

EIR – la radio por Internet

Escuche la radio con los chips más modernos

Por Von Harald Kipp y Dr. Thomas Scherer

Qué tiempos aquellos cuando aún era necesario modular los sonidos de forma analógica a una frecuencia alta para recibirlos después y de ahí conseguir algo más o menos “audible”. Hoy en día esto se hace de otra manera: se comprime la señal de audio, se organiza en paquetes IP, se transmite en flujo, se recibe, se almacena en la memoria intermedia, se descomprime y así se tiene acceso a todas las radios de Internet del mundo. Todo muy sencillo gracias al hardware de última generación del que os vamos a hablar hoy..

La radio a través de Internet es algo muy suyo, ya que frente a esta oferta de música e información ningún receptor de onda corta, por más sensible que sea, tiene nada que hacer y el tema de la calidad de sonido mejor lo dejamos. Dado que estas “emisoras de Internet” no necesitan emitir cientos de kW de alta frecuencia que generaría aún más “electromog”, el uso de una emisora de este tipo para grupos de usuarios pequeños resulta realmente barato.

Cabría elucubrar sobre las múltiples ventajas de este nuevo tipo de radios (véase el recuadro), sin embargo lo más importante es aclarar la siguiente pregunta:

¿Por qué no puro software?

En primer lugar, cabe destacar que existen múltiples programas (WinAmp, iTunes, VLC, etc.) completamente gratuitos para la mayoría de los sistemas operativos existentes con los que se puede escuchar la radio a través de Internet. De todas formas, en pleno siglo XXI todo el mundo tiene un ordenador personal, un Mac o una máquina Linux por algún lado. ¿Por qué pagar por un aparato físico, no virtual, o incluso construirlo nosotros mismos?

Pues bien, por un lado, la base de hardware de una radio de software necesita energía y, en este caso, no poca. Las personas que escuchan mucho la radio con el ordenador a través de Internet no son muy conscientes de lo nocivo que resulta para el medio ambiente. La solución que presentamos hoy tiene un consumo de energía de tan sólo 1 W. Un

ejemplo: con un funcionamiento de 10 h diarias, en un año habrías amortizado el uso la EIR (únicamente por el ahorro de energía) si lo comparamos con el ordenador de un “gamer” utilizado como radio.

Por otro lado, existen aplicaciones con las que el ordenador no funciona de forma inteligente: al conectarlo a un equipo de música estéreo, por ejemplo. Una radio por Internet casera construida con código abierto se puede ampliar fácilmente y adaptar a cualquier requisito especial y, a fin de cuentas, la EIR continúa funcionando cuando el ordenador se queda colgado o deja de funcionar ;-)

El principio

Dado que la EIR es un proyecto complejo en el que se utiliza la técnica más moderna, resulta imposible abordar todos los aspectos relevantes en un único artículo. Por lo tanto, en esta entrega nos centraremos en la descripción del hardware, su montaje y su puesta en funcionamiento. Encontrará más información en los documentos recogidos en la página web relativos a este artículo (www.elektor.es), en el sitio web de proyectos [1] y en futuras entregas.

El hecho de que tengamos que recibir, almacenar en memoria intermedia y decodificar los flujos de datos de la radio por Internet debería estar claro. Sin un microcontrolador adecuado no tenemos nada que hacer. Tal y como ya indicamos en la última entrega [3], una CPU ARM7-CPU [4] nos proporcionaría la potencia necesaria.

La **figura 1** muestra el montaje básico: la CPU está situada en el centro, hacia la parte superior, y tiene acceso a los 64 MB de SD-RAM: suficiente para la memoria intermedia y muchas “otras” cosas. La CPU tiene espacio suficiente para el firmware y, además, disponemos también de 4 MB de memoria Flash para almacenar datos. Un reloj de tiempo real, compensado con un Supercap, nos permite crear un radio-despertador o cualquier otra aplicación con función de tiempo. Para no saturar completamente la capacidad de la CPU ARM-7, en temas de decodificación de audio dispone de la ayuda del chip especializado VS1053 [5].

Y en lo que respecta al número de interfaces, la EIR no se queda corta: junto con el puerto Ethernet obligatorio (de alguna manera tendrá que acceder la EIR a Internet), dispone de una interfaz de programación USB para la transmisión de nuevo firmware, un puerto en serie y un puerto JTAG (para el debugging), así como tres conectores de ampliación tipo puerto.

Si también deseamos grabar programas, disponemos de una ranura para una tarjeta de memoria MMC-SD.

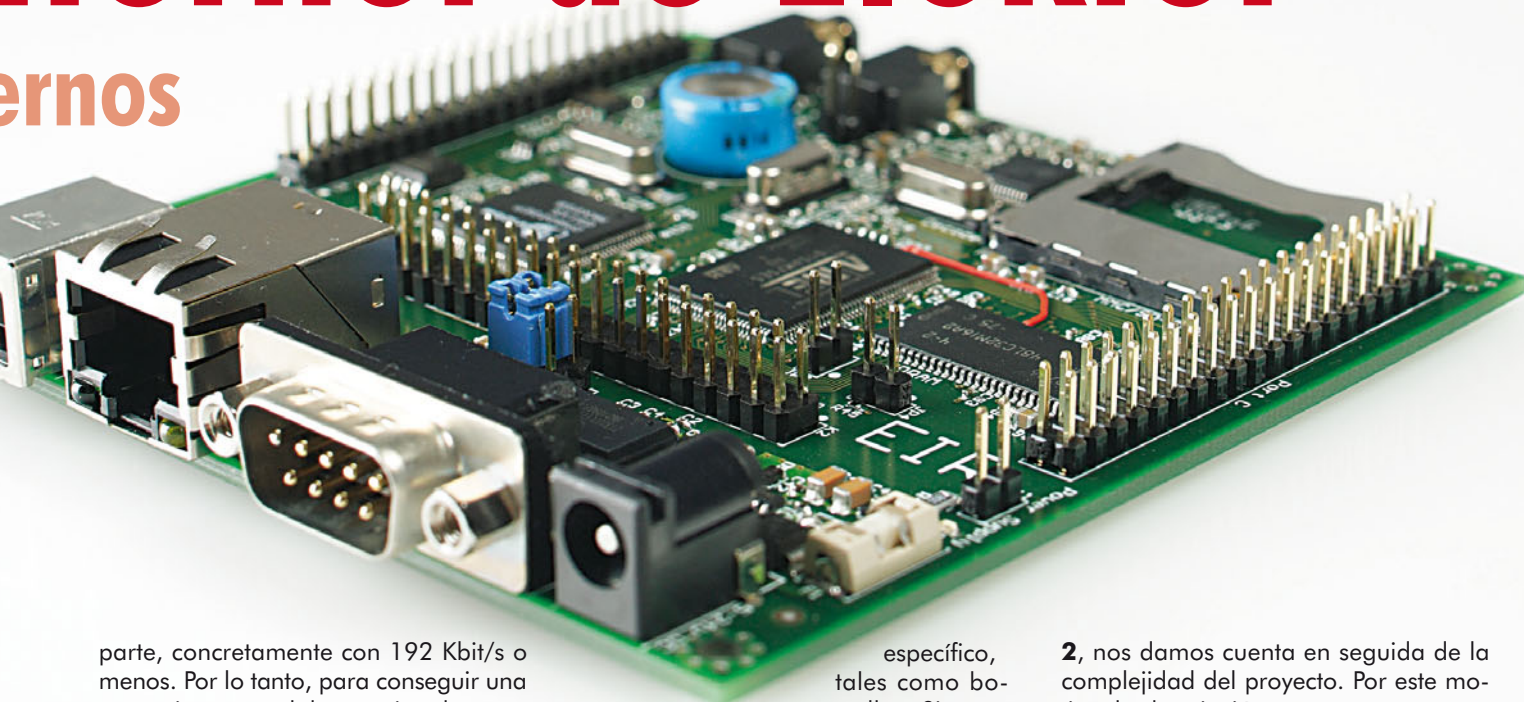
Aspectos generales

Por lo general, los flujos de datos entrantes suelen estar comprimidos de tal manera que los datos estéreo típicos, con una resolución de 16 bits y un muestreo de 44,1 kHz, se tienen que dar por satisfechos, en lugar de con aprox. 1,4 Mbit/s, con menos de una décima



Internet de Elektor

ernos



parte, concretamente con 192 Kbit/s o menos. Por lo tanto, para conseguir una memoria temporal de aproximadamente 10 s tan sólo se necesitan alrededor de 256 KB de RAM. Hoy en día esto parece poco, pero para un microcontrolador todavía es bastante. Y si queremos estar seguros y, además, deseamos tener sitio para sutilezas de Internet y "otras aplicaciones", nos ponemos en un momento en los 512 KB y más. La CPU ARM-7 elegida viene equipada con memoria SD-RAM y, de esta manera, gracias a sus 64 MB disponibles, la EIR no tendrá problema alguno de capacidad de memoria.

Como sistema operativo elegimos el relativamente modesto Nut/OS (en comparación con Linux) y que tiene suficiente con menos de 40 KB. Al fin y al cabo, para el software son necesarios aprox. 200 KB. Para los datos tendremos suficiente con 1 MB. Dado que la propia CPU dispone de más de 512 KB de memoria Flash para el software y de RAM en masa, no tendremos dificultades. Todo el software es de código abierto, con excepción del programa Flash de Atmel.

Además, el controlador es lo suficientemente potente como para utilizar la tarjeta SD para una grabación paralela del segundo flujo de audio. Y seguro que no pasará mucho tiempo hasta que alguien de la comunidad del código abierto implemente estas y cualquier otra función imaginable.

Para no limitar en ningún sentido las posibles ampliaciones, en la placa no se ha montado ningún elemento de mando

específico, tales como botones o pantallas. Sin embargo, estos pueden conectarse sin problemas a través de los conectores de ampliación disponibles. La EIR ha sido diseñada como base para ampliaciones personalizadas y, en este sentido, el firmware instalado está pensado para llevar a cabo el control a través de un sitio web integrado. Dado que el firmware es completamente libre, esto no tiene por qué seguir siendo así...

Detalles

Si echamos un vistazo al esquema de conexiones que aparece en la **figura**

2, nos damos cuenta en seguida de la complejidad del proyecto. Por este motivo, la descripción que aparece a continuación está basada en los diferentes bloques funcionales:

• Ethernet

La conexión a Internet se establece a través de una interfaz Ethernet con transformador integrado y dos LED. El LED verde se enciende durante la transferencia de datos, mientras que el LED amarillo señala la existencia de una conexión a la red. Un chip dedicado (IC10 = DM9000W) regula el tráfico Ethernet. La memoria intermedia IC9 hace posible la utilización de la entrada WAIT de la CPU con ampliaciones.

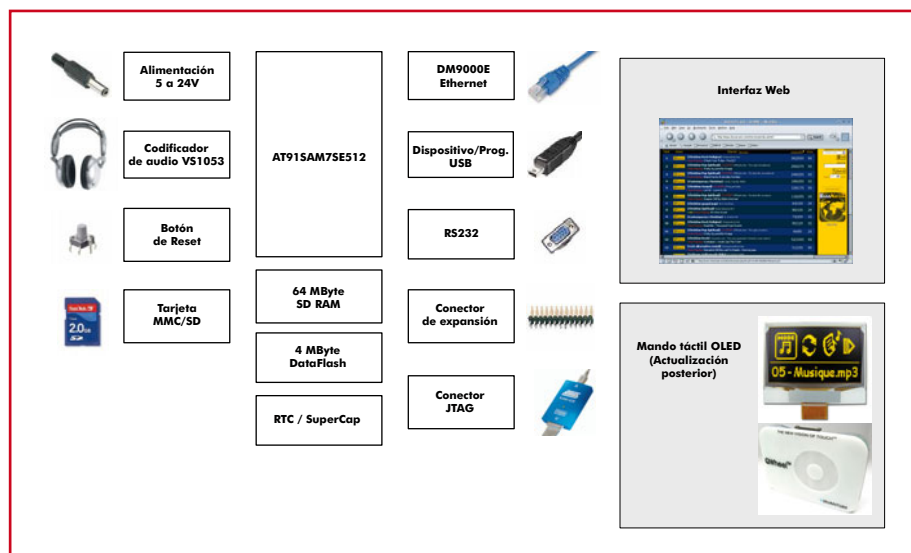


Figura 1. Circuito básico de una radio por Internet Elektor.

2a

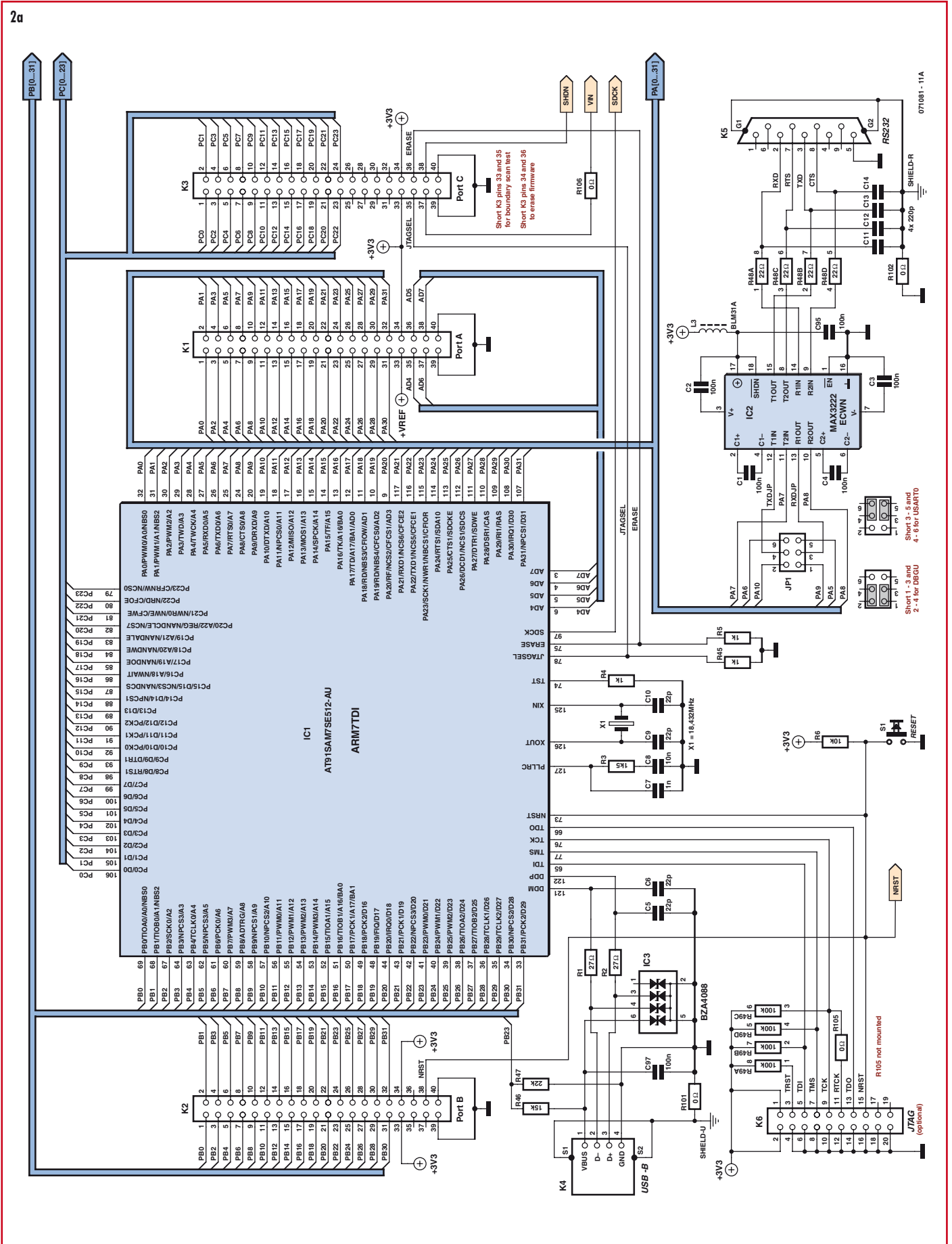
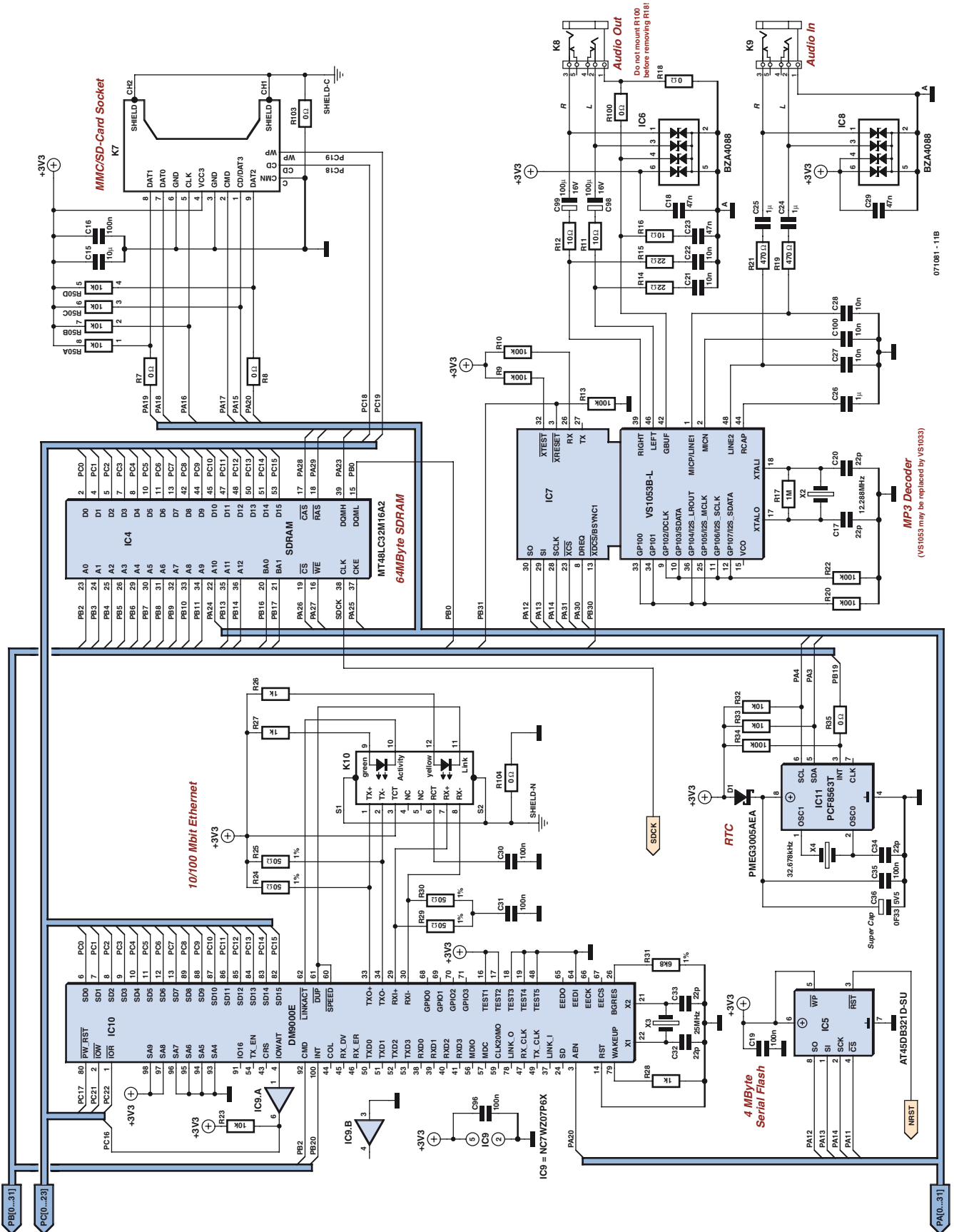


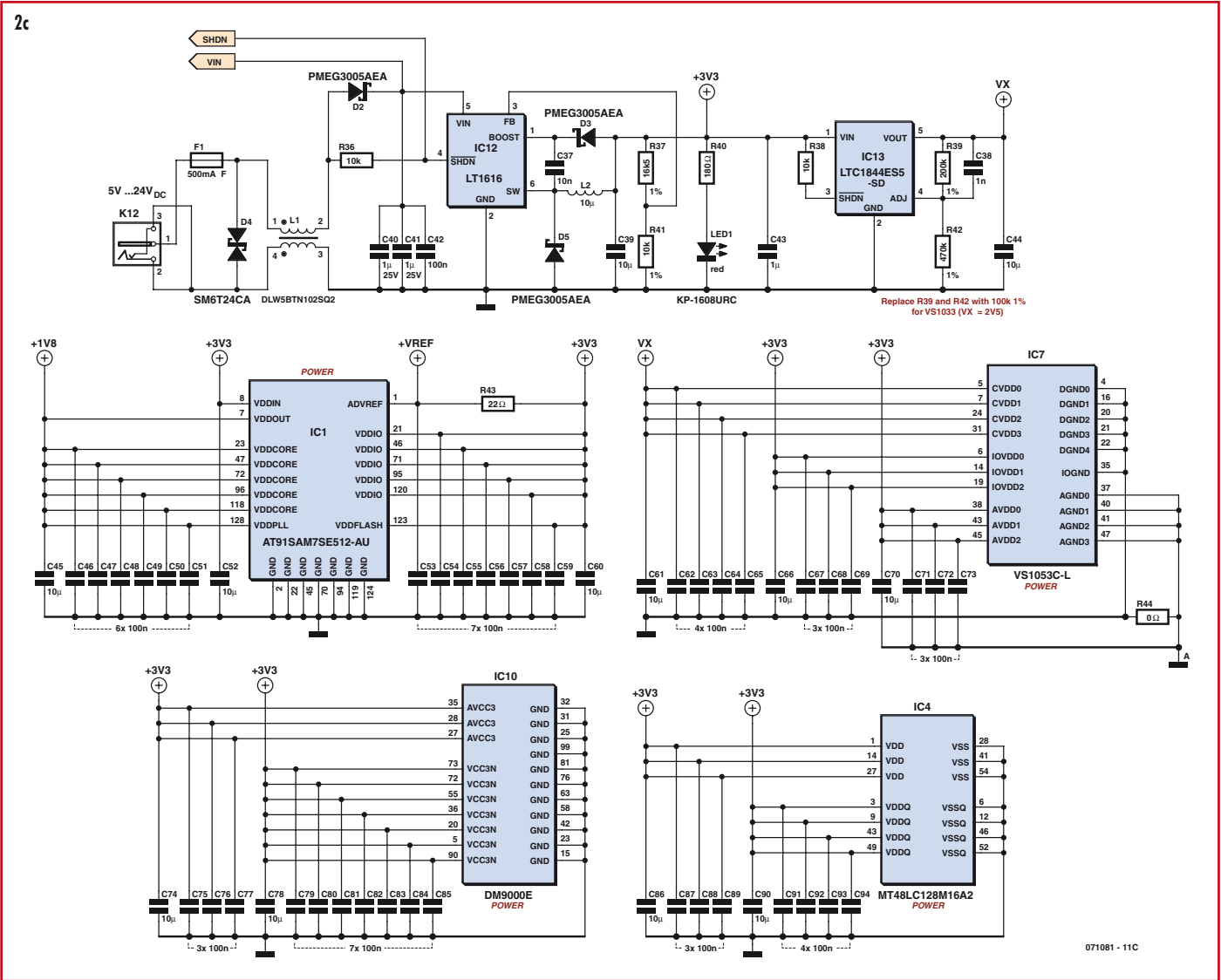
Figura 2. El circuito de la EIR pone de manifiesto de inmediato que se trata de un proyecto muy exigente.



077.081 - 11B

MP3 Decoder
(VS1053 may be replaced by VS1033)

2c



• Decodificador de audio

Aunque un ARM7 podría llegar a ser suficiente para la decodificación mediante software de datos en formato MP3 y AAC, un chip más especializado, como el IC7, descargaría en gran medida a la CPU de trabajo y decodificaría, además del formato MP3 habitual, también los formatos HE-AAC e incluso Ogg Vorbis. Al mismo tiempo, resulta evidente que el software necesario también se simplificaría. En nuestros prototipos utilizamos un modelo de VLSI. Si tenemos problemas para encontrarlo, también podremos utilizar la variante VS1033 (sin Ogg Vorbis). Aunque la CPU dispone de una salida de 1,8 V para el suministro de los periféricos, por motivos de estabilidad para el IC7 se instaló un regulador de tensión de 1,8 V propio. Dado que la variante VS1033 necesita 2,5 V, las resistencias R39 y R42 tuvieron que modificarse a 100 kΩ cada una.

• Memoria Flash adicional

Para manejar la radio se han de almacenar múltiples configuraciones, que deberán estar disponibles incluso después de un corte de corriente: isobre todo la lista de emisoras! Esto sería posible con la memoria Flash interna de la CPU, pero resulta complicado y molesto escribir en ella. Para simplificar este proceso se añadió una memoria Flash (IC5) en serie de 4 MB en la que podemos almacenar múltiples listas de emisoras y mucho más.

• Suministro eléctrico

Para que la EIR consumiera realmente poca energía se equipó con un regulador de conmutación en torno al IC12. Con tensiones de entrada entre 5 V y 24 V, disponemos de aproximadamente 5 W en el caso de 3,3 V. Dado que la EIR sólo necesita 1 W, aún nos queda suficiente energía para nuestras ampliaciones de hardware.

• Soldadura

Dado que, en este caso, se trata de una placa multicapa equipada con minúsculos SMD montados muy cerca unos de otros (véase la figura 3 y la figura 4), con los diferentes pines IC separados tan sólo 0,5 mm entre sí, Elektor ofrece una placa con todos los SMD ya montados (con VS1053). Así, sólo tendremos que montar los componentes convencionales y nos evitamos cometer errores imprevisibles. Independientemente de esto, el montaje casero con ayuda del modelo es completamente libre.

Prueba de funcionamiento

Para una primera prueba del suministro eléctrico, la parte de 3,3 V debería recibir algunos mA, es decir, no funcionar en vacío. A partir de 4 V registrados a la entrada, el regulador de conmutación debería comenzar a funcionar y, según la carga, soportar entre 50 y 150

mA. En el caso de 24 V, esto se reduce a entre 30 y 50 mA. Todo funciona correctamente: se enciende el LED 1. Una vez montados los circuitos integrados, con ayuda de un osciloscopio podemos comprobar la función de los cuarzos. Si el cuarzo X1 oscila, deberíamos poder acceder a la CPU.

La CPU viene programada de fábrica con un cargador inicial (boot loader) que, además de la mera transferencia de nuevo firmware, también hace posible la comunicación desde y hasta la memoria RAM y con la memoria Flash. En la página web de Atmel [6] encontramos el archivo "AT91-ISP.exe", que, una vez descomprimido, contiene el programa de Windows "SAM-BA". Una vez instalado éste, conectaremos la EIR con el ordenador a través del puerto USB. Al conectar la alimentación de corriente, Windows debería activar el controlador adecuado. Ahora ya podemos iniciar el SAM-BA. A continuación, debemos seleccionar "USB" como tipo de conexión y "AT91SAM7SE512-EK" como aparato compatible en gran medida con la EIR (figura 5).

Desde la página web de Elektor (en la sección del artículo) se puede descargar un firmware sencillo para fines de prueba. Con este firmware y la CPU funcionando correctamente, además de la in-

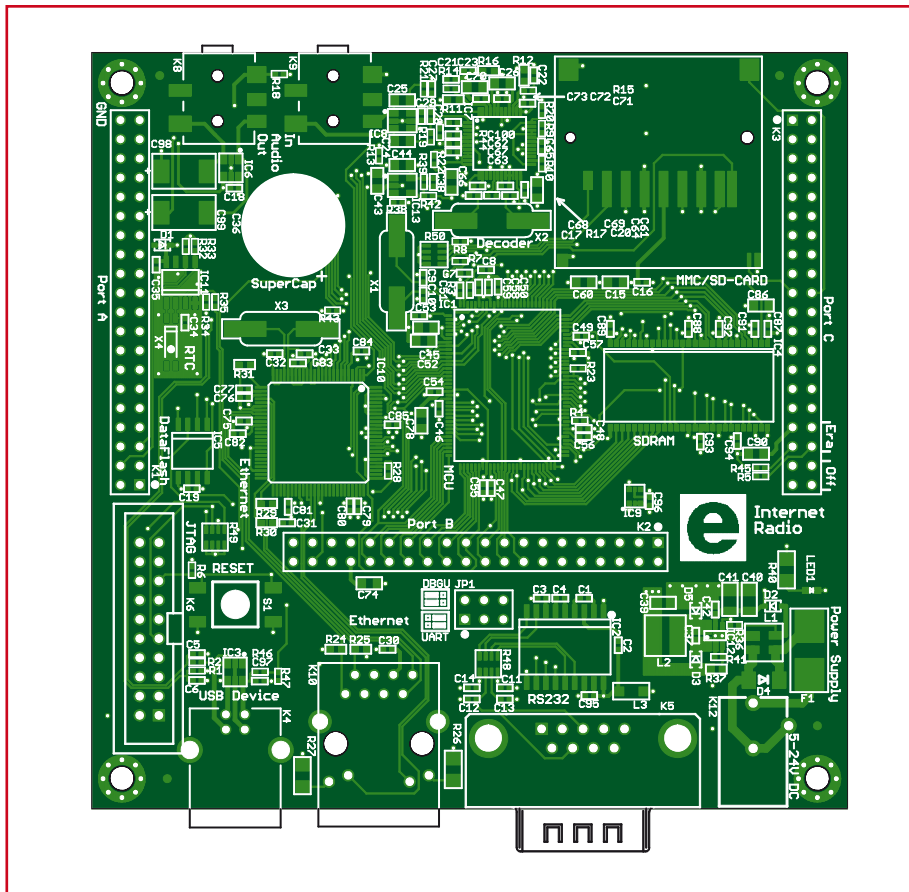


Figura 3. El plano de montaje. Para evitar problemas, existe una placa con los componentes SMD ya montados.

Lista de materiales

Resistencias:

- R1, R2 = 27Ω, SMD 0402
- R3 = 1k5, SMD 0402
- R4, R5, R28, R45 = 1k, SMD 0402
- R6, R23, R32, R33, R36, R38 = 10k, SMD 0402
- R7, R8, R18, R35, R44 = 0Ω, SMD 0402
- R9, R10, R13, R20, R22, R34 = 100k, SMD 0402
- R11, R12, R16 = 10Ω, SMD 0603
- R14, R15, R43 = 22Ω, SMD 0402
- R17 = 1M, SMD 0402
- R19, R21 = 470Ω, SMD 0402
- R24, R25, R29, R30 = 50Ω 1%, SMD 0402
- R26, R27 = 1k, SMD 1206
- R31 = 6k8 1%, SMD 0603
- R37 = 16k5 1%, SMD 0603
- R39 = 200k* 1%, SMD 0402
- R40 = 180Ω, SMD 1206
- R41 = 10k 1%, SMD 0402
- R42 = 470k* 1%, SMD 0402
- R46 = 15k, SMD 0402
- R47 = 22k, SMD 0402
- R48 = 22Ω, Array CAY16
- R49 = 100k, Array CAY16
- R50 = 10k, Array CAY16
- R100...R106 = 0Ω*, SMD 1206 (no es necesario)
- * véase el texto

Condensadores:

- (SMD cerámica de 6,3 V, salvo indicación en contrario)

- C1...C4, C16, C19, C30, C31, C35, C42, C46...C51, C53...C59, C62...C65, C67...C69, C71...C73, C75...C77, C79...C85, C87...C89, C91...C97 = 100n, SMD 0402
- C5, C6, C9, C10, C17, C20, C32, C33, C34 = 22p, SMD 0402
- C7, C38 = 1n, SMD 0402
- C8, C21, C22, C27, C28, C37, C100 = 10n, SMD 0402
- C11...C14 = 220p, SMD 0402
- C15, C39, C44, C45, C52, C60, C61, C66, C70, C74, C78, C86, C90 = 10μ, SMD 0805
- C18, C23, C29 = 47n, SMD 0402
- C24...C26, C43 = 1μ, SMD 0805
- C36 = 0,1F, Double Layer Cap FG0H104Z135
- C40, C41 = 1μ/25V, SMD 1206
- C98, C99 = 100μ/16V Tantaló, SMD

Inductividades:

- L1 = DLW5BTN102SQ2 (Murata)
- L2 = 10μ, MSS5131 (Coilcraft)
- L3 = BLM31A (Murata)

Semiconductores:

- D1...D3, D5 = PMEG3005AEA (Philips)
- D4 = SM6T24CA (STM)
- IC1 = AT91SAM7SE512-AU (Atmel)
- IC2 = MAX3222ECWN (Maxim)
- IC3, IC6, IC8 = Diodenarray BZA408B
- IC4 = MT48LC32M16A2
- IC5 = AT45DB321D-SU (Atmel)
- IC7 = VS1053C-L (VLSI)*
- IC9 = NC7WZ07P6X (Fairchild)

- IC10 = DM9000E (Davicom)
- IC11 = PCF8563T (Philips)
- IC12 = LT1616 (Linear Technology)
- IC13 = LTC1844ES5-SD (Linear Technology)
- LED1 = KP-1608URC, rot, SMD 0603 (Kingbright)

Varios:

- X1 = cuarzo de 18,432MHz, SMD HC49SM
- X2 = cuarzo de 12,288MHz, SMD HC49SM
- X3 = cuarzo de 25,000MHz, SMD HC49SM
- X4 = cuarzo de 32,678kHz, SMD MC-146
- F1 = fusible, 0,5A rápido con soporte, SMD OMNI-BLOK (fusible Littell)
- K1, K2, K3 = bloque de 40 pines, módulo de 2,54 mm, de dos filas
- K4 = conector USB-B, AMP-787780
- K5 = conector Sub-D de 9 clavijas, acodado, Norma US
- K6 = conector plano de 20 pines, módulo de 2,54 mm, de dos filas
- K7 = ranura para tarjeta SD, SMD FPS009-2700 (Yamaichi)
- K8, K9 = enchufe hembra estéreo de 3,5 mm, SMD SJ1-3515 (CUI)
- K10 = conector RJ-45 con transmisor Ethernet y LEDs, SMD, RJLD-043TC (Taimag)
- K12 = conector DC con clavija de 2 mm, TDC-002-3
- JP1 = bloque de 6 pines con 2 puentes, módulo de 2,54 mm, de dos filas
- S1 = botón, SMD. LSH (Schurter)
- Placa o placa con SMD montados EPS 071081-1
- Software de la página web de Elektor

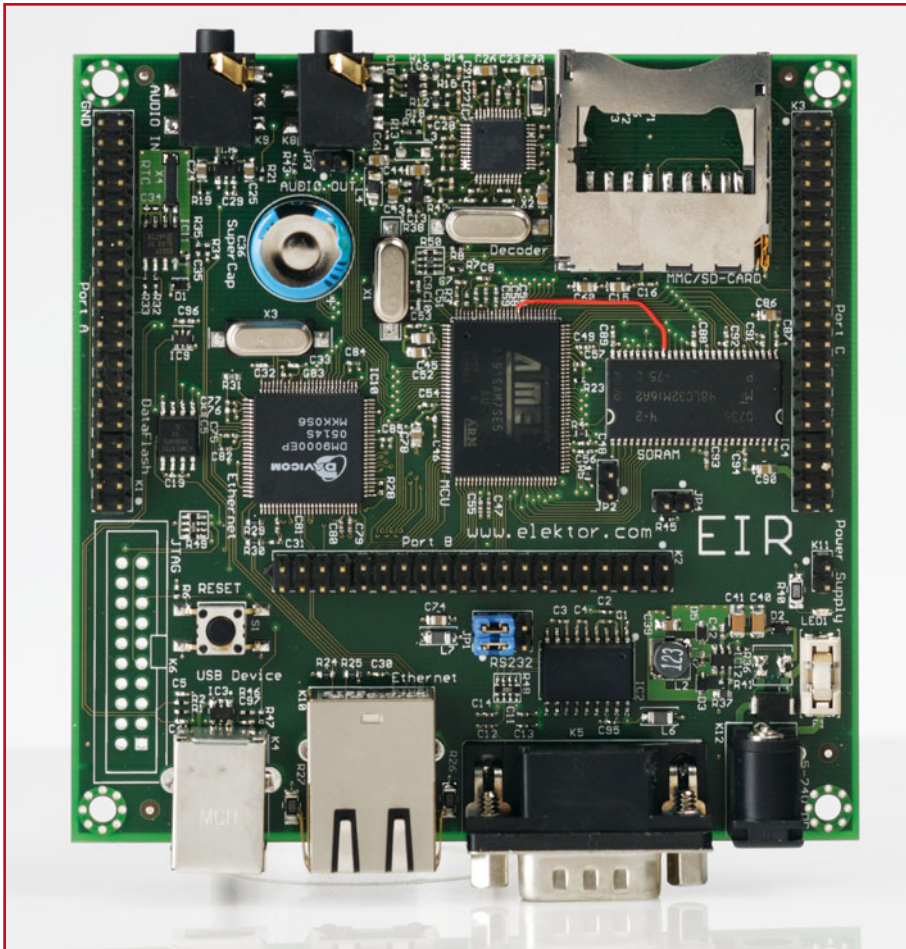


Figura 4. El prototipo equipado completamente nos muestra que soldar manualmente no resulta tan sencillo en este caso.

terfaz en serie, podemos probar ahora otros componentes, tales como Ethernet y el decodificador de audio. Una vez cargado el firmware, aún nos queda decirle a la EIR que, a partir del siguiente reinicio, debe arrancar desde ese firmware.

Para ello, en el apartado "Scripts" debemos seleccionar la rutina "Boot from Flash (GPNVM2)" y hacer clic en "Execute". A continuación, cerramos el SAM-BA y pulsamos el botón de reinicio. A partir de ahora, la EIR accede a la CPU a tra-

Tabla 1. Conector de ampliación K1					
Pin	Señal	Utilización	Pin	Señal	Utilización
1	PA0	Libre	2	PA1	Libre
3	PA2	Libre	4	PA3	TWI SDA
5	PA4	TWI SCL	6	PA5	UART0 RxD a través de JP1
7	PA6	UART0 TxD a través de JP1	8	PA7	UART0 RTS
9	PA8	UART0 CTS	10	PA9	DEBUG RxD a través de JP1
11	PA10	DEBUG TxD a través de JP1	12	PA11	DataFlash Chip Select
13	PA12	SPI MISO	14	PA13	SPI MOSI
15	PA14	SPI SPCK	16	PA15	MMC Chip Select
17	PA16	MMC Clock	18	PA17	MMC Command
19	PA18	MMC DAT0	20	PA19	MMC DAT1 a través de R7
21	PA20	MMC DAT2 a través de R8	22	PA21	Libre
23	PA22	Libre	24	PA23	SDRAM DQMH
25	PA24	SDRAM A10	26	PA25	SDRAM CKE
27	PA26	SDRAM Chip Select	28	PA27	SDRAM WE
29	PA28	SDRAM CAS	30	PA29	SDRAM RAS
31	PA30	IRQ1, MP3 Interrupt	32	PA31	MP3 Command Select
33	Vref	Referencia del convertidor AD	34	3.3V	Alimentación
35	AD4	Entrada analógica, libre	36	AD5	Entrada analógica, libre
37	AD6	Entrada analógica, libre	38	AD7	Entrada analógica, libre
39	GND	Dimensiones	40	GND	Dimensiones

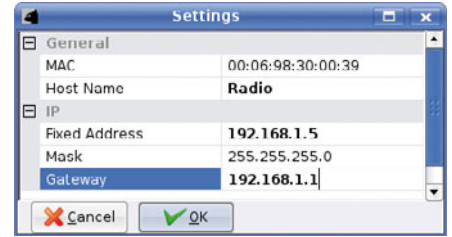


Figura 5. Captura de pantalla del SAM-BA en Windows 2000.

vés del puerto en serie mediante un cable null modem (pines 2 y 3 cruzados) y, mediante una emulación de terminal (para Windows se recomienda TeraTerm [7], para Linux, Miniterm y en el caso de los Mac siempre hay un terminal disponible), con 115,2kbaud a través de 8/0/1 (bits de datos, paridad y detención) podemos seguir las salidas de la EIR.

Escuchar la radio

Antes de que esto sea posible, debemos desinstalar el firmware de prueba de la EIR y cargar el firmware de la radio. Para poder volver a cargar el firmware, debemos puentear en primer lugar los pines 34 y 36 en K3 con un puente, pulsamos el botón "Reset" y, a continuación, volvemos a retirar el puente. A continuación, la EIR arranca de nuevo con el cargador inicial y con el SAM-BA cargaremos el firmware de la radio.

Ahora conectaremos la EIR a la red local a través de la línea Ethernet (a través de un concentrador/conmutador o un "router" con varios puertos) y en la salida de audio conectaremos unos auriculares o un amplificador.

Si la LAN o el direccionador existente disponen de un servidor DHCP activo, la EIR captura una dirección válida y comienza a reproducir la emisora de radio preconfigurada. En caso de que prefiramos direcciones IP fijas: Durante la instalación del Nut/OS ya instalamos en el ordenador la pequeña herramienta "Discover", con la que siempre podremos encontrar la EIR (figura 5) y, a continuación, configurar la dirección IP deseada. Tal y como muestra la figura 6, en la opción "Gateway" (pasarela) introduciremos la dirección del direccionador. A partir de entonces, debería ser posible escuchar la radio a través de direcciones IP fijas.

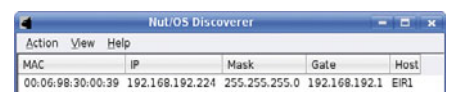


Figura 6. Con el software (en este caso bajo Linux KDE) es posible encontrar la EIR también en direcciones IP desconocidas.

Tabla 2. Conector de ampliación K2

Pin	Señal	Utilización	Pin	Señal	Utilización
1	PB0	SDRAM DQML	2	PB1	Libre
3	PB2	Bus de dirección A2	4	PB3	Bus de dirección A3
5	PB4	Bus de dirección A4	6	PB5	Bus de dirección A5
7	PB6	Bus de dirección A6	8	PB7	Bus de dirección A7
9	PB8	Bus de dirección A8	10	PB9	Bus de dirección A9
11	PB10	Bus de dirección A10	12	PB11	Bus de dirección A11
13	PB12	Libre	14	PB13	Bus de dirección A13
15	PB14	Bus de dirección A14	16	PB15	Libre
17	PB16	SDRAM BA0	18	PB17	SDRAM BA1
19	PB18	Libre	20	PB19	FIQ, RTC Interrupt
21	PB20	IRQ0, Ethernet Interrupt	22	PB21	Libre
23	PB22	DataFlash Chip Select	24	PB23	Monitor USB
25	PB24	Libre	26	PB25	Libre
27	PB26	Libre	28	PB27	Libre
29	PB28	Libre	30	PB29	Libre
31	PB30	MP3 Data Select	32	PB31	MP3 Hardware Reset
33	3,3 V	Alimentación	34	3,3 V	Alimentación
35		No utilizado	36		No utilizado
37		No utilizado	38	NRST	Hardware Reset
39	GND	Dimensiones	40	GND	Dimensiones

Futuras perspectivas

Como ya hemos mencionado en varias ocasiones, la EIR es un concepto completamente abierto. Tanto el software como el hardware se pueden ampliar a voluntad (a través de conectores de ampliación) y en Elektor seguro que dispondremos de muchas cosas para seguir leyendo al respecto.

En lo que respecta a las herramientas y fuentes de software, así como a las posibles actualizaciones de los mismos, lo mejor es echar un vistazo de vez en cuando a la página web de proyectos [1] de la empresa egnite. Aquí encontraremos textos fuente e instalaciones para Windows, Linux y OS X. Además, también encontraremos enlaces a entornos de desarrollo y otros proyectos de código abierto.

Todo es posible. Algo obvio sería, por ejemplo, un par de botones y una pantalla LCD, para que podamos utilizar la

Radio por Internet

Un vistazo a Internet y nos quedamos atónitos: si introducimos en Google la palabra clave "Internetradio", recibiremos como respuesta más de 2,3 millones de entradas. Se trata, por lo tanto, de un tema de máxima actualidad. Los primeros experimentos con "programas" transmitidos en paquetes datan de 1993, casi al mismo tiempo que el primer navegador aprovechable, el NCSA Mosaic, y, por decirlo de alguna manera, en la hora cero del Internet comercial. Las primeras emisoras de radio "reales" comenzaron muy pronto a distribuir también vía Internet-streaming sus programas que, hasta ese momento, sólo habían sido retransmitidos por radio. Hoy en día, con una conexión normal a Internet podemos acceder a decenas de miles de programas de radio. Junto con una gran cantidad de programas de diferentes temáticas, también tenemos ya acceso a casi todas las emisoras públicas y comerciales.

Por "streaming" integrado, técnica para la transmisión actual de datos (por ejemplo, audio y vídeo) en tiempo real entendemos flujos de datos lo más continuados posibles de entre los que la emisora debe ofrecer

por cliente un flujo extra, algo que genera muchísimo tráfico y, por lo tanto, podría resultar bastante costoso para muchos oyentes. Para mantener la tasa de transmisión de datos dentro de unos límites razonables, los datos se comprimen antes de su envío y se descomprimen una vez llegados a su destino. Una radio por Internet, ya sea basada en software o como hardware, debe disponer de un decodificador de streaming como MP3, Ogg Vorbis o Real Audio.

Dado que con los protocolos http y ftp habituales de Internet no se puede garantizar un tiempo de ejecución estable para los distintos paquetes de datos, el receptor precisa una memoria intermedia de datos suficiente, lo que retrasa algunos segundos la recepción y sólo es capaz de ofrecer una reproducción "casi en vivo". El zapping rápido queda por lo tanto descartado. En lugar de ello, gracias a la digitalización con una calidad de sonido estable, nos conformamos con un alcance internacional y una variedad ilimitada de programas en comparación con la radiodifusión clásica. Además, en principio es posible la "recepción" de conservas (= programas que nos hemos perdido) en forma de "Audio on Demand", algo que la radio convencional no puede ofrecer.

Tabla 3. Conector de ampliación K3

Pin	Señal	Utilización	Pin	Señal	Utilización
1	PC0	Bus de datos D0	2	PC1	Bus de datos D1
3	PC2	Bus de datos D2	4	PC3	Bus de datos D3
5	PC4	Bus de datos D4	6	PC5	Bus de datos D5
7	PC6	Bus de datos D6	8	PC7	Bus de datos D7
9	PC8	Bus de datos D8	10	PC9	Bus de datos D9
11	PC10	Bus de datos D10	12	PC11	Bus de datos D11
13	PC12	Bus de datos D12	14	PC13	Bus de datos D13
15	PC14	Bus de datos D14	16	PC15	Bus de datos D15
17	PC16	Bus de datos D16	18	PC17	Bus de datos D17
19	PC18	Bus de datos D18	20	PC19	Bus de datos D19
21	PC20	Libre	22	PC21	Bus de dirección/datos NWE
23	PC22	Bus de dirección/datos NRD	24	PC23	Ethernet Chip Select
25		No utilizado	26		No utilizado
27		No utilizado	28		No utilizado
29		No utilizado	30		No utilizado
31		No utilizado	32		No utilizado
33	3,3 V	Alimentación	34	3,3 V	Alimentación
35	JTAGSEL	Boundary Scan Enable	36	ERASE	Firmware Erase
37	VIN	5-24 V estabilizados a través R106	38	SHDN	Power Shutdown
39	GND	Dimensiones	40	GND	Dimensiones

EIR no sólo a través del navegador web, sino de forma mucho más elegante: como aparato independiente. La ranura para tarjetas pide a gritos poder ser utilizada además como reproductor MP3.

(071081e)

Literatura y enlaces

[1] Sitio web de proyecto de egnite:

www.ethernut.de/de/hardware/eir/

[2] Artículo de Wikipedia:

de.wikipedia.org/wiki/Internet-Radio

[3] Ethernut y la familia Kipp:

Elektor marzo de 2008.

[4] Información sobre la CPU ARM7:

www.atmel.com/products/at91/

[5] Información sobre VS1053:

www.vlsi.fi/en/products/vs1053.html

[6] Enlace a AT91-ISPexe:

www.atmel.com/dyn/resources/prod_documents/Install%20AT91-ISP%20v1.10.exe

[7] Windows Terminal:

tssh2.sourceforge.jp/