna bandera de solicitud de interrupción se establece en 1 (una solicitud de interrupción está pendiente) y la correspondiente bandera de habilitación de interrupción está en ”1”, la bandera de estado IRq en el registro PrimaryStatus se establece en 1. Por otra parte diferentes fuentes de interrupción pueden estar activas al mismo tiempo. Por lo tanto, se hace un OR con todos los bits de solicitud de interrupción, el resultado se envía a la bandera IRq y se conecta al pin IRQ. Los bits de petición de interrupciones están seteados de forma automática por las máquinas de estado internas del CL RC632. Adicionalmente, el microprocesador tiene acceso para setearlos o borrarlos. Una implementación especial de los registros InterruptRQ y InterruptEn permiten el cambio de un único bit de estado

sin tocar el resto. Configuración del Pin IRQ El nivel lógico de la bandera de estado IRq es visible por el pin IRQ. Además, la señal en el pin puede ser controlada por los siguientes bits del registro IRQPinConfig

- IRQInv: Si este bit es 0, la señal en el pin IRQ es igual al nivel lógico del bit IRq. Si es 1, la señal en el pin IRQ está invertida con respecto al bit IRq.

- IRQPushPull: Si este bit es 1, el pin IRQ tiene caracteristicas de una salida estandar CMOS, de otra manera la salida es open drain y un resistor externo es necesario para alcanzar un nivel alto en este pin.

Para poder hacer uso de lo descrito anteriormente se previó y reservó una entrada en el conector de expansión de la Beagleboard(ver esquemáticos), sin embargo el software empleado no hace uso del mecanismo de interrupciones sino que opera mediante polling.

Transmisor, pines Tx1 y Tx2

La señal en Tx1 y Tx2 es la portadora, centrada en 13,56 Mhz, modulada ASK 100 % con los datos a transmitir. Estos pines son conectados directamente a la antena para propagar la señal RF hacia las tarjetas RFID. La distancia de operación alcanzada es de hasta 10cm de longitud, dependiendo de la geometría de la antena, así como también adaptación de impedancia lograda, entre otros[App Notes]. Algunos registros del integrado permiten la configuración del transmisor, posibilitando entre otras cosas apagar la señal portadora en caso de ser necesario.

Conjunto de comandos

El RC632 opera como una máquina de estado capaz de interpretar y ejecutar un conjunto de comandos pre establecidos. La ejecución de uno de ellos es posible escribiendo su código correspondiente en el "Registro de Comandos", si fuera necesario el pasaje de parámetros, éstos se colocarán en el buffer FIFO mencionado antes. Una lista detallada de comandos junto con los parámetros necesarios es mostrada en la hoja de datos, entre ellos podemos destacar los siguientes: Authent, Transceive , LoadKey.

Antena RF

En lo que sigue describiremos las partes que integran la antena RF que se conecta directamente a los pines Tx1 y Tx2 del integrado descrito antes.

Filtro EMC

La frecuencia de la portadora de la señal transmitida se centra en 13,56 Mhz, sin embargo se generan también armónicos de mayor frecuencia. Para cumplir con la regulación internacional EMC es que se agrega este filtro pasa bajos, cuya frecuencia de corte debe ubicarse en 14,4 Mhz, o sea 13,56 Mhz más 847,5 Khz para permitir el ancho de banda necesario que logre el baud rate requerido en la transmisión de los bits. En sintesis el filtro ayuda a mejorar la relación señal a ruido para la señal recibida y decrementa el sobretiro en los pulsos transmitidos mejorando la calidad de la señal transmitida. Los valores propuestos para los componentes de este filtro se encuentran en las notas de aplicación [app notes.pdf]

Matching

Por su parte el circuito de adaptación de impedancia permite que la antena resuene a la frecuencia deseada, en este caso 13,56 Mhz. Los valores de los elementos que conforman este circuito deben ser estimados y sintonizados a partir del diseño del inductor de la antena. El factor de calidad total de la antena debe ser tenido en cuenta para cumplir con los requerimientos establecidos en la norma ISO14443. El mecanismo para el cálculo de los elementos que foman este circuito se detallan en las notas de aplicación [app notes.pdf].

Inductor

El inductor de la antena es quien propaga el campo magnético para la transmisión de datos hacia las tarjetas. El diseño de la antena comienza a partir de este elemento. El calculo detallado del valor del inductor se encuentra muy bien detallado en las notas de aplicación [microchip.pdf], aunque su costo y tiempo en la práctica son considerables; una estimación del valor de la inductancia puede verse en el [anexo], en el que se debe tener en cuenta los siguientes elementos: geometría de la antena, ancho y espesor del conductor del PCB, longitud de una espira, número de vueltas, etc.

Receptor

El circuito receptor de la antena se encuentra bien detallado en las notas de aplicación [app notes.pdf] y no fue necesario efectuar ningún cambio para lograr buenos resultados en nuestro diseño particular.

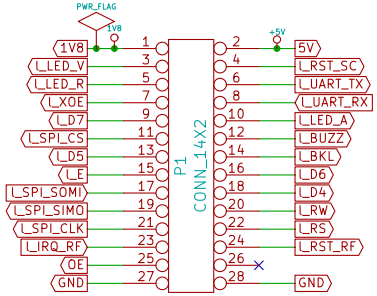
Capítulo 5

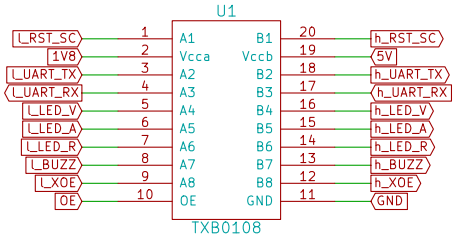
Documentos y esquemáticos del hardware

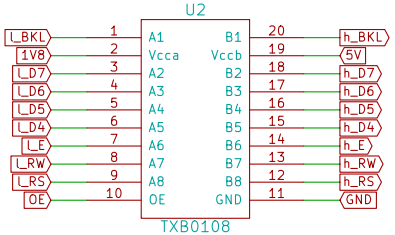
5.1. Herramientas de diseño

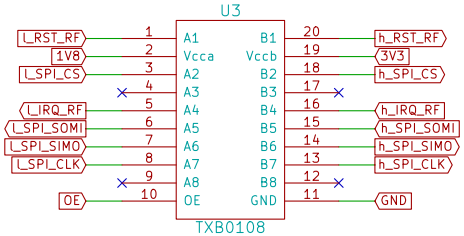
5.1.1. SBC

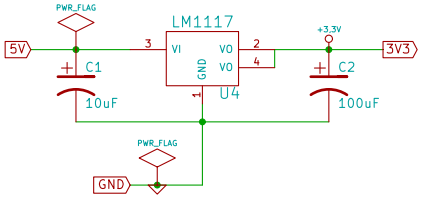
5.1.2. VLT - Conversor de Voltajes

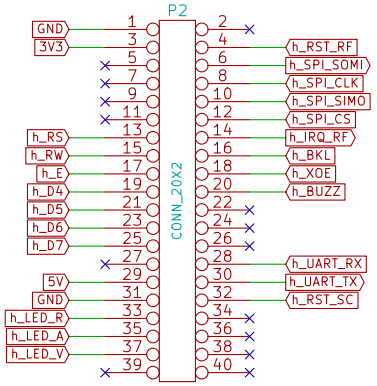






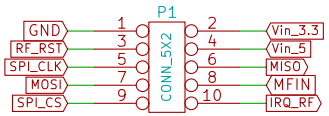


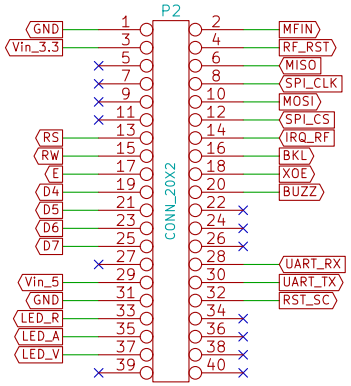


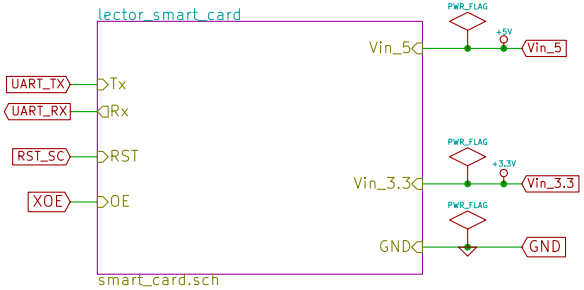


File: vlt.sch		
Sheet: /		
Title: Voltage Level Translator		
Size: A4	**Date: 20 jul 2011**	**Rev: 1.0**
KiCad E.D.A.		Id: 1/1

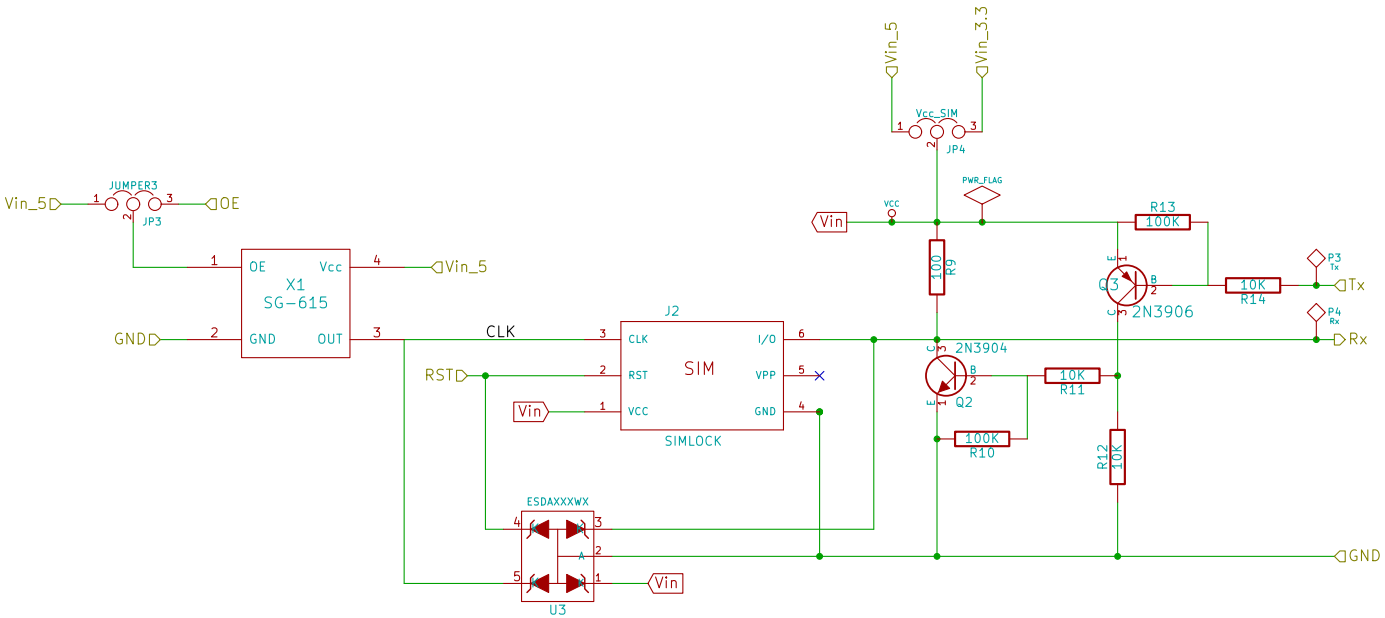
5.1.3. SCUI - Lector de tarjetas de contacto e Interfaz de Usuario



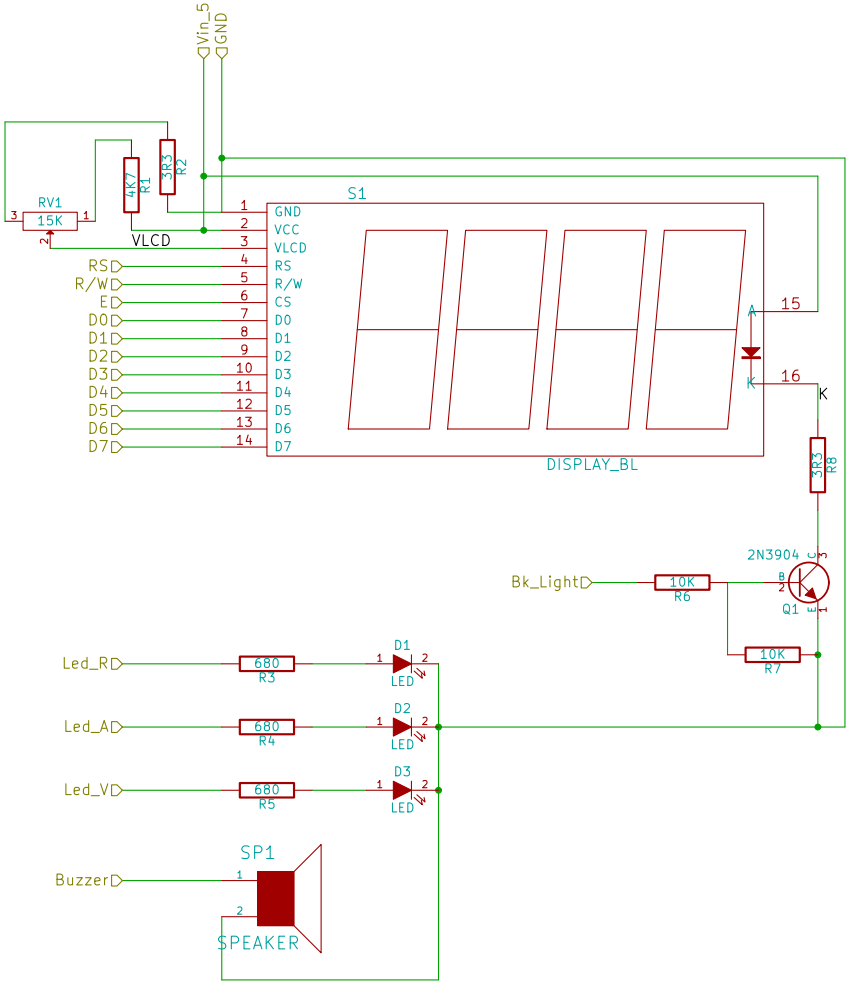




File: scui.sch		
Sheet: /		
Title: Smart Card reader & LCD16x2 User Interface		
Size: A4	**Date: 15 jun 2011**	**Rev: 2.1**
KiCad E.D.A.		Id: 1/3

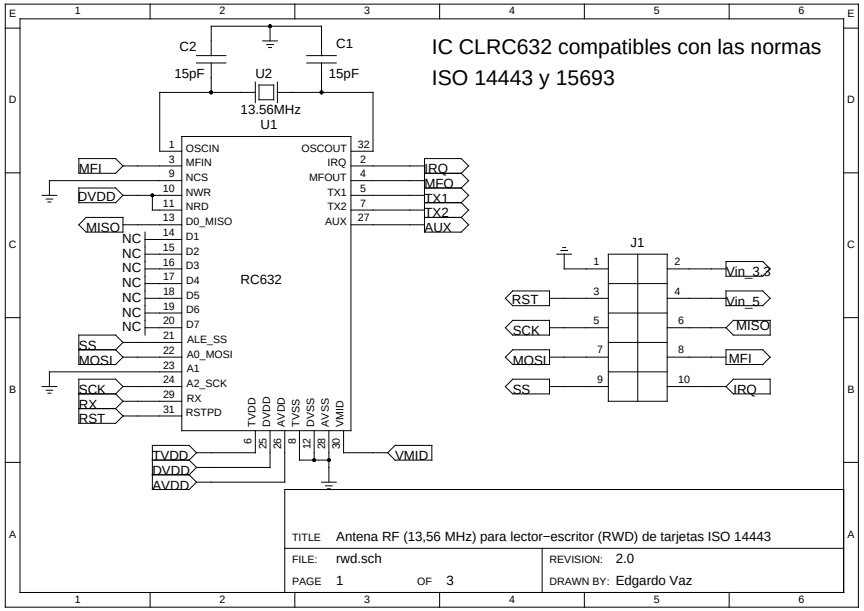


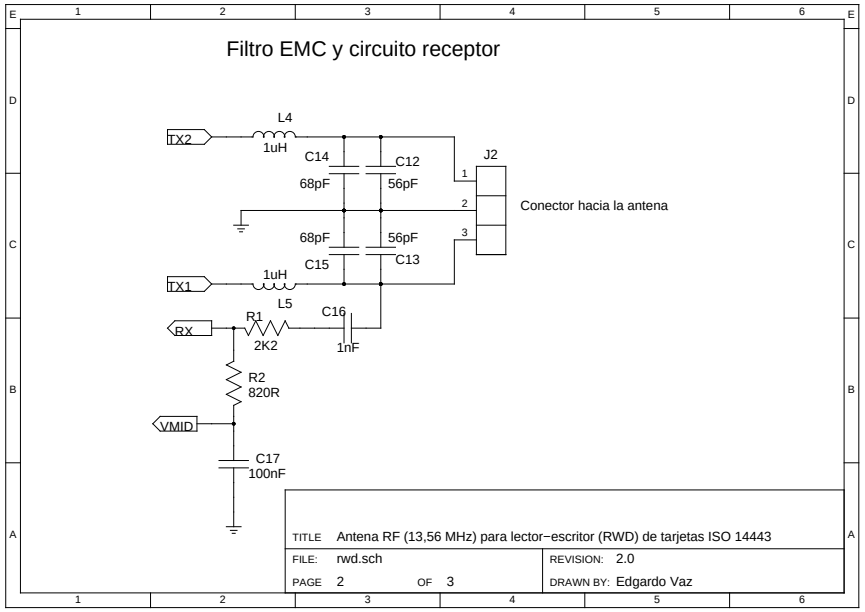
File: smart_card.sch		
Sheet: /lector_smart_card/		
Title: Smart Card reader		
Size: A4	**Date: 15 jun 2011**	**Rev: 2.1**
KiCad E.D.A.		Id: 2/3

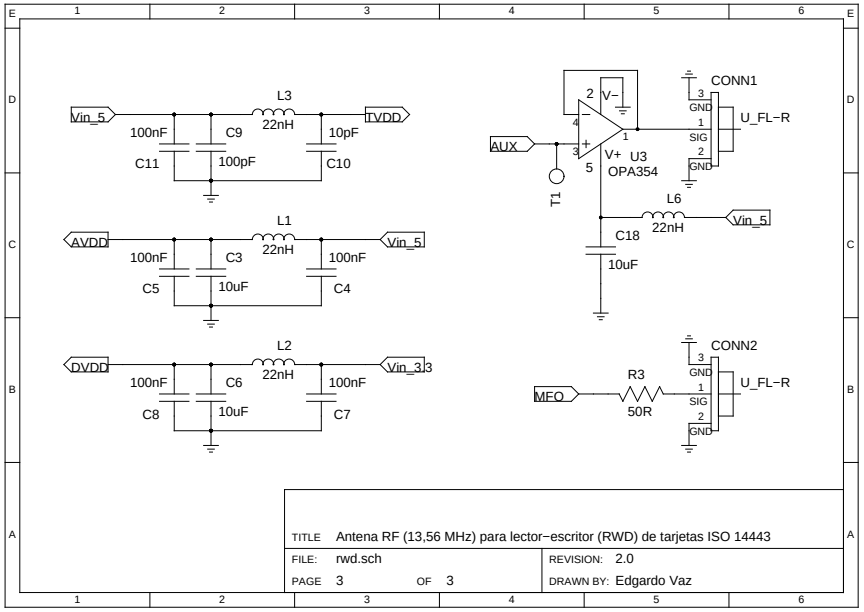


File: interfaz.sch		
Sheet: /interfaz para usuario/		
Title: LCD16x2 User Interface		
Size: A4	**Date: 15 jun 2011**	**Rev: 2.1**
KiCad E.D.A.		Id: 3/3

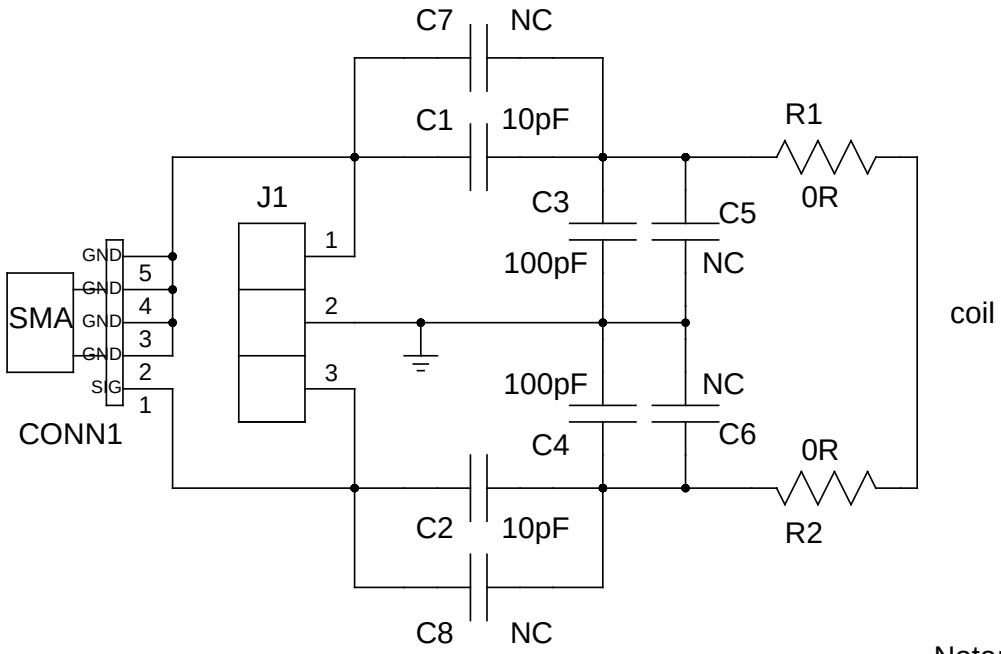
5.1.4. Lector-Escritor RFID


IC CLRC632 compatibles con las normas
ISO 14443 y 15693
C2
15pF
C1
15pF
U2
13.56MHz
U1
RC632
OSCIN
MFIN
NCS
NWR
NRD
D0_MISO
D1
D2
D3
D4
D5
D6
D7
ALE_SS
A0_MOSI
A1
A2_SCK
RX
RSTPD
OSCOUT
IRQ
MFOUT
TX1
TX2
AUX
TVDD
DVDD
AVDD
TVSS
DVSS
AVSS
VMID
MFI
DVDD
MISO
SS
MOSI
SCK
RX
RST
IRQ
MFO
TX1
TX2
AUX
J1
Vin_3.3
Vin_5
MISO
MFI
IRQ
RST
SCK
MOSI
SS
TITLE Antena RF (13,56 MHz) para lector−escritor (RWD) de tarjetas ISO 14443
FILE: rwd.sch
REVISION: 2.0
PAGE 1 OF 3
DRAWN BY: Edgardo Vaz


Filtro EMC y circuito receptor
TX2
L4
1uH
C14
68pF
C12
56pF
J2
Conector hacia la antena
C15
68pF
C13
56pF
TX1
1uH
L5
RX
R1
2K2
C16
1nF
R2
820R
VMID
C17
100nF
TITLE Antena RF (13,56 MHz) para lector−escritor (RWD) de tarjetas ISO 14443
FILE: rwd.sch
REVISION: 2.0
PAGE 2 OF 3
DRAWN BY: Edgardo Vaz


Vin_5
L3
22nH
TVDD
100nF
C9
10pF
C11
100pF
C10
AVDD
L1
22nH
Vin_5
100nF
C3
100nF
C5
10uF
C4
DVDD
L2
22nH
Vin_3.3
100nF
C6
100nF
C8
10uF
C7
AUX
V−
V+ U3
OPA354
T1
CONN1
GND
SIG
GND
U_FL−R
L6
22nH
Vin_5
C18
10uF
CONN2
GND
SIG
GND
U_FL−R
MFO
R3
50R
TITLE Antena RF (13,56 MHz) para lector−escritor (RWD) de tarjetas ISO 14443
FILE: rwd.sch
REVISION: 2.0
PAGE 3 OF 3
DRAWN BY: Edgardo Vaz

Antena formada por el inductor(coil) y el circuito de adaptación de impedancia



Nota:
Número impar de vueltas en el inductor no necesita su punto central conectado a tierra.

TITLE	Antena RF (13,56 MHz) para lector-escritor (RWD) de tarjetas ISO 14443		
FILE:	coil.sch	REVISION:	2.0
PAGE	1 OF 1	DRAWN BY:	Edgardo Vaz

Capítulo 6

Software

6.1. Introducción

6.2. Arquitectura de Software

6.2.1. Descripción

Un sistema linux en general se compone de diferentes partes que interactúan entre si, en capas ordenadas según distintos grados de abstracción respecto al hardware. Esto lo podemos apreciar en la (FIGURA) donde se muestra a grandes rasgos nuestro sistema.

El bootloader es la parte del sistema más primitiva y su función es la de cargar el kernel en memoria RAM para su ejecución. En general el bootloader se divide en dos etapas, la primera etapa del bootloader se encarga de buscar la segunda etapa del bootloader en particiones activas para luego cargarlo en RAM y ejecutarlo. La segunda etapa del bootloader se encarga de cargar una imagen comprimida del kernel en RAM y ejecutarlo. En este momento se descomprime el kernel y se cede el control al kernel. El kernel se encarga a grandes rasgos de habilitar interrupciones, configurar la memoria y montar un sistema de archivos primitivo que permite a su vez cargar los módulos necesarios para la interfáz con periféricos. Luego se monta el verdadero sistema de archivos (fileSystem). En este nuevo sistema de archivos es donde se instalarán diferentes programas y librerías para una correcta ejecución de nuestra aplicación.

En funcionamiento toda la comunicación con periféricos se realiza a través del ker-

nel que es la parte más cercana al hardware. Cada vez que ejecutamos una aplicación, esta hace uso de las librerías para poder comunicarse con el kernel, y éste se encarga de la comunicación con los periféricos. Las librerías pueden ser nativas como es el caso de la librería de lenguaje C o desarrolladas para que nuestra aplicación funcione correctamente.

6.2.2. Sistema Operativo

La beagleboard al arrancar tiene la posibilidad de buscar el bootloader en NAND o en dispositivos extraíbles tales como memorias USB o memorias SD, lo mismo sucede con el kernel. Para nuestro sistema, elegimos un arranque a través de una memoria SD ya que es más fácil de manipular.

En la siguiente FIGURA se puede ver como queda compuesta la SD. En la FIGURA se pueden distinguir dos particiones, una en formato FAT32 y otra en formato ext3. La partición con FAT32 es la partición de arranque donde se encuentra el bootloader(MLO, u-boot.bin) y el kernel(uImage). La partición con ext3 es la partición donde se encuentra el sistema de archivos(fileSystem).

El MLO es el equivalente bootloader de la primera etapa y el u-boot.bin el equivalente al de la segunda etapa. El uImage es el kernel. El fileSystem es el correspondiente a una distribución Angström.

6.2.3. Librerías

6.3. Herramientas utilizadas en el desarrollo del sistema

6.3.1. Introducción

6.3.2. Generación de MLO, u-boot.bin y uImage

6.3.3. Generación de FS

6.3.4. Croscompilación

6.3.5. Depuración de código

6.3.6. Librerías

6.4. Desarrollo

6.4.1. MLO

6.4.2. u-boot

6.4.3. uImage

6.4.4. FileSystem

6.4.5. Librerías

6.5. Ejecución de programa principal

6.5.1. Script para ejecución autónoma

Parte III

Compras

Parte IV

Anexos

Parte V

Bibliografía