

Elektor Bus
vuelve a la vida

+ Reflectometría en el Dominio del Tiempo

Elektor

www.elektor.es

Cálculo analógico redescubierto

Máquina del CAOS

Hacer malabares con bits de audio
DSP de audio para aplicaciones DIY

Inversor de Onda Senoidal
con Corrección de Factor de Potencia

El Sueño del Vuelo Eléctrico
volando en silencio y sin contaminar

Platino

Circuito multifuncional para
aplicaciones con microcontrolador

ISSN 0211-397X



9 770211 397008

Más allá de lo impreso

Muchos de vosotros asociará Elektor con la revista impresa de la que la edición española está en circulación en los quioscos desde nada menos que 1981. De hecho, ahora estás leyendo la edición número 376 – en papel, y tenemos la intención de imprimir un buen número de ediciones más para que las hojees, leas, des al perro tras la orejas o las manches con el soldador. Sin embargo en los últimos años Elektor ha desarrollado un buen número de actividades en las que el papel ya no tiene nada que ver, aunque las consideramos parte integrante de nuestras actividades de publicación en general. Además de la descaradamente obvia web elektor, es algunas de estas actividades pueden ser desconocidas para ti, así que mencionaré algunas. Para empezar, están nuestras PCBs, DVDs, libros, ediciones especiales, kits y módulos. Las PCBs de Elektor –reconocidas por su calidad– están a la venta desde la primera edición aunque los kits y módulos se han incorporado hace relativamente poco y son particularmente útiles para aquellos de vosotros que se mueren por construir nuestros más ambiciosos proyectos pero temen o no pueden encontrar o manejar los componentes empleados. Nuestro primer ‘módulo’ fue el legendario GBDSO (Game Boy Digital Sampling Oscilloscope). Algunos de nuestros kits más recientes son ‘híbridos’ en el sentido en el que vienen con los componentes SMD pre-montados en la placa. De las opiniones de los lectores aprendimos que muchos de vosotros disfrutáis aún soldando lo componentes de taladros pasante, por lo que decidimos suministrarlos separadamente con el kit para que podáis empuñar el viejo soldador (de baja potencia, claro está) durante poco más de una hora. Elektor también está fortaleciendo su cartera de “e-ventos” mediante la organización de webinarios sobre publicaciones de éxito. Nuestros socios son tanto empresas como autores reconocidos como autoridades en sus respectivos campos. Nuestros webmasters Patrick y Denis han creado un canal Elektor en YouTube, www.youtube.com/elektorim. Hemos descubierto que pequeñas películas (no importa cuán primitivamente estén hechas) sobre nuestros temas técnicos son una excelente manera de atraer no solo a los recién llegados de la generación MTV si no también a los que ahora están disfrutando de la jubilación y han redescubierto Elektor buscando con éxito en Google. Todo el mundo es más que bienvenido, es de esperar que haya algo que te agrade en nuestra creciente gama de productos, ya sean de papel o no.

¡Disfruta de esta edición!
Eduardo Corral, Editor



6 Colofón

Información Corporativa de la revista Elektor.

8 Noticias Locales

Un paseo mensual por lo último en el mundo de la electrónica.

14 El sueño del vuelo eléctrico

¿Hemos recorrido un largo camino desde la aeronave eléctrica (1884) hasta el Green Flight Challenge (2011)?

22 Audioguía

La primera aplicación de la placa Platino de Elektor es un reproductor MP3 activado por RFID utilizado como un guía personal en museos.

26 ¡Que viene el bus! (8)

Unos pocos componentes unidos al bus nos permiten ver cómo funciona el protocolo en una aplicación real de control.

32 Reflectometría en el Dominio del Tiempo

Una guía de iniciación a la localización de cortos y abierto en cables largos, complementada con el esquema de un sencillo generador de pulsos.

36 Alternativa al Fan-Flash

La lógica digital de la Vieja Escuela imita un estroboscopio que hace que las palas del ventilador de tu PC parezcan detenidas.

39 Resultados mixtos

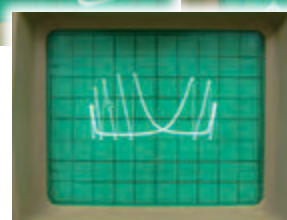
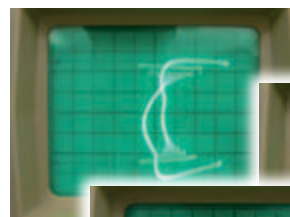
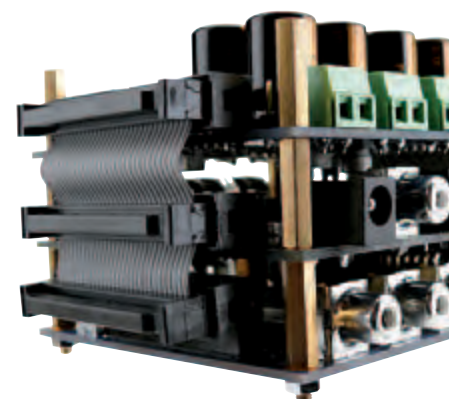
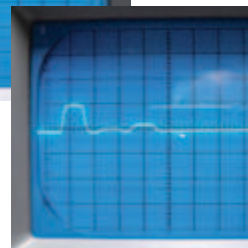
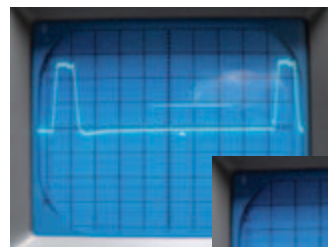
Un breve informe de los intentos de dos estudiantes para diseñar un mezclador de audio como parte de su formación electrónica.

40 Detector de radiación verificado

El detector de rayos gamma de bajo coste de Elektor fue puesto a prueba en un laboratorio nuclear profesional.

42 Bits Rebeldes

La doma de un generador de velocidad de transmisión en un PIC.



SUMARIO

Volumen 32
Octubre 2011
nº 376

32 Reflectometría en el Dominio del Tiempo

Desde luego, un óhmetro es un instrumento muy útil para comprobar circuitos abiertos y cortos en hilos y cables. Pero encontrar un abierto o un corto no es fácil cuando el cable es largo, o si el fallo está oculto dentro de un muro o bajo la calle. Localizar abiertos y cortos parciales, o simplemente malas conexiones, es a veces incluso más difícil. Aquí es donde la TDR se vuelve útil.



46 Placa Polivalente Para Circuitos con Microcontroladores AVR

Platino nació el 20 de Noviembre de 2010 en Holanda; es una versátil placa para circuitos basados en un controlador AVR de 8 bits. Tras varios meses de preparación, este hermano pequeño de J²B se reunió con su primer microcontrolador AVR en Junio de 2011. Dado que aún no se sentía adecuadamente preparada para enfrentarse a la jungla electrónica, decidió optimizarse tomando a su ídolo Arduino como ejemplo.



60 Hacer Malabares con bits de Audio

Los aficionados al audio limitan, generalmente, su afición al dominio analógico dado que las posibilidades de experimentar uno mismo con lo digital son muy limitadas. Hay muy pocos aparatos asequibles con los que realizar todo tipo de tratamientos digitales de audio e incluso empezar con DSP requiere un conocimiento profundo de esta materia. Con los módulos de miniDSP se pueden realizar todo tipo de funciones de audio fácilmente sin necesidad de profundizar en el tratamiento digital de audio.



66 Retrónica: La Máquina del Caos (2)

El ordenador analógico que nos propusimos describir en la entrega anterior fue construido con módulos de cálculo independientes para la multiplicación, integración, suma y escalado, combinados para representar el sistema de ecuaciones de Lorenz de 1963. En la práctica, puede hacer más que 'solo' generar la mariposa de Lorenz.



42 Útil, este display

¿Estás aburrido de romper esos frágiles LCD de cristal?

43 Inversor de Onda Senoidal con Corrección de Factor de Potencia

Un Proyecto de un Lector que describe un inversor de 24 VDC a 230 VAC optimizado para alimentar bombillas Osram de bajo consumo de hasta 100.

46 Platino: Placa polivalente para circuitos con microcontroladores AVR

Un encuentro inusual con Platino, una "placa que viene con un circuito".

52 Curso de audio DSP 4 - Parte 4: probando el hardware

Este mes probamos el hardware con la ayuda de unos pocos programas DSP pequeños.

60 Hacer malabares con bits de audio

Un DSP de audio para aplicaciones de construcción propia.

66 Retrónica: La Máquina del Caos: Cálculo Analógico Redescubierto (2)

Las usuales características de la electrónica "extraña y antigua".

70 Hexadoku

Nuestro rompecabezas mensual con un toque de electrónica.

76 Próximo número

Un avance de los contenidos de la próxima edición.

El sueño del vuelo eléctrico

De los aviones eléctricos (1884) al Green Flight Challenge (2011)



Ernst Krempelsauer

El sueño del vuelo eléctrico ha dejado de ser un sueño como tal. El primer avión de transporte apto para un funcionamiento eléctrico está en producción desde 2004, y la compañía que gestó el Airbus, EADS, ha estado desarrollando conceptos de propulsión completamente eléctrica para los aviones comerciales del futuro.

Hacemos repaso del estado actual de la movilidad eléctrica y al concurso más prestigioso al respecto, el Green Flight Challenge, para el cual la NASA aporta 1,65 millones de dólares.

(Imagen: EADS)

Mientras los coches eléctricos despiertan un enorme interés, la movilidad eléctrica en el campo de la aeronáutica no llama apenas la atención. Pero la electricidad en los aviones es casi tan antigua como lo es la propia aeronáutica: su nacimiento se remonta al

año 1884, con el vuelo de una aeronave propulsada gracias a una enorme batería eléctrica.

De las aeronaves a los aviones eléctricos

La aeronave “La France” (**figura 1**), construida por Charles Renard y Arthur Krebs en las inmediaciones de París, funcionaba con un motor de corriente continua [1] de 5,6 kW (posteriormente 6,3 kW), alimentado mediante una batería de 435 kg de zinc-cloro. La fuente de alimentación inventada por Renard está considerada la primera aplicación de la batería de flujo redox [2], redescubierta en los años 50 y utilizada actualmente como buffer en la red. Al motor eléctrico le costaría casi 70 años volver a volar:

El constructor de maquetas de avión Fred Militky, tras numerosos intentos fallidos, desarrolló una aeronave de funcionamiento eléctrico llamada “Silentius” [3], que en 1960 salió al mercado como kit de montaje. Para su funcionamiento se servía de un propulsor operado por un pequeño motor sin núcleo de 2 a 4 V (Micro T 03/15) de Faulhaber con una corriente máxima de 1,5 A, con lo que no era capaz de entregar más de 5 W de potencia, con un rendimiento del



Figura 1. Pionero por excelencia: aeronave “La France”, de 1884, la primera en ser totalmente dirigible y funcionar eléctricamente, aparte de suponer la primera aplicación de la pila de flujo redox del mundo. (Imagen: Wikimedia Commons/Foto de 1885, 2001 National Air and Space Museum, Smithsonian Institution)

70 %. Su fuente energética era una pequeña batería de plomo Rulag de 2 V. El modelo de 140 gramos aún puede construirse hoy en día a partir de los planos; ¡el motor de Graupner [4] lleva disponible 51 años!

El primer vuelo eléctrico tripulado “más pesado que el aire” data de 1973, y la iniciativa también vino por parte de Fred Militky.

Planeador con motor eléctrico

El primer vuelo eléctrico tripulado del mundo, sugerido por Militky [5] tuvo lugar en Octubre de 1973 en Wels (Austria): un planeador a motor HB-3, que obtenía electricidad a partir de gasolina mediante un motor de corriente continua Bosch de 10 kW [6] y una pesada batería NiCd Varta, de 125 kg. Con ello el constructor Heino Brditschka realizó un vuelo eléctrico de 14 minutos, alcanzando una altura de 360 metros. Esta potencia se superó en 1981 – con un avión solar basado en sus portentosos predecesores de los años 70.

Mientras que el funcionamiento con células solares por ahora sólo es algo experimental, ha podido apoyarse en modernos sistemas de baterías ya disponibles en el mercado. Las mejores condiciones para “electrificar” establecen que se trate de un planeador ligero con motor reversible, en el cual el propulsor se oculte tras la cabina cuando esté en modo planeador. El motor y el propulsor sólo se utilizarían para el despegue y para alcanzar la altura necesaria para planear, así como un rotor adicional y una hélice por si acaso, para evitar descensos bruscos en aterrizajes fuera de pista. La solución convencional se basa en un motor de dos tiempos o Wankel de relativa baja potencia (de 15 a 50 kW) y poco combustible, con algunos inconvenientes que no existen con un funcionamiento eléctrico: ruido, vibraciones, arranque y funcionamiento no del todo fiables, aparte de su tecnología, de cierta complejidad (motor de arranque, freno propulsor, bloqueo para la posición de reposo, etc.) en un espacio reducido. Aquí, el motor eléctrico representa una solución más elegante. Sin embargo, esto no es cierto en el caso de la batería: un tanque de gasolina (lleno) con un volumen de 20 l es capaz de entregar una energía 175 kWh, mientras que una batería de ión-litio del mismo tamaño almacena tan sólo alrededor de 5 kWh; esto lo hace aún más difícil. No obstante, el sistema eléctrico (motor y regulador), con su rendimiento de hasta el 90 %, resulta tan eficiente, que a pesar de la baja densidad energética de la batería es capaz de suministrar al planeador la potencia necesaria. Aunque claro, antes que las ventajas que ofrece el funcionamiento eléctrico, en el mercado prevalece la pregunta de los costes.

El primer planeador a motor eléctrico comercial con propulsor que se ocultaba fue el AE-1 Silent, comenzando su desarrollo en 1991 por parte de Air Energy [7] en Aachen, logrando en 1997 su primer vuelo y cuyo prototipo obtuvo la calificación en 1998 del primer avión eléctrico del mundo (con 12 m de envergadura y un peso en vacío de 195 kg, siendo un avión ultraligero). El motor eléctrico es de 13 kW y pesa 8,5 kg, la versión actual de la batería de ión-litio pesa 35 kg es capaz de almacenar 4,1 kWh.

El primer planeador eléctrico de alta potencia fue el Antares de Lange Aviation [8], que voló para el DLR también como avión de investigación (**figura 2**), funcionando con una pila de combustible de hidrógeno [9]. Se fabrica en serie desde 2004, tratándose



Figura 2. El Antares 20E de Lange Aviation – aquí en su versión DLR con tanque de hidrógeno y pila de combustible; se trata del primer avión considerado eléctrico EASA debido a su motor. (Imagen: DLR)

del avión con el único motor aprobado por EASA como eléctrico: El EM42 es un motor de continua sin escobillas (motor de rotor externo, con imanes permanentes), de 25 cm de diámetro y 27 cm de longitud, que con una tensión de 190 a 288 V y una corriente de funcionamiento de 160 A entrega una potencia máxima de 42 kW. El motor en su versión de serie ofrece un rendimiento superior al 90% y un par motor máximo de 216 Nm con un peso de 29 kg, aparte de aproximadamente otros 10 kg más correspondientes a toda la electrónica de potencia. El motor y la electrónica [10] fueron desarrollados entre 1996 y 1998 en el HTA Biel (hoy en día el BFH TI de Berna), en suiza. Una de sus peculiaridades es el funcionamiento con motor paso a paso, capaz de dar la posición exacta del propulsor cuando este se guarda.

Como almacenamiento energético se utilizan 72 células de ión-litio VL41M de SAFT [11]. Una célula sola suministra 44 Ah a 3,7 V, resultando en un total de 266 V y 12 kWh. Así, Antares puede alcanzar en escasos cuatro minutos una altura de 1000 m y en unos 13 una (máxima) de 3000 m. Sin turbinas es capaz de mantener esa altura durante más de 1,5 horas, y más de 150 km. La electrónica incluye el sistema de control del motor, la supervisión de la batería (telemetría vía módem GSM inclusive), el calefactor de la batería y el cargador para ésta, montado fijo (carga de 9 h a 230 V o 110 V). Las células SAFT-VL41M pueden alcanzar los 3000 ciclos de carga y a 20 °C una vida útil de 20 años, además, estarán disponibles hasta 2031.

La técnica utilizada actualmente se basa en el motor del planeador de alta potencia biplaza Arcus-E de Schempp-Hirth Flugzeugbau [12] (cuyo primer vuelo fue en 2010).

No obstante, el primer avión eléctrico biplaza del mundo fue el planeador a motor Taurus Electro de Pipistrel [13], Eslovenia, que ya

Vuelos con pila de combustible de H₂

Ya tenemos constancia de proyectos sobre aviones eléctricos con pila de combustible desde hace unos 10 años. Uno de ellos por parte de la NASA, el otro de la mano de Boeing. En el Aero 2003 (Friedrichshafen) se mostró un planeador a motor (Super-Dimona) como ensayo del proyecto de Boeing, y en breve se anunció su primer vuelo el 17 de Diciembre de 2003 (100º aniversario del vuelo con motor). En realidad finalmente tuvo lugar en marzo de 2008, tratándose del primer vuelo tripulado con una pila de combustible por hidrógeno. Su potencia sólo le permitía planear, con lo que para el despegue se implementó una batería de ión-litio. El despegue sin ayuda de una batería llegó en Julio de 2009, con el planeador a motor eléctrico Antares descrito en este artículo, el avión de investigación Antares DLR-H2 [27] con tanque de hidrógeno y pila de combustible de 25 kW, albergados en dos compartimentos adicionales bajo las alas. Con ello, el Antares DLR-H2 es capaz de alcanzar los 170 km/h y recorrer una distancia de 750 km, lo que equivale a 5 horas de tiempo de vuelo, a una altura record de 2558 metros.

Desde 2010, se encuentra en desarrollo el sucesor basado en el Antares, el DLR-H3 [28], que incorporará cuatro contenedores externos,



un propulsor fijo y contará con 23 m de envergadura, siendo en 2012 uno de los aviones con mayor rendimiento: 6000 km de distancia y una duración del vuelo de 50 horas, transportando 200 kg de carga útil. (Imagen: DLR)

realizó un vuelo en 2007. Su versión en serie lleva “en el aire” desde ese año y aún con la aprobación provisional de avión ultraligero. Está equipado con un motor externo cuya potencia de arranque es de 40 kW y una de funcionamiento de 30 kW. Los packs de baterías son opcionales y constan de 128 o 192 células LiPo de 10 Ah (4,75 kWh o 7,1 kWh con un peso correspondiente a cada batería de 42 kg o 55,6 kg). El Taurus sirvió como base a Pipistrel para su avión de competición en el “Green Flight Challenge”.

Carácter experimental

En los últimos años se ha desarrollado una creciente comunidad en torno a la aeronáutica eléctrica, abarcando desde parapentes motorizados a los helicópteros eléctricos de Sikorsky. Muchos proyectos giraban en torno a la “electrificación” de prototipos ya existentes, la mayoría en diseños amateur o ultraligeros. Por ejemplo, en 2009 un equipo de la universidad de Turín “electromotorizó” con 62 kW el avión ultraligero biplaza Alpi 300 y estableció un record de velocidad en aviones eléctricos con un vuelo de prueba a 250 km/h [14]. El avión fue capaz de volar durante menos de unos 15 minutos. El proyecto Sikorsky-Firefly [15] se sirve de un helicóptero de serie (Hughes/Schweizer 300) adaptado, que incorpora un motor eléctrico de

140 kW y baterías alemanas de ión-litio (GAIA), podrá volar en breve (y brevemente), e incluso Cessna demostrará próximamente que un C172 (el avión de cuatro plazas más fabricado del mundo) puede volar durante al menos algunos minutos eléctricamente [16].

También existen desarrollos comerciales especiales sobre aviones eléctricos: el biplaza Yuneec E430 [17] de China (**figura 3**) y el monoplaza Elektra One (**figura 4**) de la marca alemana PC-Aero [18]. Mientras que el Yuneec está diseñado como un planeador ligero a motor, el Elektra One depende de su escaso peso, reducidas dimensiones y buenas características aerodinámicas para ahorrar energía. Los datos más importantes se resumen en la **tabla 1**. Resulta curioso que Yuneec por ahora sólo ha fabricado un modelo de avión eléctrico, pero en tiradas muy grandes...

Green Flight Challenge

La NASA otorgó 1,65 millones de dólares como premio para el Green Flight Challenge (GFC), gestionado desde 2009 por el CAFE (Comparative Aircraft Flight Efficiency Foundation) para el diseño más efi-



Figura 3. El Yuneec E430 es el primer avión eléctrico chino desarrollado por un fabricante únicamente de modelos eléctricos. (Imagen: Yuneec)



Figura 4. El Elektra One de la compañía alemana PC-Aero participa en el Green Flight Challenge (GFC) y se oferta con el hangar solar incluido, de Solarworld, para cargar las baterías. (Imagen: PC-Aero, Copyright Shahn Sederberg)

cienta posible de avión [19]. Otro sponsor es Google, y el concurso tendrá lugar entre el 25 de septiembre y el 02 de octubre de 2011 en Santa Rosa, California.

El concurso no especifica ningún funcionamiento determinado. Sencillamente requiere poder recorrer una distancia de 200 millas (322 km) a una velocidad media de 100 millas/h (161 km/h) y con un rendimiento energético de 200 pasajeros-milla por galón de gasolina, o lo que es lo mismo, 33,7 kWh (equivalente en energía). Por pasajero se entiende todo ocupante a bordo. ¡En un avión biplaza esto significa que el consumo máximo ha de ser 2,36 l/100 km y en un monoplaza, 1,18 l/100 km!

Por lo tanto, en aviones eléctricos el límite está en 21 kWh (biplaza) o en 10,5 kWh (monoplaza) por cada 100 km.

Tabla 1. Comparativa del Elektra One y el Yuneec E430

	Elektra One	Yuneec 430
Número de plazas	1	2
Envergadura	8,6 m	13,8 m
Peso en vacío (sin batería)	100 kg	171,5 kg
Peso de la batería	Máx. 100 kg	83,5 kg
Peso en vacío (con batería)	Máx. 200 kg	255 kg
Carga útil	100 kg	175 kg
Peso máximo	300 kg	430 kg
Potencia máxima del motor	16 kW (22 PS)	40 kW (54 PS)
Tipo de batería	LiPo	LiPo
Capacidad	n.a.	100 Ah
Tensión de la batería	n.a.	133,2 V
Velocidad de crucero	160 km/h	95 km/h*
Duración del vuelo	> 3 h	Aprox. 2 h*
Distancia	Máx. 500 km	Aprox. 190 km*
* Valores provisionales		

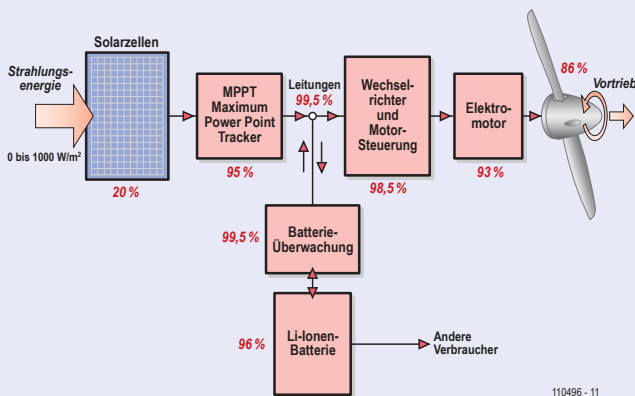
Tabla 2. Equipos y aviones clasificados para el Green Flight Challenge

Nº	Equipo	Avión (prototipo participante)	Número de plazas	Envergadura	Potencia máxima	Funcionamiento (motor participante)
1	Einar Enevoldson PC Aero (Alemania)	Elektra 1	1	8,6 m (28,2 ft)	16 kW (21 PS)	Eléctrico
2	Gene Sheehan Feuling GFC (EEUU)	Team Feuling GFC	1	5,1 m (16,7 ft)	16 kW (21 PS)	Eléctrico
3	Gregory Cole Windward Performance (EEUU)	Goshawk	2	15,5 m (51,0 ft)	n.a.	Eléctrico
4	Lawrence Speer Green-Elis (Francia)	Greenelis PXL	2	10,8 m (35,5 ft)	30 kW (41 PS)	Biodiesel (Smart-Diesel)
5	Mike Stude Michael Stude (EEUU)	Wings of Salvation	1	5,1 m (16,7 ft)	32 kW (44 PS)	Etanol
6	Richard Anderson Embry-Riddle Aeronautical University/Stemme (Alemania)	EcoEagle (Stemme S10)	2	23 m (75,0 ft)	100 kW (136 PS)	Biogasolina-híbrido (Rotax 914F)
7	John W. McGinnis Synergy (EEUU)	Synergy	6	9,8 m (32,0 ft)	142 kW (193 PS)	Biodiesel
8	Greg Stevenson GSE-Aerochia (EEUU)	Econo-Cruiser 3000	2	14,7 m (48,3 ft)	15 kW (20 PS)	Biogasolina-híbrido
9	Ira Munn IKE Aerospace (EEUU)	SERAPH	1	4,6 m (15,0 ft)	30 kW (41 PS)	Biogasolina-híbrido
10	Eric Raymond e-Genius/Universität Stuttgart (Alemania)	e-Genius	2	16,9 m (55,4 ft)	60 kW (82 PS)	Eléctrico
11	Jim Lee Phoenix Air (República Checa)	PhoEnix (Phoenix)	2	14,4 m (47,3 ft)	44 kW (60 PS)	Eléctrico
12	Scott Sanford Yuneec (China)*	Yuneec E 1000	3	17,0 m (56,0 ft)	120 kW (163 PS)	Eléctrico
13	Jack Langelaan Penn State University/Pipistrel (Eslovenia)	Taurus G4 (Taurus)	4	21,0 m (69,1 ft)	145 kW (197 PS)	Eléctrico

* Clasificación parcial, véase el texto

Aviones solares

La historia de los vuelos solares [29] comienza con los aviones no tripulados. El Sunrise I de Ray Boucher, con 10 kg de peso fue el primero en hacerlo, en 1974 en California. El siguiente modelo ya fue capaz en 1975 de alcanzar una altura de 5000 m. En Europa fue Fred Militky el que introdujo en 1976 el primer avión solar a control remoto. Los potentes aviones Gossamer Condor y Albatros del legendario Paul MacCready sentaron las bases de los vuelos solares tripulados. Una versión reducida del Albatros, el Gossamer Penguin, resultó ser en 1980 el primer avión solar con paneles y motor eléctrico tripulado por una mujer, en el cual la piloto Janice Brown recorrió tres kilómetros en un vuelo de 14 minutos. El siguiente fue el MacCready's Solar Challenger, que con una envergadura de sólo 14 m voló en julio de 1981 de París a Londres.



En Alemania, el profesor de diseño Günther Rochelt fue capaz de realizar un vuelo de casi seis horas en 1983 con su planeador ultraligero eléctrico solar, el Solair 1 –aunque con cierta ayuda térmica–. Con un diseño similar de planeador solar, el Sunseeker, Eric Raymond atravesó los EEUU en 1990 en dos semanas. Ambos aviones contaban con una potencia de arranque de sólo 2,2 kW, de menor magnitud que el Icaré 2 de la Universidad de Stuttgart, que con 14 kW ganó en 1996 la competición Berblinger. La potencia de las células solares hizo posible que el avión recorriera una distancia de casi 500 km sin ayuda de motores térmicos.

De un orden de magnitud superior, el proyecto Solar Impulse [30] de los suizos Bertrand Piccard y André Borschberg es por ahora el

más ambicioso. Con él será posible dar la vuelta al mundo en un avión sin combustible [31]. La imagen muestra el principal problema al que se enfrentan, con una representación básica del funcionamiento solar.

Del rendimiento podemos deducir que de la energía solar recibida durante el día, (en verano aproximadamente unos 500 W/m²), sólo cerca del 13 % llega al propulsor. Esto se suma a la baja eficiencia de las células solares, cercana al 20 %. Ya que hay que almacenar energía para la noche, esto hace que no pueda funcionar durante 24 horas. El resultado es un avión con una gran superficie solar, que ha de ser extremadamente ligero, a pesar de tener una escasa potencia. Si contamos con un funcionamiento de 24 horas, con una potencia solar de 250 W/m², dispondremos de una potencia en el motor de aproxi-



madamente el 12 % = 30 W por metro cuadrado de células solares, teniendo en cuenta las pérdidas en la batería. La superficie instalada entre las alas y el fuselaje del Solar Impulse suma un total de 200 m², con lo que puede alcanzarse una potencia de 6 kW (8,2 PS), capaz de mantener un avión con un peso de 1,6 toneladas en el aire, durante día y noche y a unos 70 km/h (en comparación, el primer avión de los hermanos Wright en el año 1903 tenía 12 PS).

Este concepto es posible, como ha demostrado el primer prototipo del Solar Impulse (con su primer vuelo en 2009), batiendo récords (9000 m de altura y un tiempo de vuelo de 26 h). (Imagen DLR: Solar Impulse/EPFL Claudio Leonardi)

Para participar se han clasificado 13 equipos (tabla 2), cuyos aviones abarcan todos los modos de funcionamiento y distribuciones posibles: de 1 a 6 plazas, de 5 a 23 m de envergadura, potencias de 15 a 145 kW (20 a 197 PS) basándose en motores de gasolina, (bio) diesel, etanol, eléctricos e híbridos. Los que peor lo tienen son los motores con biocombustibles (biodiesel/etanol), y por ello, debido al “precio del biofuel” [20] se les ha permitido requerimientos más reducidos.

Para participar en el GFC hace falta un líder de equipo americano, aunque seis de los equipos participantes provienen de Europa. El único avión chino (Yuneec E1000) abandonó el concurso debido a un accidente en las pruebas. Mientras que los aviones europeos como el Elektra 1 y el Greenelis parten de planeadores a motor, algunos de los diseños estadounidenses son

realmente futuristas, como el “Synergy” (figura 5) y el “Seraph” (figura 6). Pero un planeador a motor (figura 7) también puede parecer exótico...

Los que más oportunidades tienen son los aviones eléctricos de dos o más plazas. Por ejemplo, el Instituto Aeronáutico de la Universidad de Stuttgart [21], bajo la tutela del profesor Rudolf Voit-Nitschmann, desarrolló y construyó el biplaza e-Genius (figura 8 y tabla 3), sin tener ningún problema para cumplir los requisitos del GFC. En un vuelo de prueba en Junio de 2011 fue capaz de recorrer una distancia de 341 km con una velocidad media de 164 km/h y un consumo de 46 kWh (13,5 kWh/100 km), lo cual equivale en gasolina a 1,5 l/100 km, o lo que es lo mismo, 310 *passenger miles per gallon* (PMPG), como dicen los americanos. Han sobrepasado los requisitos del GFC en un 55 %. ¡Nuestra más sincera enhorabuena!



Figura 5. El Synergy, de seis plazas, participa en el GFC con un potente motor biodiesel de 142 kW (193 PS).
(Imagen: CAFE Foundation Blog)



Figura 6. El monoplaza Seraph con un motor híbrido de biodiesel, está dentro de los diseños poco convencionales del GFC.
(Imagen: CAFE Foundation Blog)

El futuro de los “E-vuelos”

Básicamente, con la movilidad eléctrica en el aire pasa lo mismo que con la de tierra firme: ¡la batería es lo que cuenta! En cuanto al funcionamiento –la electrónica y el motor– ya se han alcanzado rendi-

mientos de hasta el 90 %. El estado actual de las baterías basta para planeadores con y sin motor capaces de despegar por sí solos [22]. Los desarrollos como el e-Genius y el Elektra 1 pronto podrían dar origen a un mercado para planeadores a motor y aviones ultralige-

Enlaces:

- [1] http://rbmn.free.fr/Ballon_photos_10.html
- [2] www.poweringnow.com/technology/history
- [3] www.modellflugsport.ch/upload/museum/geschichte/modelle/Silentius.pdf
- [4] www.graupner.de/de/products/a1899b43-43cd-4cea-8e22-e05d381420e6/1725/product.aspx
- [5] www.flightglobal.com/pdfarchive/view/1973/1973%20-%202921.html
- [6] www.airventure.de/historypics/Emose1.jpg
- [7] www.airenergy.de/index.html
- [8] www.lange-aviation.com
- [9] www.dlr.de/tt/desktopdefault.aspx/tabid-4935/8219_read-13587
- [10] http://ecweb.redcor.ch/fachtag-energie/referate/referate/ws2_2_andreavezzini_light.pdf
- [11] www.saftbatteries.com/Produit_Large_VLM_cell_range_301_62/Language/en-US/Default.aspx
- [12] www.schempp-hirth.com/index.php?id=nimbus-4dm1
- [13] www.pipistrel.si/plane/taurus-electro/overview
- [14] www.skyspark.eu/web/eng/index.php
- [15] <http://blog.cafefoundation.org/?p=1438>
- [16] <http://blog.cafefoundation.org/?p=1422>
- [17] <http://yuneec.com>
- [18] <http://www.pc-aero.de>
- [19] http://cafefoundation.org/v2/gfc_main.php
- [20] http://cafefoundation.org/v2/pdf_GFC/2011_06_30_GFC_Prize_Structure.pdf
- [21] <http://www.ifb.uni-stuttgart.de/index.php/forschung/flugzeugentwurf/hydrogenius>
- [22] <http://yuneec.com/paramotor.html>
- [23] www.flightdesign.com/index.php?page=presentation
- [24] www.elektor.de/elektronik-news/hybrid-elektrisches-flugzeug.1871506.lynx
- [25] <http://www.pipistrel.si/plane/panthera/overview>
- [26] www.eads.com/dms/eads/int/en/press/documents/Dossiers/Downloads/EADS-Brochure_VoltAir_English.pdf
- [27] www.dlr.de/tt/desktopdefault.aspx/tabid-4935/8219_read-13587
- [28] www.dlr.de/desktopdefault.aspx/tabid-6216/10226_read-26189
- [29] www.solarflugzeuge.de (UK: www.asl.ethz.ch/research/asl/skysailor/History_of_Solar_Flight.pdf)
- [30] www.solarimpulse.com
- [31] www.mp.haw-hamburg.de/pers/Scholz/ewade/2009/EWADE2009_Ross_Paper.pdf



Figura 7. Pipistrel de Eslovenia ha combinado dos motores de planeador Taurus con un motor eléctrico de 145 kW para lograr un avión de doble propulsión en el GFC. En el concurso, esto tiene sentido a la hora de evaluar el consumo por pasajero. (Imagen: Pipistrel)

ros. En unión con los módulos solares de los aviones de hangar (PC-Aero/SolarWorld) o de transporte (Taurus-G2/Pipistrel), así como los que incluyen un generador eólico (Arcus-E/Windreich) para cargar las baterías, el sueño de poder realizar vuelos libres de emisiones, con energías renovables, comienza a ser una realidad.

En los motores de avión también son aptas las soluciones híbridas, como ya demostraron Flight Design [23] y EADS/Siemens/Diamond-Aircraft [24].

El verdadero progreso ya sólo puede conducir a mejoras en la tecnología de las baterías. El fabricante esloveno Pipistrel tiene fe en esto, y está actualmente desarrollando su avión de pasajeros Panthera [25] que junto a la híbrida-eléctrica también ha planificado otra versión (¿o visión?) puramente eléctrica. El motor eléctrico de 145 kW refrigerado por agua ya ha sido puesto a prueba para el GFC en el Taurus-G4 (figura 7).



Figura 8. El anteriormente conocido como “Hydrogenius”, actualmente e-Genius, que utiliza una pila de combustible, se trata del avión eléctrico de la Universidad de Stuttgart presentado en el GFC. (Imagen: e-Genius-Team, IFB Universität Stuttgart)

Tabla 3. Datos técnicos del e-Genius	
Número de plazas	2
Envergadura	16,9 m
Peso en vacío (sin batería)	171,5 kg
Peso de la batería	83,5 kg
Peso en vacío (con batería)	255 kg
Carga útil	180 kg
Peso máximo	850 kg
Potencia media/pico del motor	60/100 kW (82/136 PS)
Tipo de motor	Motor síncrono de imanes permanentes, refrigerado por agua
Peso del motor/diámetro/longitud	27 kg/25 cm/28 cm
Diámetro del propulsor	2,2 m
Batería	ión-litio/56 kWh
Peso total del sistema (motor, electrónica y batería)	336 kg
Rendimiento total del motor y el inversor	> 90 %
Velocidad de crucero	De 140 a 235 km/h
Potencia máxima de elevación con 850 kg	4,5 m/s
Duración del vuelo	Max. ca. 4 h
Distancia	> 400 km

La empresa que concibió el Airbus, EADS, evalúa actualmente varios conceptos para los aviones eléctricos del futuro. El conceptual VoltAir [26], publicado en Mayo, utiliza baterías de litio-aire con 1000 Wh/kg y materiales superconductores para los motores eléctricos y cableados con refrigeración por nitrógeno (figura 9). Con sus proyectados 7-8 kW/kg, estos motores superan incluso las potencias de los turbopropulsores actuales. Las mejora significativas en la eficiencia, así como en la aerodinámica, son un sueño para las aerolíneas de aquí a 25 años: económicas, silenciosas, libres de emisiones y sobre todo muy cómodas, sobre todo considerando la pesadilla ambiental que suponen los vuelos hoy en día.

(110496)

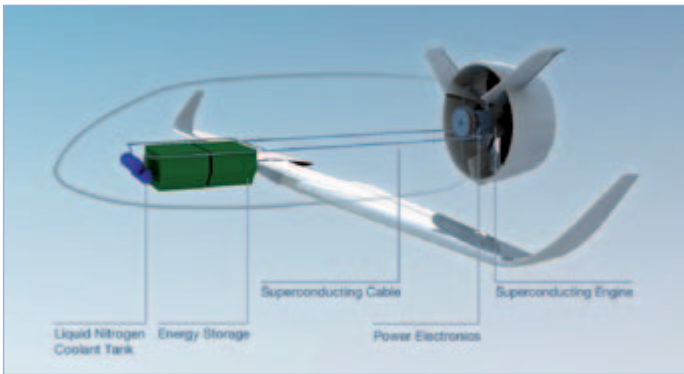
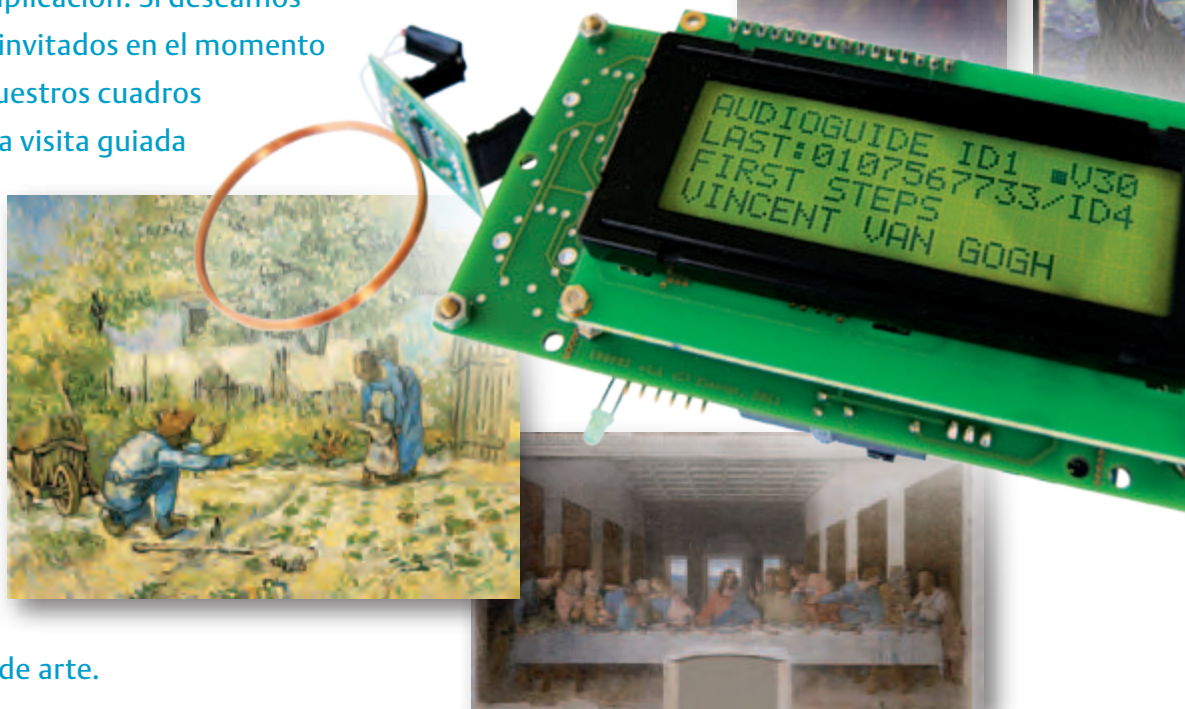


Figura 9. El diseño conceptual de EADS VoltAir en cuanto a aeronáutica eléctrica utiliza materiales superconductores y refrigeración por nitrógeno para los cableados y el motor. (Imagen: EADS)

Audioguía

Primeros pasos con Platino

Hablar para no decir nada, es exactamente una situación que vamos a prohibir en esta nueva aplicación. Si deseamos impresionar a nuestros invitados en el momento de la presentación de nuestros cuadros más bellos, facilitando la visita guiada a una exposición, ésta es la aplicación que necesitamos. Bastará con acercar este montaje al cuadro para que nuestros atentos invitados escuchen automáticamente los comentarios que describen nuestra obra de arte.



Clemens Valens y Grégory Ester (Francia)

Este montaje hace uso de dos placas: Platino [3], el cerebro del montaje, y rMP3 [2] que permite reproducir ficheros de audio almacenados en una tarjeta SD, SDHC o MMC de una capacidad comprendida entre 8 MB y 32 GB. En la **Figura 1** se muestra el diagrama de bloques del montaje.

Platino y rMP3 se comunican entre sí gracias a un enlace serie TTL, con lo que, en lo sucesivo, nos es posible añadir sonido a un acontecimiento. El acontecimiento del que hablamos aquí se activa acercando al objeto a identificar, la antena [6] conectada al lector RFID [4] de la Audioguía. El cuadro que hay que identificar (o el extremo situado en las proximidades) contiene el transceptor [5] y su identificador único se transmite, sin hilos, al lector RFID, el programa compara el identificador con los valores guardados previamente en la base de datos, y activa automáticamente la lectura del fichero que contiene las explicaciones de la obra que hay que comentar

Leer un fichero mp3

La placa rMP3 es una tarjeta de extensión compatible con Arduino, es decir, un escudo («shield»), que permite reproducir grabaciones MP3 previamente almacenados en una tarjeta de memoria. Este módulo de reproducción sonora acepta comandos que le son enviados gracias a un enlace serie asincrónico. Así, es posible leer un fichero MP3, proseguir con el siguiente, hacer un avance rápido, parar la lectura, modificar el volumen sonoro... Hasta podemos incluso utilizar la tarjeta de memoria como unidad de almacenamiento. Las funciones son numerosas y se aconseja, en la medida de lo posible, consultar la documentación en línea disponible en la página web del constructor [7].

Un conector tipo «jack» hembra, de 3 mm, sirve para conectar un casco o unos altavoces estéreo de 16 Ω.

El módulo rMP3 y Platino se acoplan fácilmente gracias a los conectores compatibles Arduino, disponibles en las dos placas pero,

antes de hacerlo, es necesario hacer una pequeña manipulación. Es el puerto serie físico USART0 del ATmega1284P de Platino, el que se utiliza para controlar el módulo musical, por lo que hay que desviar las líneas RXD0 y TXD0 hacia TX y RX del rMP3. La **figura2** muestra cuales son los dos terminales que hay que cortar: el hilo conductor naranja unirá, a partir de ahora, el terminal 0/R (RXD0) y el terminal 6 (TX-rMP3), y el hilo conductor amarillo establece la unión entre 1/T (TXD0) y el terminal 7 (RX-rMP3). Ahora formateamos la tarjeta SD con FAT32 (si fuese necesario). Creamos un directorio llamado «audioguide» en el raíz de la tarjeta de almacenamiento SD. Elegimos una canción mp3 que nos guste y la almacenamos en nuestra tarjeta SD, en el directorio audioguide. Renombramos el fichero copiado con el nombre de «music.mp3» e insertamos la tarjeta SD en el conector adecuado de la placa rMP3.

Ha llegado el momento tan esperado: después de haber colocado un visualizador LCD



de 4 x 20 sobre el conector K9 de Platino, vamos a poder acoplar la placa rMP3 a Platino.

Cargaremos en la memoria flash del microcontrolador el microprograma («firmware») «110544-I_audio-guide_test_rmp3_lcd.hex» [1].

La lectura del fichero musical se inicia y se muestra la pantalla visible en la **Figura 3**. La primera línea corresponde a la versión del firmware, seguido por su número de serie. La segunda línea nos señala que el volumen de la vía izquierda es idéntico al de la vía derecha, es decir, 16, lo que corresponde a $-0,5 \times 16 = -8$ dB. Sobre la cuarta línea tenemos la lectura del título de la canción que se inició hace 2 s, la frecuencia de muestreo es igual a 44,1 kHz y la velocidad binaria es de 128 kbit/s. El parámetro «J» indica que la codificación del fichero mp3 es mixta (estéreo/mono).

En esta fase hemos validado el buen funcionamiento del módulo rMP3.

Identificación por RFID

El lector RFID ya ha sido objeto de un artículo [8]. Está constituido de una placa basada en un CI EM4095, acompañado por su antena. El EM4095 ha sido diseñado para una gama de frecuencias comprendida entre 100 kHz y 150 kHz. Así pues, ha sido la familia de los transceptores [5] RFID pasivos que trabajan en la frecuencia 125 kHz la que ha sido elegida.

El alcance máximo de lectura es de unos 12 cm para una antena de 53 mm [6] de diámetro, que se reduce a un alcance de 6 cm para un diámetro de antena de 25 mm.

La escritura de la función RFID se hará sin rechazo gracias a las cinco conexiones provenientes de la pequeña placa y etiquetadas como +5V, SHD, DEMOD OUT, MOD, GND; que se conectarán, respectivamente, con las líneas +5V, 8, 10, 9, GND de los terminales hembra del módulo rMP3, ya insertado sobre Platino.

Configurar Platino

Con el fin de facilitar el interfaz Hombre/Audioguía, éste último estará equipado con

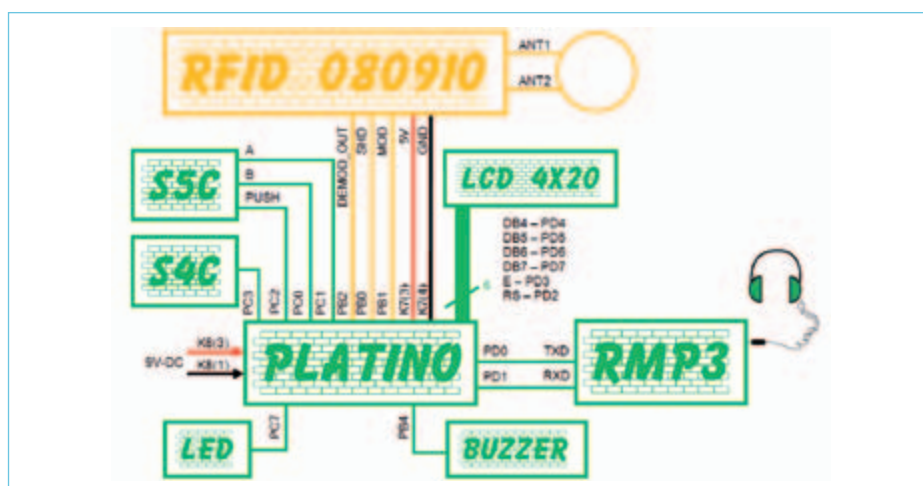


Figura 1. El diagrama de bloques de la Audioguía.

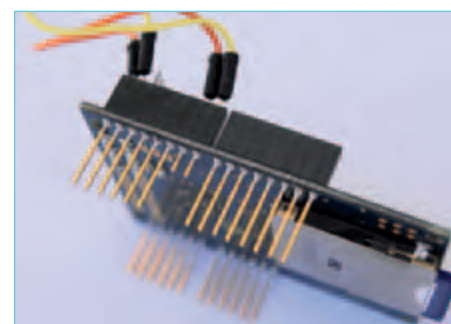


Figura 2. Platino y rMP3: un acoplamiento conseguido.

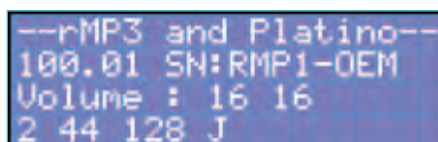


Figura 3. Informaciones provenientes del módulo rMP3.

un pulsador y un codificador rotativo. En la cara de cobre de la placa Platino, a la derecha, el pulsador será soldado en el emplazamiento S4C y el codificador en posición S5C. Las **Figuras 4 y 5** permiten visualizar la posición de los puentes de soldadura de Platino. S4C estará conectado a PC3, gracias a JP7, una pequeña gota de estaño sobre JP4 y JP5 bastará para establecer el enlace entre los terminales B y A de S5C con PC0 y PC1 del microcontrolador. El sentido horario o antihorario con el que podemos girar el codificador no son las dos únicas informaciones podemos obtener. También es posible pulsar sobre el codificador para validar una elección. Por ejemplo, el contacto de cierre correspondiente comunicará su estado sobre PC2 gracias al puente de soldadura JP6.

Sólo se necesita un LED verde, JP14 valida su uso sobre PC7 y, por tanto, habrá que sol-

dar una resistencia de 470 Ω en el emplazamiento de R8 con el fin de limitar la intensidad de la corriente en el LED.

La retroiluminación de la pantalla LCD no se utiliza. Necesitaremos un zumbador piezoeléctrico en BUZ1, que será conectado a PB4 gracias al puente de soldadura JP1.

El microcontrolador utilizado es un ATmega1284P DIL40, que es programable en el propio circuito (ISP de *In System Programming*) gracias a una conexión serie síncrona. El conector K3 (HE10 de seis terminales) permite la conexión del programador [9]. Las conexiones equipotenciales entre las líneas MOSI, MISO y SCK del conector HE10, PB5, PB6 y PB7 del microcontrolador son posibles gracias a JP13, JP12 y JP11.

La tensión necesaria para el funcionamiento del conjunto se aplica en K8 (1), para GND y en K8 (3), para la tensión positiva. Es el puente de soldadura JP14 el que asegura la alimentación bajo una tensión de +5V del módulo rMP3.

El funcionamiento del conjunto

Cada obra que identifiquemos se asocia con un transceptor único. Es cuando acercamos la Audioguía al transceptor cuando se activa la reproducción del comentario grabado,

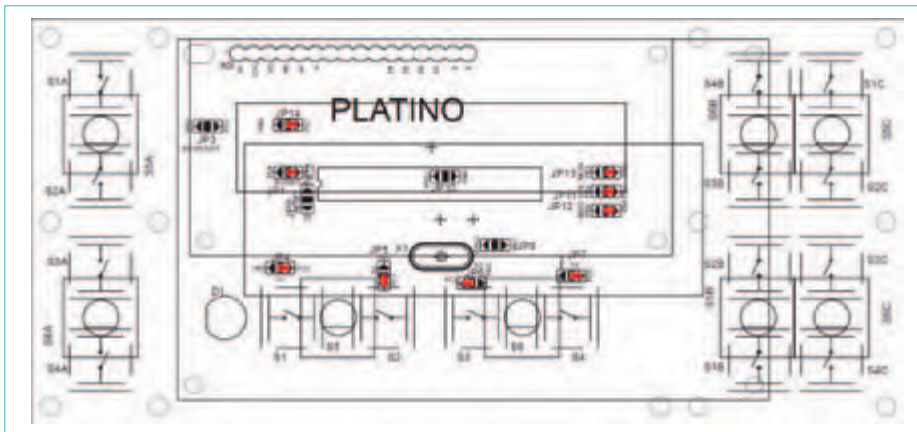


Figura 4. ¡En la parte superior ("TOP") los puentes de soldadura!



Figura 5. Cara "BOTTOM": un solo puente de soldadura.

que podemos oír en unos cascos o unos altavoces. El LED verde se enciende y el zumbador se deja oír en cuanto se detecta una tarjeta RFID. El programa «110544-I_audio-guide_firmware_v1.2.bas» está asociado a un fichero ASCII llamado «tags.txt», el cual contiene las informaciones de identificación de cada obra, que serán mostradas sobre la pantalla LCD (ver **Listado 1**).

En el momento de la compilación, estas líneas de programa son tenidas en cuenta gracias a la instrucción `$include tags.txt`. Las explicaciones vocales son almacenadas en la tarjeta SD, el nombre de los ficheros mp3 registrados debe respetar la sintaxis `IDx.mp3`, donde «x» debe ser idéntico al índice utilizado en el fichero de texto «tags.txt». Por ejemplo, el comentario correspondiente a la obra nombrada como «Los primeros pasos de Vincent Van Gogh» será registrado, sin tener que respetar la caja, bajo el nombre de `ID4.mp3`.

El volumen se ajusta de 0 (min) a 100 (max) en incrementos de 5, gracias al codificador rotativo. Además de las dos rotaciones en el sentido horario (C.W.: *Clock Wise*) o antiho-

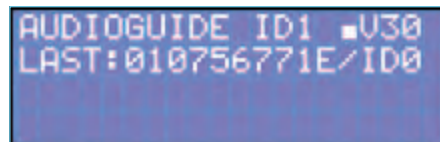


Figura 6. ID=0, el transceptor no ha sido registrado aún.



Figura 7. Grabador PaMus MP3: un magnetófono digital muy práctico.

rario (C.C.W.: *Counter Clock Wise*), también es posible cerrar un tercer contacto asociado con el mismo codificador. En lugar de girar el botón a la derecha o a la izquierda, pulsaremos hacia arriba, lo que permite de

forma alternativa, detenerse o reproducir un comentario manualmente. El fichero reproducido será el correspondiente al ID mostrado sobre la primera línea. Es el cierre de S4C el que permite mostrar sucesivamente sobre la tercera y la cuarta línea, el nombre de la obra y del artista asociados a cada ID. Así, es posible activar la lectura de los ficheros mp3 sin necesidad de acercarnos a las tarjetas RFID. Si se nos escapa algún detalle en la explicación de la obra, podemos reproducir todo el comentario en forma de bucle.

El transceptor registrado en la primera posición de la lista (RFID(1)) reproducirá el fichero musical `ID1.mp3`. Esto nos permitirá disponer de tiempo para ajustar el volumen sonoro a nuestro gusto, gracias al codificador rotativo y escuchando nuestra música preferida. Después de haber copiado los ficheros mp3 con los ID del 1 al 6 en la tarjeta SD y cuando hayamos cargado el firmware «110544-I_audioguide_firmware_v1.2.hex» por primera vez, sólo funciona el modo manual. Ningún transceptor será identificado (ID=0) con una obra, pero su

Listado 1.

«tags.txt» contiene las informaciones a mostrar en la pantalla LCD. El nombre de los títulos (Title(x)) y de los artistas (Artist(x)) no debe sobrepasar los 20 caracteres.

```
const maxi = 6
Dim Rfid(maxi) As String * 10
Dim Title(maxi) As String * 20
Dim Artist(maxi) As String * 20

Rfid(1) = «0107567708»
Title(1) = «VOLUME SETTING»
Artist(1) = «MARCUS MILLER/POWER»
Rfid(2) = «0107567790»
```

```
Title(2) = «LA GIOCONDA»
Artist(2) = «LEONARDO DA VINCI»
Rfid(3) = «01075677B4»
Title(3) = «STARRY NIGHT»
Artist(3) = «VINCENT VAN GOGH»
Rfid(4) = «01075677AE»
Title(4) = «FIRST STEPS»
Artist(4) = «VINCENT VAN GOGH»
Rfid(5) = «010756785E»
Title(5) = «FRONT COVER»
Artist(5) = «ELEKTOR 1978»
Rfid(6) = «01075676E6»
Title(6) = «THE LAST SUPPER»
Artist(6) = «LEONARD DE VINCI»
```

número único, compuesto por diez caracteres hexadecimales, se mostrará, a pesar de todo, sobre la segunda línea (ver **Figura 6**). Lo primero a hacer es referenciar los transceptores numerándolos.

Luego, los acercaremos a la antena, uno por uno, y volveremos a copiar los números en el fichero «tags.txt» a medida que vayamos obteniéndolos. Previamente, habremos abierto el fichero «tags.txt» con la ayuda de un editor de texto como el «Bloc de Notas» (o «Notepad»).

Por último, grabaremos el conjunto de las modificaciones realizadas. Recopilamos el programa «110544-I_audioguide_firmware_v1.2.bas» y nuestra Audioguía estará lista. Verificaremos que tenemos las informaciones correctas y que son las que se muestran en pantalla y se escuchan al pasar por delante de las seis tarjetas.

De este modo podemos registrar hasta un total de noventa y nueve transceptores, donde la constante (`const maxi = 6`) debe ser igual al número total de tarjetas, seis en el caso actual.

Grabador PaMus MP3

El Grabador PaMus MP3 [10] es un programa que permite registrar su voz en formato MP3 con la ayuda de un micrófono. Fácil de instalar y de utilizar, la aplicación funciona como un magnetófono (ver **Figura 7**). Podemos ajustar la calidad sonora de la grabación (hay tres niveles disponibles). La generación de los ficheros es automática y es posible renombrarlos o borrarlos directamente desde el interfaz del programa. Muy práctico y gratuito, sería una pena privarnos de él.

(110544)

Enlaces en Internet

- [1] www.elektor.es/110544
- [2] www.lextronic.fr/P18805-module-de-restitution-rmp3-pour-arduino.html
- [3] www.elektor.es/100892
- [4] www.elektor.es/products/kits-modules/modules/080910-91-rfid-savvy.925192.lynx
- [5] www.lextronic.fr/P1503-carte-transpondeur-rfid-card1.html
- [6] www.lextronic.fr/P1478-antenne-rfid-125-khz-ant-rfid2.html
- [7] www.roguerobotics.com/wikidocs/rmp3/start
- [8] www.elektor.es/080910
- [9] www.elektor.es/080083
- [10] <http://papiermusique.fr/dossier03.php>

Publicidad

Todo un laboratorio de electrónica en un solo dispositivo USB



Con **NI myDAQ**, tu ordenador se convierte en el instrumento. Al instalar el controlador de hardware **NI ELVISmx** dispondrás de: Multímetro Digital, Osciloscopio, Generador de Funciones, Analizador Bode, Analizador de Señales Dinámicas, Generador de Forma de Onda Arbitraria, Generador de Salidas Digitales y Lector de Señales Digitales.

Ocho instrumentos de ingeniería para trabajar como y donde tu quieras.

¡NI myDAQ por solo 149 Euros!

(incluyendo licencias de estudiante de LabVIEW y Multisim)

Contenido del paquete:

Dispositivo NI myDAQ, DVD de instalación que contiene NI ELVISmx y NI DAQmx, así como las versiones para estudiante de LabVIEW y Multisim con sus licencias correspondientes, Cable USB, Puntas de prueba para el multímetro, Cable de audio (3,5 mm), Conector de tornillos con destornillador NI, Tarjeta informativa de iniciación, Tarjeta informativa de programas de ejemplo de TI, Bandeja de almacenamiento reutilizable.



Más información y pedidos en: www.elektor.es/myDAQ

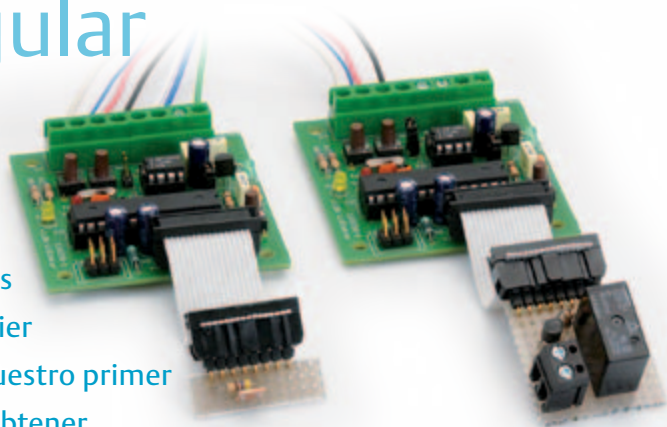
¡Atención! Para poder adquirir el NI myDAQ deberás acreditar tu condición de estudiante.



¡Que viene el bus! (8)

Medir, controlar, regular

En la última entrega presentamos un sencillo protocolo para el procesamiento de los valores de medida; ahora lo utilizaremos para un pequeño sistema de control. En los nodos experimentales sólo se conectan unos cuantos componentes como fotoresistencias y relés, que cualquier electrónico tiene a mano sin problema. Sin embargo, nuestro primer nodo sensor no tiene nada que envidiar, ya que puede obtener valores de distintas unidades y establecer límites de forma automática.



Jens Nickel

Los lectores incondicionales de esta sección del Bus ya lo saben de sobra: comenzamos cada sección con un pequeño recordatorio de la última "lección". En la edición de septiembre presentamos un sencillo protocolo que mostraba cómo utilizar los ocho bytes de datos útiles contenidos en mensajes de 16 bytes. Con esto nos aseguramos que los distintos nodos (sensores, actuadores controladores) también interactúen entre sí.

Con ocho bytes un dispositivo del bus puede transmitir hasta cuatro valores de medida a la vez. Pensemos por ejemplo en un nodo con cuatro sensores de temperatura conectados. Del mismo modo, cuatro actuadores pueden controlarse enviando una sola orden.

Para saber qué bytes dentro de los datos útiles pertenecen a cada sub-unidad dentro de un nodo, basta con evaluar su posición; así, los bytes 6 y 7 pertenecen al *Channel0*, los bytes 8 y 9 a *Channel1* y así en adelante (ver la **figura 1**). Mediante cada uno de estos canales podemos transmitir un valor de 10 bits con signo, la **figura 2** muestra otra vez cómo están divididos los datos. El bit *xH.5* nos dice si es un valor actual o deseado

(1 = deseado, o sea *SetValue*; 0 = actual, o sea *CurrentValue*). El bit *xH.4* se utiliza para distinguir un *AcknowledgeMessage* (= 1) del mensaje original.

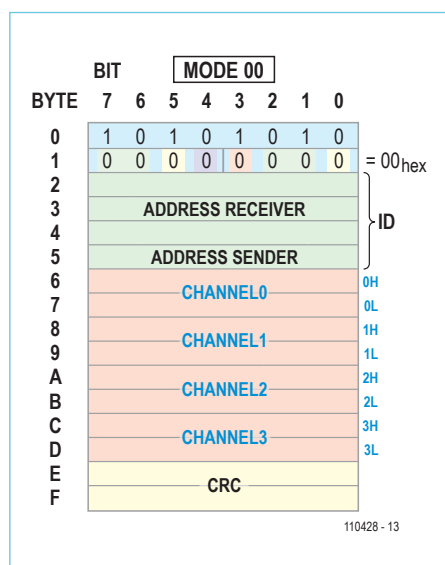


Figura 1. En el 2ByteMode nuestros ocho bytes útiles están separados en cuatro pares. Así, pueden transferirse los valores de cuatro canales (sensores/actuadores en los nodos) uno tras otro.

Ya hemos comentado que añadiremos otro modo que utiliza cuatro bytes por canal, con lo que pueden transmitirse valores más exactos así como comandos especiales (*4ByteMode*). Para distinguirlo del *2ByteMode* siempre se utiliza el bit 6 del primer byte (1 = 2 bytes por canal; 0 = 4 bytes por canal). En adelante también hemos descrito los bits que cambian, lo cual lógicamente no es algo para todo el mundo. Para acelerar la implementación del protocolo, en la **tabla 1** se han resumido los valores decimales más importantes.

El primer sensor

Para poder empezar a medir algo real, nos servimos del sencillo circuito de la **figura 3**. K1 se conecta 1:1 con el puerto de expansión K4 en nuestro nodo experimental [1]. Hemos montado nuestro circuito sobre una pequeña tarjeta perforada y diseñado el correspondiente cable plano (para cada uno de los conectores de 2x8 nombrados sólo se utiliza una fila). Podemos utilizar cables de este tipo en más ocasiones.

Quien haya programado en los nodos el firmware de la anterior entrega, puede probarlos inmediatamente tras terminar el montaje del hardware. A los nodos con

Productos de Elektor y soporte

- Nodos experimentales (tarjeta 110258-1 o tercer set de tarjetas 110258-1C3)
- Conversor USB/RS485 (debidamente montado y probado, 110258-91)

- Descarga gratuita del software (firmware del controlador más software de PC)

Todos los productos y descargas están disponibles en la página de este artículo: www.elektor.es/110428

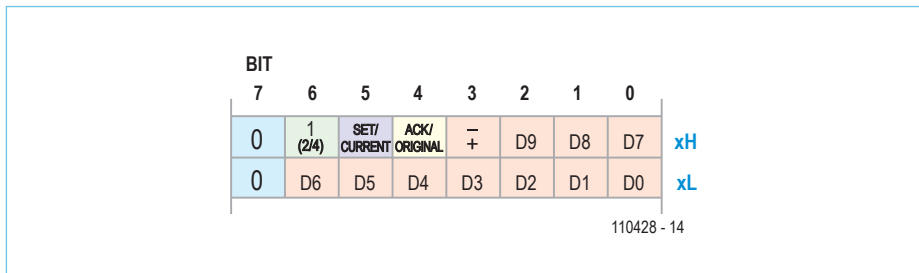
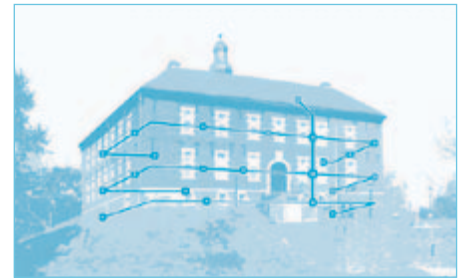


Figura 2. El 2ByteMode al detalla: una resolución de 10 bits (más el signo) es suficiente para la mayoría de aplicaciones.

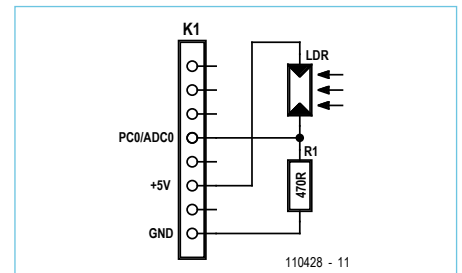


Figura 3. Nuestro primer sensor. K1 está conectado 1:1 con K4 en nuestro nodo experimental.

fotoresistencia ha de asignárseles la dirección de bus 02 y un *Devicemode* de 01, lo cual se hace sencillamente escribiendo los bytes correspondientes en las direcciones 002 y 006 de la EEPROM. En la 004 grabamos un 01 (*Code-Variable Scheduled*), de

modo que el nodo sepa que será preguntado periódicamente por el planificador (software de PC). Tanto el software de PC como el firmware de la última entrega están disponibles en la página del proyecto [2].

Tras el arranque del planificador en el software del PC, se transfiere el valor correspondiente del ADC en intervalos regulares desde el nodo 2 al PC (a *Channel0*, gracias a los primeros dos bytes útiles). El valor ADC indicado en el cuadro de texto sirve como

Tabla 1: guía rápida de valores correctos				
Transmite valores de -1023 a +1023 SIGN = 8, si es negativo, de lo contrario = 0 LOW = 7 bits inferiores (BASCOM: Low = Wert And 127) HIGH = 3 bits superiores (BASCOM: Shift Wert, Right, 7 : High = Wert)				
	Byte 1	Byte 2		
Lectura de valores de medida	64 + SIGN + HIGH	LOW		
Selección de valores fijados	96 + SIGN + HIGH	LOW		
Activar	96	1		
Desactivar	96	0		
Notificación del receptor: cada vez el byte 1 más 16				
Tamaño, unidad y escalado CH = número de canal POT = potencia de diez de la unidad PSIGN = cuando la potencia de diez sea negativa, de lo contrario = 0				
	Byte 1	Byte 2	Byte 3	Byte 4
Fijar	40 + CH	193	Ver tabla 2	PSIGN + POT
Tensión en V	40 + CH	193	16	0
Tensión en mV	40 + CH	193	16	19
Corriente en mA	40 + CH	193	17	19
Comunicar el tamaño/unidad fijados por el sensor: byte 1 = 8 + CH				
Valor límite y alarma CH = número de canal				
	Byte 1	Byte 2		
Fijar el límite inferior	104 + CH	209		
Fijar el límite superior	104 + CH	210		
¡Alarma! Valor límite no alcanzado	72 + CH	209		
¡Alarma! Valor límite sobrepasado	72 + CH	210		
Medida asociada al valor límite	72 + CH	208		
Notificación del receptor: cada vez el byte 1 más 16				

Tabla 2: magnitudes físicas		
Byte (hex)	Byte (dec)	Magnitud
01	1	Valor del ADC sin procesar
10	16	Tensión / Voltage
11	17	Corriente / Current
12	18	Resistencia / Resistance
14	20	Potencia / Power
21	33	Temperatura / Temperature
22	34	Humedad / Humidity
24	36	Presión / Pressure

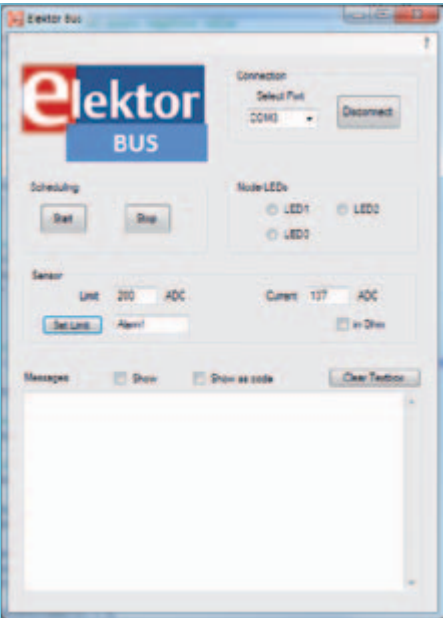


Figura 5. Entorno de usuario en nuestra pequeña estación de control.

medida de la luminosidad. Sin embargo, a los electrónicos probablemente también les interese saber el valor actual de la resistencia del fotosensor. Si conocemos el valor de la resistencia fija del circuito adicional, la resolución del ADC y la tensión de referencia (= 5 V), lógicamente podemos calcular la resistencia del sensor mediante el software de PC desde el valor transmitido del ADC. Pero ahora supongamos que alguien aparte ha fabricado el sensor, y se encuentra en una carcasa de difícil acceso. Por ello estaría bien que pudiésemos comunicar a nuestro sensor inteligente directamente mediante el bus que diera siempre la misma resistencia sin depender del valor del ADC.

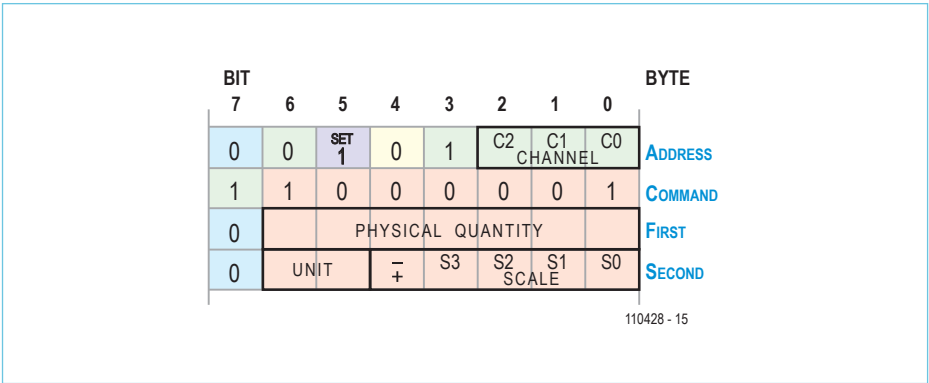


Figura 4. Para determinar la magnitud, unidad y escalado de un sensor inteligente, utilizamos cuatro bytes en total.

Otra unidad

Nuestro ElektorApplicationProtocol ahora ofrece esta posibilidad. Esta vez utilizamos cuatro bytes sucesivos: Address, Command, First y Second, como muestra la figura 4. En este caso el protocolo no describe el lugar que ocupan los cuatro bytes en nuestro mensaje. Por lo tanto, podemos utilizar tanto los primeros cuatro bytes útiles, como la segunda mitad de estos datos. La última opción permitiría, por ejemplo, transmitir dos valores a los actuadores (a Channel0 y Channel1). Todas las funciones especiales se marcan mediante un 1 en el bit más significativo del segundo byte. En el protocolo, el hecho de que utilicemos el bit más significativo supone una gran excepción, porque generalmente éste siempre es = 0. Hemos de recordar que tenemos que excluir el byte útil = AA_{hex} = 170, ya que este valor esta reservado al byte de start para nuestros mensajes. Pero en este caso utilizamos el valor fijo C1_{hex} = 193, principalmente para identificar el “tamaño, unidad y escalado” de la función. Consideremos ahora el pri-

mer byte. Ya que somos nosotros los que fijamos el tamaño/unidad y no queremos leerlos, establecemos SetBit = 1. Los bits C2 a C0 direccionan el canal correspondiente, o sea, el sensor (el que vamos a utilizar, que ya no está definido mediante la posición de los bytes). Ya que nuestro sensor de luminosidad se encuentra en Channel0, aquí se pone 0-0-0. El bit 3 del primer byte en el 4Byte-Mode es siempre = 1. Con este fácil truco nos aseguramos de que cualquier paquete de datos de 4 bytes siempre esté precedido de un byte mayor de 0. Con un valor de 0 en el byte el receptor identificará sin lugar a dudas que no se ha enviado “nada”. Resumiendo, en nuestro ejemplo esto significa lo siguiente: si queremos fijar para un sensor en Channel0 el tamaño, unidad y escalado, tendremos que enviar primero 28_{hex} y después C1_{hex}. Los últimos dos bytes son fáciles de explicar. El tercer byte determina el tamaño de lo que ha de transmitirse (puedes encontrar una de mis primeras propuestas en la tabla 2). Ya que para algunas dimensiones (por ejemplo la temperatura) es normal que existan

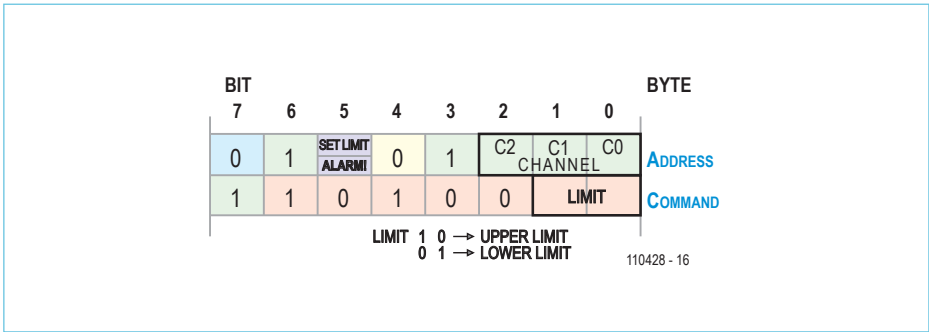


Figura 6. Cuando el bit 5 del primer byte es = 1, el sensor almacena el valor de medida actual como límite superior o inferior (Upper/Lower Limit). Con los dos bytes casi iguales el sensor dispara la alarma; en tal caso el bit 5 del primer byte es 0.

Números en coma flotante

Para tomar medidas eléctricas es necesario un rango de medida de varias potencias de 10 con la suficiente precisión. En nuestro 2ByteMode no podemos transferir semejante rango dinámico de valores, ya que sólo disponemos de 10 bits más el signo. Por ello, en este caso nos servimos de 4 bytes para transmitir un valor de medida o fijado por el usuario. En la figura puede verse otra vez el direccionamiento descrito

en el texto de un sensor o actuador individual en el nodo (gracias a los bits de canal C2 a C0 incluidos en el primer byte). Los bytes High,

BIT										BYTE	
7	6	5	4	3	2	1	0				
0	0	SET CURRENT	ACK ORIG.	1	C2	C1	C0	ADDRESS			
0	0	Float NO Float	MSIGN +	M3 D18	M2 D17	M1 D16	M0 D15	HIGH			
0	D13	D12	D11	D10	D9	D8	D7	MIDDLE			
0	D6	D5	D4	D3	D2	D1	D0	Low			

110428 - 18

Middle y Low transfieren el verdadero valor numérico. High.6 = 1 especifica que se trata de un número en coma flotante; High.5 determina el signo. MSIGN, M3, M2, M1 y M0 definen la magnitud (potencia de diez), nos quedan 14 bits (D13 a D0) para el verdadero valor numérico. Así, pueden representarse números de hasta $16.383 \cdot 10^{+15}$ (sin cambiar de escala).

Si High.6 = 0, tendremos disponibles 19 bits para representar un número entero.

distintas unidades, en el último byte podemos codificar la que estemos utilizando. Para las unidades del Sistema Internacional como el Voltio, el Ohmio y el Amperio simplemente dejamos los bits correspondientes = 0. Los cinco bits más bajos del último byte se utilizan para el escalado. Los bits S3 a S0 para la magnitud y el signo permiten valores desde 10^{-15} a 10^{+15} . Esto debería bastar para la mayoría de aplicaciones de nuestros lectores ;-).

En nuestro ejemplo (la resistencia se da simplemente en Ohmios) hemos de enviar la tabla 12-00_{hex}; por lo tanto, el comando completo es 28-C1-12-00_{hex}.

Software Demo

Todo se ha incluido ya en el software demo, que puede descargarse gratuitamente en la página web de este artículo [3]. Al contrario que en las versiones previas del software, ahora el master domótico (dirección de nodo 10) también tiene que informar regularmente al planificador si este se lo pide (hasta ahora, nuestra central domótica sólo se dedicaba a enviar *AcknowledgeMessages*). En el software de PC se integra la instrucción por la cual el master domótico envía sus mensajes, por su simplicidad directamente en el bucle del planificador. Naturalmente, esto no es que sea lo más ortodoxo, ya que en realidad el planificador y el master son unidades independientes del bus. Sin embargo, este truco nos ahorra un montón de programación en cuanto a hilos de ejecución y sincronización de la temporización.

Si marcamos la casilla correspondiente en el entorno de usuario (ver la captura de la figura 5), el master domótico enviará los bytes 28-C1-12-00_{hex} a los nodos con sensor, recibiendo a partir de ahora los valores de medida en ohmios. Los bytes 28-C1-01-

00_{hex} reactivan la representación original. En el firmware en BASCOM la fotoresistencia se calcula mediante cifras en coma flotante, lo cual mantiene el código más ligero. En la línea *Resistor = ...* se configura el valor fijo de la resistencia.

Sin embargo, no pueden transmitirse valores de resistencia superiores a 1023 Ohmios. El 2ByteMode en principio no permite coma flotante, como ocurre por ejemplo en un multímetro con selector de escala. En el 4ByteMode sí es posible (véase el cuadro), pero en esta entrega no hacemos uso de él.

Todo bajo control

Al principio del texto prometí que pasaríamos a la primera aplicación. ¿Qué tal una pequeña luz de noche, que se encienda automáticamente cuando la claridad disminuya hasta un cierto nivel? Dicho umbral podrá ser chequeado desde la central domótica, pero en este caso optamos por otro método para volver a nuestro sensor algo más inteligente. Cuando se hace de noche, sencillamente hacemos que los nodos con sensores guarden el valor de medida actual como umbral límite. Siempre que se atraviese este valor, el sensor avisará con un mensaje especial de "alarma". Cuando se hace de día otra vez, el sensor comunicará nuevamente que se ha vuelto a cruzar el umbral de la alarma.

La figura 6 muestra la estructura de estos mensajes. Necesitamos dos bytes que puedan situarse en el lugar que queramos, dentro de los datos útiles, como ocurre en todas las funciones especiales. Por lo tanto, el sensor que envíe la alarma del umbral se identifica mediante los primeros con su número de canal (bits C2 a C0). Los valores correspondientes para el segundo byte son D1_{hex} para cuando el umbral se ha cruzado por abajo, D2_{hex} para cuando se ha cruzado

por arriba y D0_{hex} para enviar una notificación, como podría ser "todo dentro de los límites".

El master domótico utiliza un mensaje de este tipo para informar al sensor correspondiente de si el valor actual se trata del umbral superior o inferior. En este caso, mediante *SetBit*, que significa "valor fijado". En nuestro ejemplo (se atraviesa el umbral inferiormente en Channel0) la alarma se fija con 48-D1_{hex}, y el límite con 68-D1_{hex}. Por seguridad, el master domótico confirma el mensaje con 58-D1_{hex}, y el valor del umbral con 78-D1_{hex} (*AcknowledgeBit* fijado). Esta primera aplicación en práctica muestra lo útil que supone esta característica (véase el cuadro de texto).

En una de las próximas entregas de la serie mostraremos la forma de definir valores límite a voluntad (y no sólo el valor instantáneo de medida).

El primer actuador

En el software demo del PC, cuando se cruza el valor umbral se muestra la palabra "Alarm!" en un cuadro de texto. El valor límite se fija haciendo clic en el botón "Set Limit", y al mismo tiempo se mantiene el dicho límite actual en otro cuadro adicional. Pero en este caso todavía no hay control de ningún tipo.

Por ello, en el nodo 1 conectamos el pequeño circuito de la figura 7. Mediante el pin ADC0/PC0 del conector controlamos un pequeño relé de 5 V gracias a un transistor. El montaje completo puede verse en la figura 8. Para los nodos 1 y 2 se utiliza el mismo firmware, la variable *Devicemode* (que se halla en el byte 006 de la EEPROM) determina si se trata de un actuador (relé), o de un sensor (fotoresistencia). Correspondientemente, mediante ADC0/PC0 se configura como entrada analógica o salida digital.

¡El bus en marcha!

Nuestro lector André Goldberg ha sido el primero en construir la primera aplicación práctica conocida para el bus: un medidor de nivel para un depósito de agua. Naturalmente, aquí no hace falta un PC continuamente, sino que el planificador corre sobre un microcontrolador. Sin más dilación, este fan de la electrónica ha implementado un pequeño planificador con la ayuda de un sencillo temporizador en BASCOM. Después de que yo enviase un esbozo del protocolo y el software demo para esta entrega, André desarrolló un control automático para llenar este depósito. Lamentablemente, se percató de que el mensaje de alarma no llegaba siempre correctamente al receptor (aunque teniendo en cuenta el mínimo tiempo utilizado, tampoco podemos lamentarnos). Para aplicaciones críticas, el protocolo ofrece la posibilidad de que se

confirme el mensaje por parte del receptor. El emisor debe entonces repetir la alarma hasta recibir un AcknowledgeMessage de vuelta (con los mismos bytes, pero con el AcknowledgeBit activado). Esto ya se ha implementado en el software demo de esta entrega. El Sendalarmflag del firmware del controlador tiene que ser = True, si se atraviesa el valor límite, e = False, cuando se recibe la confirmación del mensaje de alarma del master domótico. Los nodos con sensores, a los que se les pregunta regularmente, sólo envían las alarmas mientras Sendalarmflag se encuentre activo. Si entre tanto se vuelve a atravesar el valor límite, se cancela la alarma, activando correspondientemente Sendresetalarmflag (reseteando a la vez Sendalarmflag, o a la inversa). En el software de PC nos servimos de las variables enteras intSetAlarmstatus e intResetAlarm-

Status. Un 2 significa que se ha recibido la alarma y ha de activarse el relé, mientras que un 1 que la alarma ha de confirmarse por el sensor, y un 0 que no hay que enviar más mensajes. Los estados transcurren en este orden: 0 > 2 > 1 > 0... Primero se envía el mensaje de activación al relé, para no perder tiempo. Aquí se lanza otro mensaje de alarma, que pone intSetAlarmstatus a 2, a la vez que intResetAlarmstatus a 0 (y viceversa), en caso contrario los cambios demasiado rápidos podrían dar como resultado estados indefinidos.

Actualmente André Goldberg está experimentando con un módulo servidor web de Ulrich Radig, para poder visualizar los valores de medida (desde un navegador web). Más acerca de este interesante tema en la siguiente entrega de la serie.

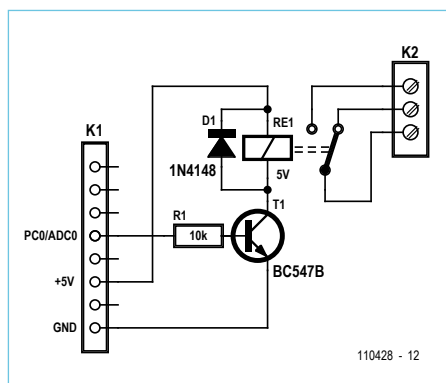


Figura 7. El circuito del relé puede montarse sobre una pequeña tarjeta perforada.

Si el master domótico recibe la alarma del umbral del nodo 2, envía al nodo 1 los bytes 60-01_{hex}, poniendo así PC0 a nivel alto. Cuando 2 nodos indican que la alarma ya no está activa, la central domótica envía 60-00_{hex} al nodo 1. Éste pone PC0 a nivel bajo y el relé se desactiva nuevamente. Quien haya prestado atención anteriormente sabe de sobra que 60_{hex} corresponde al bit 6 (2ByteMode) y a activar SetBit.

Si ya hemos establecido un valor límite (y todo está correctamente soldado, programado y cableado), ahora la iluminación de

emergencia conectada al relé se encenderá cuando la claridad sea inferior a dicho valor. ¡Voilà!

(110428)

¡Forma parte del desarrollo! ¡Cualquier consejo o sugerencia son bienvenidos en el mail de nuestra redacción, redaktion@elektor.de

Enlaces

- [1] www.elektor.es/110258
- [2] www.elektor.es/110382
- [3] www.elektor.es/110428

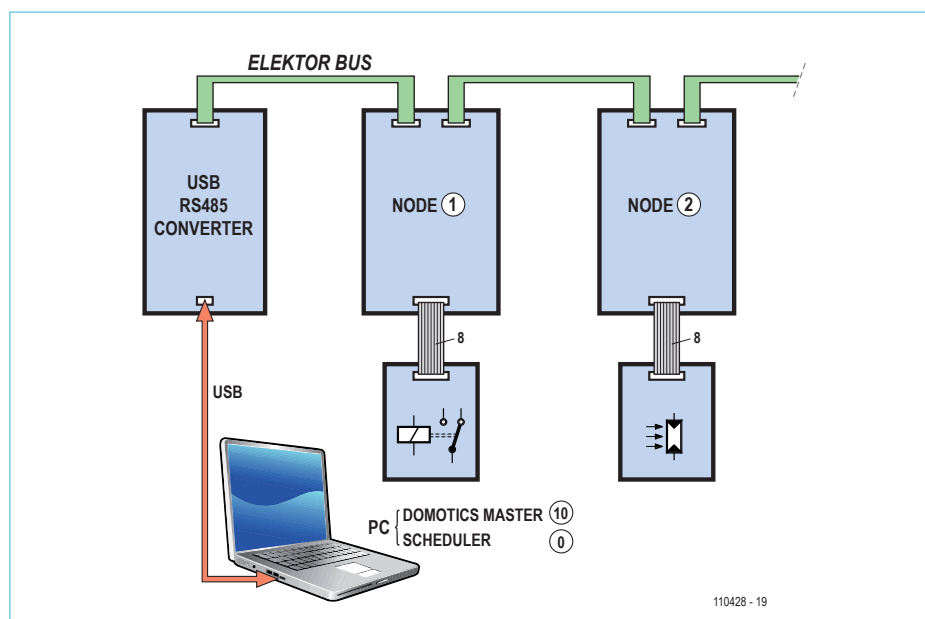


Figura 8. La pequeña aplicación de un vistazo. El PC hace al mismo tiempo de planificador del bus (dirección 0) y de central de la aplicación (dirección 10).

Reflectometría en el Dominio del Tiempo

Guía para principiantes para la localización de cortocircuitos y circuitos abiertos en cables largos

Seguro, un óhmetro es un instrumento manual que permite comprobar hilos y cables para descubrir circuitos abiertos y cortocircuitos. Pero no es tan sencillo el localizar un circuito abierto o un cortocircuito cuando el cable es muy largo,

o si el fallo está escondido dentro de un muro o bajo una calle. Localizar circuitos abiertos o cortocircuitos parciales, o tan solo malas conexiones, es incluso peor. Este es el motivo de que el TDR sea tan práctico y útil.

Peter A. Stark (USA)

TDR significa dos cosas: el proceso de Reflectometría en el Dominio del Tiempo (o "Time Domain Reflectometry"), o el dispositivo que lo hace, llamado Reflectómetro en el Dominio del Tiempo (o "Time Domain Reflectometer"). También tenemos el OTDR, que no es otro que un TDR Óptico para fibras ópticas. La idea básica es la de enviar una señal, normalmente un pulso, sobre el cable o fibra óptica y, a continuación, ver si hay algún problema en el cable con una parte o una total reflexión del pulso que vuelve hacia la entrada. La naturaleza de esta reflexión nos da una idea de cuál es el problema, mientras que el tiempo que tarda la reflexión en alcanzar de nuevo la entrada nos dice lo lejos de dicha entrada en que se encuentra el problema.

Generalmente, los instrumentos de prueba TDR son bastante caros y no se encuentran como equipo básico en el banco de trabajo de un aficionado a la electrónica. Incluso muchos comerciales no nos podrán proporcionar uno. Pero, si estamos dispuestos a hacerlo, podemos fabricarnos un útil instrumento TDR con tan sólo un generador de

pulsos y un osciloscopio. Podemos construir fácilmente un útil generador de pulsos con un pequeño esfuerzo y el osciloscopio no tiene por qué ser caro. Obviamente, cuanto mejor sea, mejores resultados obtendremos, pero incluso un viejo osciloscopio de hace 15 años, de 10 MHz, sería suficientemente bueno (ese es el que hemos utilizado para las fotos de este artículo).

Transitorios y estado estacionario

El circuito de la **Figura 1** es simple: con el interruptor abierto, no hay corriente. Pero con el interruptor cerrado la ley de Ohm dice que por el circuito fluye una corriente de 0,1 A (10 V divididos por 100 ohmios). Pero vamos a hacer el circuito algo más interesante: vamos a hacer que los hilos que van desde la resistencia al conmutador sean mucho más largos (por ejemplo, unos 300.000 km de largo). Para hacer las cosas más sencillas, vamos a suponer que el hilo es perfecto y que tiene una resistencia cero. ¿Ahora qué? La ley de Ohm no tiene en cuenta cuál es la longitud del cable, es decir, la ley dice que $I = V/R$, por lo que la corriente

sigue siendo de 0,1 A. ¡Pero hay mucho más que contar!

La razón de haber elegido 300.000 km para la longitud del hilo es que la luz viaja a la velocidad de 300.000 km/s. Si estuviésemos situados en la posición del interruptor, con un telescopio mirando hacia la resistencia, la luz tardaría 1 segundo en alcanzarnos. Lo que vemos no es lo que está sucediendo ahora, sino lo que sucedió hace 1 segundo. Cuando cerramos el interruptor, ¡incluso no podemos estar seguros de que la resistencia está aún allí! Supongamos que hay un gracioso que nos cambia la resistencia por otra de otro valor o que, incluso, la desconecta, justo antes de cerrar el interruptor. No sabremos que se ha producido hasta un segundo después. Por lo tanto, ¿cómo podemos saber (o mejor aún, como sabe la pila) que hay que enviar 0,1 A de corriente sobre la línea, hacia la resistencia, cuando cerramos el interruptor?

La respuesta es que no lo sabemos. La corriente que se inicia en el hilo depende del hilo, no de la carga. Incluso si asumimos que el hilo no tiene ninguna resisten-



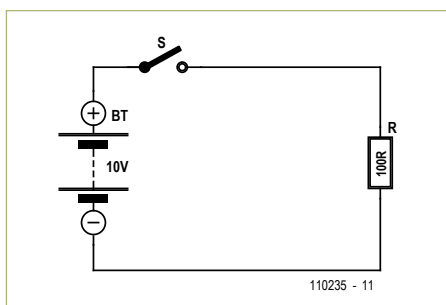


Figura 1. Un sencillo circuito DC... ¿o no?

cia, éste posee alguna inductancia (aunque pensemos que no está bobinado sobre una bobina) y algún tipo de capacidad entre los dos hilos. Esta inductancia y capacidad limitan el flujo de corriente a un cierto valor. En nuestro caso, la corriente tardará algo más de un segundo en alcanzar la resistencia (ya que la corriente en el hilo viaja más lenta que la velocidad de la luz), pero si el cable fuese infinitamente largo, esta corriente continuaría fluyendo de forma indefinida. Sabiendo la tensión y la corriente, podemos calcular una resistencia equivalente que podría provocar el mismo flujo de corriente. El valor de dicha resistencia equivalente es lo que llamamos la *impedancia característica* del cable. El símbolo de esta impedancia es Z_0 . (La llamamos impedancia incluso si sabemos que se comporta como una resistencia, tan sólo para que sea algo más general.) Por ejemplo, el cable coaxial utilizado para muchas antenas tiene una impedancia característica Z_0 de 50 Ω , por lo que si conectásemos un cable infinitamente largo a nuestra pila de 10 V, la corriente que pasaría por el cable (usando de nuevo la ley de Ohm), sería de:

$$I = V / Z_0 = 10 / 50 = 0,2 \text{ A o } 200 \text{ mA.}$$

Vale, ahora vamos a suponer que estamos utilizando algo similar a un cable coaxial (pero hecho con un hilo perfecto de resistencia cero) entre la pila y nuestra resistencia de 100 Ω de la **Figura 1**. La pila no tiene forma de saber cuál es la longitud real del cable, por lo que comienza a suministrar los 200 mA de nuestro ejemplo. Pasado algo más de un segundo, estos 10 V y 200 mA alcanzan la resistencia. En ese momento, la resistencia de 100 Ω pone una objeción: “¡Habéis enviado o mucha corriente o muy poca tensión! Si insistís con los 10 V, entonces dadme 0,1 A o 100 mA. Pero si me dais 200 mA, entonces la tensión correcta sería de 20 V (V igual a I veces R). ¡Solucionadlo!” En este punto, el cable responde con: “Lleguemos a un compromiso. Vamos a aumentar la ten-

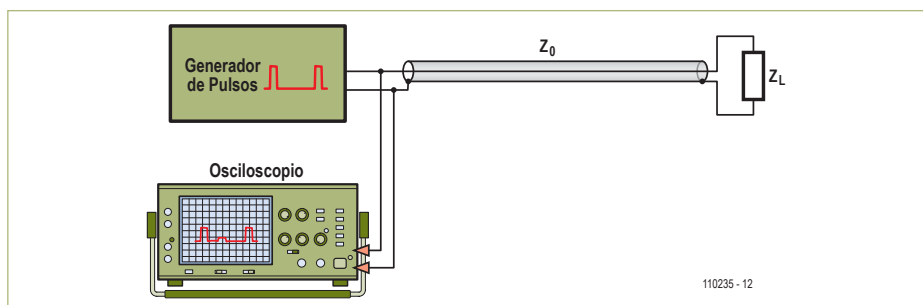


Figura 2. Circuito para enviar pulsos dentro de un cable.

sión un poco y bajar la corriente en el mismo porcentaje.” Así pues, hacen un pequeño cálculo y deciden hacer un cambio de 1/3: aumentarán la tensión en un 1/3 hasta los 13,3 V, y disminuirán la corriente en la misma proporción de 1/3 hasta los 133 mA. Ahora la resistencia ya es feliz ya que:

$$I = V / R = 13,3 \text{ V} / 100 \Omega = 0,133 \text{ A.}$$

que equivale a 133 mA.

Un nuevo problema: la tensión y la corriente en el extremo derecho del cable es diferente del extremo izquierdo. Así que la nueva tensión y corriente comienzan su viaje de retorno hacia la pila.

Consideremos tan sólo la tensión: los 10 V están viajando desde la izquierda hacia la derecha pero una tercera parte de dicha tensión (3.3 V), está siendo reflejada desde la derecha hacia la izquierda. Más o menos un segundo después, esta tensión reflejada alcanza la pila y ahora la pila dice: “¡De ninguna manera, \$%&* &@! ¡Quiero 10 V!” Por lo que obliga a que la tensión sea de nuevo de 10 V, lo que también implica el reducir la corriente. Aproximadamente otro segundo más tarde, la resistencia...

Bien, ya podemos ver lo que está sucediendo. La pila intenta bajar la tensión, la resistencia intenta bajarla y, durante todo este tiempo, la corriente va bajando lentamente. Finalmente, los valores se establecerán en 10 V y 0,1 A, que es lo que esperábamos, pero eso lleva su tiempo. El estado inicial de compromiso se denomina *transitorio*; la situación final a la que se llega, con un valor establecido, se denomina *estado estacionario*.

Esta interacción entre la tensión y la corriente sucede siempre, independientemente de la longitud del cable, pero si el cable es muy corto este hecho se produce muy rápido. Como valor aproximado, las señales eléctricas en un cable viajan a unos 20 cm/ns (que es la milmillonésima parte de un segundo), por lo que si un hilo desde la pila a la resistencia tiene una longitud de 1 m, el proceso completo dura unos pocos nanosegundos.

Antes de lo que dura un parpadeo, la tensión y la resistencia han ido cambiando su valor hasta alcanzar su estado estacionario y nadie se ha dado cuenta de que se ha producido el estado transitorio. Necesitaríamos un osciloscopio con capacidad de almacenamiento extremadamente rápida, para intentar ver que esto sucede. Por otro lado, si disponemos de un cable lo suficientemente largo, incluso con un osciloscopio moderadamente decente, se podría ver el estado transitorio. Esto es todo lo que podemos decir sobre la TDR o *Reflectometría en el Dominio del Tiempo*.

El Coeficiente de Reflexión

¿Cómo han decidido el cable y la resistencia que 1/3 o el 33% sería un buen valor de compromiso? Han usado la siguiente ecuación:

$$\text{Coeficiente de Reflexión} = (Z_L - Z_0) / (Z_L + Z_0)$$

donde Z_L es la resistencia de la carga y Z_0 es la impedancia característica del cable. (Como hemos dicho antes, dicha carga es, normalmente, resistiva pero los ingenieros utilizaron la Z para la impedancia... como en este caso.) Con nuestras cifras, tenemos un coeficiente de reflexión de:

$$= (100 \Omega - 50 \Omega) / (100 \Omega + 50 \Omega) = 50 \Omega / 150 \Omega = 0,333 = 1/3.$$

Esto se denomina el *coeficiente de reflexión* ya que nos indica que fracción de la señal generada vuelve de nuevo reflejada hacia el principio.

Puede ser tanto positivo como negativo (si $Z_0 - Z_L$ es negativo) o incluso cero (si Z_L es igual a Z_0), así pues, la reflexión puede ser positiva, negativa o cero.

Los Pulsos son lo mejor

Supongamos que, en lugar de cerrar el interruptor y dejarlo cerrado, lo cerramos justo durante un instante e, inmediatamente después, lo abrimos de nuevo. Los 10 V serán sólo un corto pulso que viajar directamente;

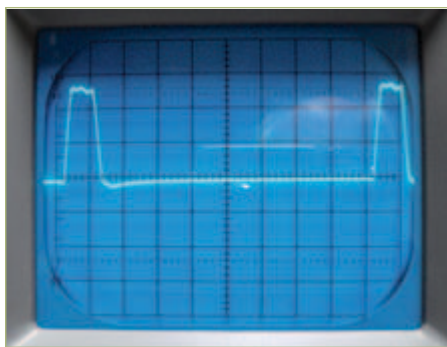


Figura 3. Salida del generador de pulsos.

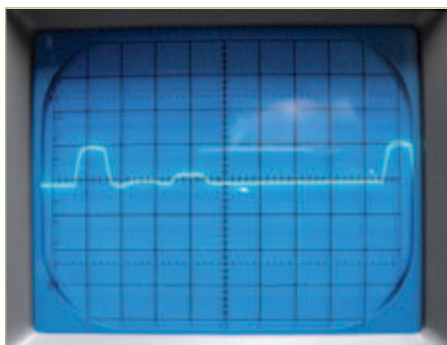


Figura 4. Pulsos reflejados desde una resistencia de 100 Ω.

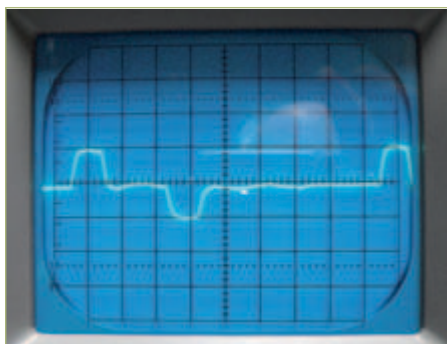


Figura 5. Reflexión desde un cortocircuito.

mientras que los 3,3 V serán también un corto pulso que vuelve hacia atrás un instante después.

Como tan sólo hay un pulso disminuyendo su nivel y uno de vuelta, necesitamos un osciloscopio con almacenamiento para capturarlo, por lo que vamos a repetir el pulso a muy alta frecuencia, usando un generador de pulsos conectado como se muestra en la **Figura 2**. (Si es posible utilizaremos una punta de prueba de osciloscopio con relación de 10:1.) Aunque el esquema eléctrico muestra un cable coaxial, podríamos utilizar cualquier tipo de cable.

La **Figura 3** muestra la salida del generador cuando se lleva directamente a un osciloscopio y no hay cable. El osciloscopio ha sido configurado a una sensibilidad vertical de 5

V/div y una velocidad de barrido horizontal de 0,1 μ/div, de manera que los pulsos tengan, aproximadamente, 0,1 μs de ancho y unos 0,9 μs de separación.

Vamos a conectar ahora unos 3 m de cable coaxial RG-58U, de 50 Ω, con la resistencia de 100 Ω en su extremo más lejano, lo que nos da un montaje como el que se muestra en la **Figura 4**. Vemos dos cosas: nuestros pulsos originales se han hecho más pequeños debido a que la impedancia característica del cable está cargando al generador, y que ahora hay un tercer pulso pequeño en la foto. Este nuevo pulso (el segundo desde la izquierda en la foto) es la reflexión del primer pulso proveniente del terminal abierto del cable. El pulso tarda unos 150 ns en alcanzar el terminal lejano del cable y otros 150 ns en retornar hacia el osciloscopio, de manera que el retardo total es de unos 300 ns o 0,3 μs. Así pues, el pulso reflejado se muestra unas tres divisiones a la derecha del pulso de salida. Si la pantalla del osciloscopio fuese más ancha, podríamos ver un pulso reflejado a la derecha de cada pulso de salida. Como hemos calculado que el coeficiente de reflexión era de 1/3, el pulso reflejado tiene alrededor de 1/3 de la altura del pulso de salida. Hay algunas pérdidas en el cable, por lo que, generalmente, el pulso de retorno será un poco más pequeños de lo que se espera.

Vamos a mirar con más detalle el retardo. El osciloscopio estaba configurado a 0,1 μs/div, y la distancia entre los dos pulso es de unas tres divisiones, por lo que el viaje de ida y vuelta dura unos 0,3 μs o 300 ns. Para saber si es correcto, necesitamos saber lo rápido que las señales eléctricas viajan en un cable en concreto (ya que la velocidad depende del cable).

Buscando en las especificaciones de un cable coaxial RG-58U normal, vimos un parámetro denominado el *factor de velocidad*, que era igual a 0,66. Esto significa que, en este cable en particular, las señales viajan a 0,66 veces la velocidad de la luz. Como la velocidad de la luz en el vacío es de 300.000 km/s, esto equivale a unos 198.000 km/s. Este valor lo multiplicamos por 1000 para convertir el dato en m/s, lo que nos da un valor de 198.000.000 de m/s, lo que equivaldría a unos 198 m/μs o 0,198 m/ns. Por lo tanto, los 300 ns de retardo de la **Figura 4** equivalen a 300 veces 0,198 m, es decir,

unos 59,4 m. Debemos recordar que esta es la longitud del viaje de ida y vuelta, lo que equivaldría a un cable de unos 30 m. Si pudiésemos medir el retardo de manera más precisa en el osciloscopio, podríamos utilizar este método para medir la longitud del cable de modo bastante preciso, sin tener que desenrollarlo de su carrete.

Por supuesto, hay una pega: necesitamos un buen osciloscopio bien calibrado y también necesitamos saber el valor exacto del factor de velocidad. Pero un equipo TDR comercial puede hacer un trabajo bastante preciso con todo ello.

¿Y qué hay de las otras cargas?

Además de la resistencia de 100 ohmios, hay otros tres valores que son de nuestro interés: un cortocircuito (o más comúnmente, un “corto” o 0 Ohmios), un circuito abierto (una resistencia infinita o un valor infinito de ohmios) y una resistencia exactamente igual a la impedancia característica del cable (es decir, $Z_L = Z_0$). Vamos a echar una ojeada a estos tres casos:

A. Si el terminal del cable está cortocircuitado, es decir, Z_L es 0, y, por lo tanto, el coeficiente de reflexión es

$$(Z_L - Z_0) / (Z_L + Z_0) = (0 \Omega - 50 \Omega) / (0 \Omega + 50 \Omega) = -50 \Omega / 50 \Omega = -1$$

Esto significa que el 100% del pulso de salida es reflejado, pero el pulso es negativo, con lo que la polaridad se ha invertido. Esto se muestra en la **Figura 5**.

B. Si el extremo está abierto, Z_L es infinita y, por lo tanto, Z_0 es tan pequeña en comparación con la enorme Z_L que podemos olvidarnos de ella:

$$(Z_L - 50 \Omega) / (Z_L + 50 \Omega) \approx Z_L / Z_L = +1$$

El coeficiente de reflexión es +1, lo que significa que el pulso completo es de nuevo reflejado pero, en este caso, la polaridad es la misma. Esto se muestra en la **Figura 6**.

C. Por último, supongamos que la resistencia colocada en el extremo del cable es igual a la impedancia característica del cable. Así que tenemos:

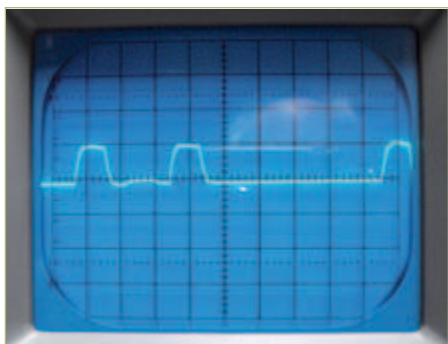


Figura 6. Los pulsos rebotados en un circuito abierto son positivos.

$$(Z_L - Z_0) / (Z_L + Z_0) = (50 \Omega - 50 \Omega) / (50 \Omega + 50 \Omega) = 0 / 100 = 0$$

Esta vez el coeficiente de reflexión es 0, por lo que no hay ninguna reflexión. Este caso es el que se muestra en la **Figura 7**, donde he colocado una resistencia de 51 ohmios al final del cable de 50 ohmios. (Hay un ligero pico después del pulso de salida, causado por mi cable de 3 m, que realmente está compuesto de dos longitudes de 1,5 m unidos entre sí. Esto muestra una ligera reflexión en el punto donde se conectan. Haciendo que la resistencia de carga sea igual a la impedancia característica del cable evitamos las reflexiones y a esto se le denomina la *terminación adecuada del cable*. Así pues, aquí tenemos una manera fácil de encontrar la impedancia característica de un cable: intentaremos colocar diferentes valores de resistencia al final del cable hasta que no tengamos ninguna reflexión y, en ese momento, Z_0 es igual al valor de la resistencia de carga.

Así pues, todo lo que necesitamos es un generador de pulsos

Si ya disponemos de uno en nuestro banco de trabajo, tenemos todo lo que necesitamos. Pero si no es así, podemos construir uno por menos de 4 €. La **Figura 8** muestra un sencillo esquema eléctrico de cómo trabaja. Sólo necesitamos un temporizador 555, que proporciona unos pulsos negativos muy cortos a una velocidad de, aproximadamente, 1 MHz (la frecuencia exacta no es importante y varía debido a las tolerancias), seguido por un amplificador que controlará el cable. Seis inversores digitales en paralelo invierten el pulso para hacerlo positivo y también proporcionan suficiente potencia de salida. La resistencia de 33 Ω de la salida incrementa la resistencia de salida para evitar pulsos de retorno de rebote de los inversores y que vuelvan de nuevo al cable por segunda vez.

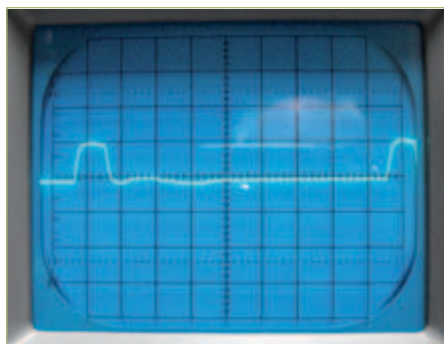


Figura 7. No hay reflexión en un cable correctamente terminado.

A pesar de la aparente sencillez del circuito, necesitamos seleccionar nuestros componentes cuidadosamente. A los circuitos integrados temporizadores 555 normales no le gusta trabajar a las rápidas velocidades que necesitamos, por lo que nuestra mejor elección es la de un temporizador TLC555 CMOS, de Texas Instruments. Afortunadamente, Radio Shack lo suministra con su código "276-1718". Para los inversores, se recomienda utilizar un inversor 4069 CMOS de 6 elementos, principalmente debido a que puede trabajar con la tensión de 9 V de una pila. También podemos utilizar un circuito integrado TTL 7404 o cualquiera de los circuitos integrados de la familia 74xx04 que sean compatibles a nivel de terminales, pero tendríamos que alimentar, tanto al 555 como al inversor con una tensión de alimentación de 5 V, lo que incrementaría la complejidad del circuito. El circuito debería ser fácil de montar sobre un trozo de placa perforada o si lo prefieren nuestros lectores, sobre una placa de circuito impreso diseñada expresamente para ello.

(110235)

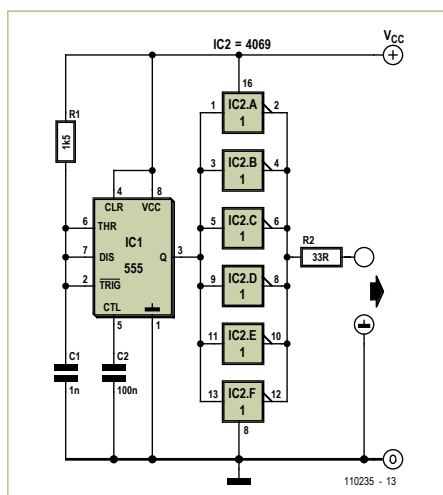


Figura 8. Esquema eléctrico de un generador de pulsos.

Deberes para ti

Una vez que tenemos nuestro generador de pulsos funcionando, os presentamos aquí algunos experimentos sencillos que podéis intentar. Es posible que necesitemos ajustar el mando de barrido del osciloscopio para conseguir una pantalla estable:

1. Conseguir 1,5 m o 3 m de cualquier tipo de cable, lo extenderemos y le pediremos a algún amigo que conecte algo en el extremo más lejano. Practicaremos con nuestro TDR las técnicas vistas con el generador de pulsos y el osciloscopio para intentar adivinar lo que hay conectado en el otro extremo. El tipo de cable no es importante.
2. Comprobar si hay mucha diferencia entre el cable totalmente estirado o enrollado.
3. Mediremos la longitud real del cable y comprobaremos cuánto tarda un pulso en hacer el recorrido completo y calcularemos su factor de velocidad.
4. Comprobar lo corto que puede ser el cable antes de que el pulso de salida y el pulso de retorno se superpongan uno sobre otro, de manera que no podamos verlos como pulsos separados. Si el cable es lo suficientemente corto, podemos ver que los dos pulsos se suman para formar un pulso con el doble de tensión.
5. Tomaremos ahora un cable desconocido, como el cable de unos altavoces o el cable que se utiliza para dar tensión a una bombilla, y mediremos su impedancia característica.
6. Conectaremos ahora dos cables diferentes, extremo a extremo. Si sus impedancias características son suficientemente diferentes, podremos ser capaces de ver una reflexión de la conexión donde los dos cables diferentes se unen.
7. Usando una resistencia igual a la Z_0 del cable, la conectaremos al extremo del mismo con sus terminales muy cortos. No deberemos ver ninguna reflexión. Ahora la conectaremos al mismo cable pero con unos terminales de unos 30 cm o 60 cm. Probablemente seremos capaces de ver unas pequeñas pero claras reflexiones. ¿Qué está sucediendo? ¿Son estos terminales los que hacen la diferencia?
8. Usemos algo de nuestro tiempo pensando sobre ello: supongamos que la señal que pasa por el cable fuese una onda senoidal en lugar de pulsos. ¿Qué sucedería?

Alternativa al Fan-Flash

Efectos estroboscópicos con lógica digital clásica

Este proyecto está basado en el 'Fan-Flash' que publicamos en la edición de diciembre de 2010 [1], un circuito con el que, aparentemente, podemos parar las aspas del ventilador de un PC con la ayuda de luz pulsante. A raíz de los comentarios de que el valor educativo del proyecto sería mayor si realizásemos el circuito sin microcontrolador, presentamos aquí un circuito con la misma función, pero construido únicamente con lógica digital.

Raymond Vermeulen (laboratorio de Elektor)



El principio del circuito es el mismo: hacemos que por cada revolución del ventilador, un(os) LED de(n) algunos destellos, que se corresponden con el número de aspas del ventilador. El esquema describe nuestro circuito, que fue diseñado para un ventilador con una velocidad de giro de 750 revoluciones por minuto (rpm) y nueve aspas.

Si quieres utilizar un ventilador con otro número de aspas tendrás que adaptar unas cuantas cosas.

LED

Ten en cuenta cuánto quieres que consuma(n) el/los LED. Elige un valor apropiado para R1 basándote en el consumo requerido (ver **figura 1**). Una buena regla de tres para LED normales es de 20 mA en modo continuo y un máximo de 500 mA por pulso. Seleccionamos 3,5 V como la tensión umbral. El cálculo que hemos hecho para el ventilador utilizado y cuatro LED de 5 mm de color blanco puestos en paralelo es el siguiente:

El consumo total de los LED asciende a $(12 \text{ V} - 3,5 \text{ V}) / 12 \Omega = 0,708 \text{ A}$. De modo que cada LED consume 0,177 A durante

un pulso. El ventilador gira con 750 rpm = 12,5 Hz y hay que generar nueve pulsos (el número de aspas) por revolución. De esto se deduce que hay que dar $12,5 \times 9 = 112,5$ pulsos por segundo. Un solo pulso tiene un ancho de 0,11 ms (ver párrafo *El NE555*), de modo que el ancho de pulso total por segundo asciende a $112,5 \times 0,11 \text{ ms} = 12,38 \text{ ms}$. Con esto fijamos el ciclo de trabajo en 1,238%. La corriente media que tiene que soportar cada LED por segundo asciende a $177 \text{ mA} \times 0,01238 = 2,19 \text{ mA}$. Así que ambos valores de la corriente quedan holgadamente dentro del consumo máximo.

Igual que en el artículo original, puedes elegir libremente el tipo de LED utilizado. Es fácil dimensionar el circuito tanto para LED de potencia como para unos LED de 5 mm puestos en paralelo. Asegúrate, en este último caso, de que sean LED del mismo tipo y preferiblemente del mismo lote, ya que en caso contrario existe la posibilidad de que la corriente no se reparta homogéneamente sobre los LED, provocando que algunos se iluminen más fuertemente y otros más débilmente.

MOSFET

En el esquema se ve claramente que la parte que lleva una intensidad elevada ha quedado, en gran medida, igual. Las diferencias son que se ha optado por otro MOSFET y que el conector del ventilador de 3 vías no se conecta a la placa madre sino directamente sobre el circuito. Por eso se ha colocado una resistencia pullup entre la línea taco y la de 12 V.

El MOSFET no tiene que ser necesariamente un IRF3704. Si optas por otro, hay que asegurarse de que sea uno del tipo canal-n y de que se abra completamente con 4 V. Por ejemplo, una V_{th} por debajo de unos 2,5 V vale perfectamente. Un consejo: Busca en la hoja de características la gráfica con V_{ds} en el eje horizontal e I_d en el eje vertical. Escoge la línea de 4 V (o la línea más próxima a esta), síguela hasta que $V_{ds} = 12 \text{ V}$ y lee el valor correspondiente de I_d . La curva tiene que ser plana e I_d tiene que ser mayor a la corriente pico que queremos que consuma(n) el/los LED. Selecciona un amplio margen por seguridad. Asegúrate también de que se cumplen los tiempos de 'rise' y 'fall'. Adopta como

norma el 10% del tiempo del pulso para mantener la forma del pulso limpio. La mayoría de los MOSFET lo cumplen sin ningún problema.

Lógica digital

Seguimos con ‘el cerebro’ del esquema. Para sincronizar la frecuencia de parpadeo de los LED con la señal ‘taco’ de la velocidad de giro medida del ventilador, empleamos un PLL (bucle de enganche de fase): el 4046 (IC3). Este compara la frecuencia de las señales en el terminal 3 (CIN) y el terminal 14 (SIGN) y adapta a continuación la frecuencia de su señal de salida en el terminal 4 (VCO), de tal modo que no quede ninguna diferencia entre ambas señales de entrada. Utilizando un divisor de frecuencia (IC1 junto con IC5) en el bucle de realimentación, lo convertimos en un multiplicador de frecuencias.

Esto funciona así:

Supón que el terminal 14 recibe una señal de 1 Hz. El terminal 3 aún no recibe ninguna señal (la frecuencia es 0 Hz). La salida (terminal 4) intenta eliminar la diferencia y genera, por ejemplo, 1 Hz. El contador y el biestable dividen esta frecuencia por un número determinado (en nuestro caso 18). Entonces la diferencia con la frecuencia de la señal en el terminal 14 disminuye, pero aún no es 0. Así que aumenta la frecuencia en la salida tanto como se necesite para que las frecuencias en ambas entradas sean las mismas. Como en nuestro caso el contador y el biestable dividen la señal por 18, en la salida del PLL aparece una señal 18 veces la frecuencia de la señal de entrada del terminal 14. Con esto hemos configurado la frecuencia de pulso con el ventilador utilizado. A parte de eso se necesita el biestable (IC5A), porque la salida de IC2B lleva pulsos cortos (debido a que la realimentación está conectada al reset del contador con IC2B). El biestable lo convierte de nuevo en una señal con un ciclo de trabajo del 50 % (y divide la frecuencia de pulso por dos). IC2B añade un retardo adicional para estar seguro de que IC1 no se resetea demasiado rápido.

En la entrada taco, el biestable DIC7 reduce a la mitad la frecuencia de pulso de la señal taco. Esto es necesario porque IC1 sólo puede dividir la frecuencia por un número entero. Como IC5A reduce a la mitad la fre-

cuencia de referencia', se vuelve a hacer de nuevo en la entrada (con IC7B).

A 750 revoluciones por minuto la señal taco del ventilador del PC genera 1500 pulsos por minuto. La señal taco tiene, por lo tanto, una frecuencia de 25 Hz. Esto se divide por cuatro y se multiplica por 18, de modo que haya una frecuencia de 112,5 Hz en el terminal 4 de IC3.

El NE555

Convertimos esta frecuencia en una señal modulada por ocho de pulsos para el con-

trol del/de los LED. Para eso utilizamos el famoso 555 (IC4), configurado como monoestable. El pulso en la entrada de disparo no puede ser más ancho que el pulso que sale del 555. La línea de 112,5 Hz tiene que conectarse de forma capacitiva (C11) debido a la baja frecuencia y al ciclo de trabajo del 50%. Así surge un pico suficientemente estrecho en la entrada de disparo del 555 por cada flanco bajante. El ancho de pulso de la señal que el 555 genera, se determina con la siguiente fórmula:

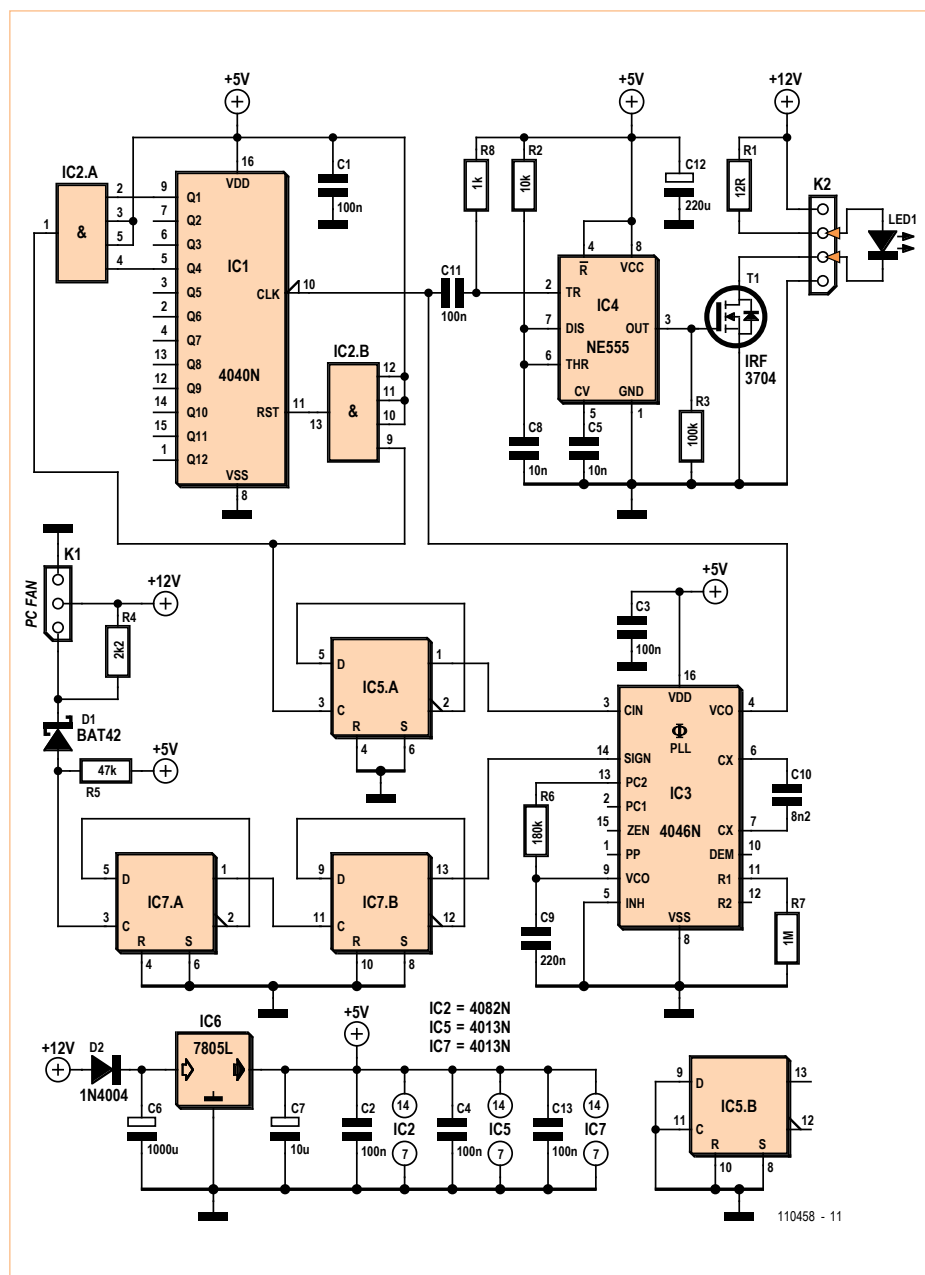
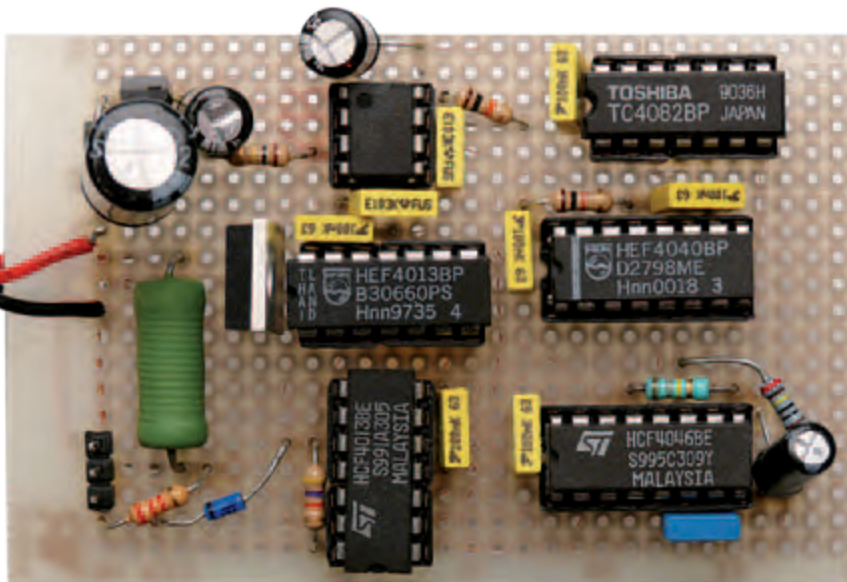


Figura 1. En el esquema vemos ahora una construcción ‘clásica’ de un circuito digital.



Tiempo(s) = $1,1 \times R2 \times C8$.

Con los valores utilizados se generan pulsos de 0,11 ms en la salida (terminal 3). Por cierto, C12 evita que la línea de 5 V se hunda en el momento en que el 555 genera el pulso.

Con C10 y R7 se determina el alcance con el que el PLL puede enganchar. Este se puede aproximar con la fórmula:

$$2 \times f_{center} = f_{max} = \frac{1}{0,5 \times R7 \times C10}$$

Esto proporciona un alcance de entre 0 hasta 244 Hz con los valores de los compo-

ponentes prescritos. La frecuencia utilizada — 112,5 Hz — encaja perfectamente dentro de este alcance. Si quieres frecuencias más altas, puedes sustituir C10 con un valor de capacidad más bajo.

El filtro de paso bajo formado por R6 y C9 determina el área de frecuencia en el que la entrada se engancha. Esto es según la fórmula:

$$f_{lockrange} = \left(\frac{1}{\pi}\right) \times \sqrt{\frac{2 \times \pi \times f_{center}}{R6 \times C9}}$$

por lo que surge un alcance de 44,3 Hz con los componentes utilizados. Por eso puede

parecer que el ventilador balancea durante el uso. Modificando estos componentes, surgen efectos variables.

El circuito se alimenta con una tensión de 12 V que generalmente está siempre disponible en un PC. Un 7805 la convierte en la tensión de 5 voltios estabilizados, necesaria para los integrados.

Consejos adicionales:

Si se utiliza un ventilador con otro número de aspas, puedes adaptar el circuito conectando la salida del integrado contador IC1 al IC2A de forma correspondiente (ahora se divide por 9 porque el ventilador utilizado tiene nueve aspas).

Puedes sustituir el BAT42 del esquema por, por ejemplo, un BAT48 o un BAT43.

Puedes sustituir el 1N4004 por, por ejemplo, un 1N4007.

No se ha utilizado el IC5B, es una buena práctica conectar con masa las entradas de componentes no utilizados.

Si no puedes encontrar un conector de ventilador de 3 vías, puedes utilizar también una simple tira de pines. El paso de esta asciende a 2,54 mm igual que en la conexión especial Molex.

(110458)

Enlaces Web:

[1] www.elektor.es/100127

Publicidad

EURO
CIRCUITS

**The European reference for
PCB prototypes & small series**
www.eurocircuits.com

Resultados mixtos

Koen Beckers (Estudiante en prácticas del laboratorio de Elektor) y Jesper Raemaekers

Un estudiante en prácticas: Koen Beckers nos cuenta algunas cosas sobre el proyecto que realizó junto con su compañero, el también estudiante Jesper Raemaekers, durante sus estudios de electrotécnica en el instituto Leeuwenborgh de Sittard (Países Bajos).

“El instituto nos pidió crear un proyecto que llevara a la práctica la teoría aprendida y con ello prepararnos para cosas que no aprendes con tan sólo teoría. Nosotros mismos pudimos elegir el tema. Como ambos trabajamos mucho con el sonido por ser esta nuestra afición, elegimos como tema un mezclador de audio de 4 canales.

Para el procesamiento de cada canal optamos por un LM1036. Es un integrado que lleva incorporadas todas las opciones de control estéreo de tono, balance y volumen. Esto sustituye la funcionalidad de un botón de electrónica analógica y además te permite controlar este integrado fácilmente con cuatro tensiones. Tres de estas tensiones se generan con la ayuda de tres potenciómetros conectados al regulador de tensión interno del LM1036. La cuarta tensión se obtiene de un DAC (MCP4921) con un operacional a continuación que adapta la tensión al nivel correcto. Esto es necesario porque la tensión de salida máxima del DAC es de 5 V y el LM1036 necesita una tensión de 5,4 V para que la atenuación sea la mínima.

Con una fila de LED controlada por un LM3914 conectado a la señal de salida del DAC, hacemos visible el nivel de dicha señal. La **figura superior** muestra los primeros croquis que dibujamos justo cuando iniciamos el proyecto.

Calculamos que necesitamos unos 100 LED y que la fuente de alimentación debe suministrar por lo menos 2 A (20 mA por LED). Si encargamos esto a un 78S05, su temperatura llegaría a subir mucho. Entonces el integrado disiparía por lo menos $(7V-5V) \times 2A = 4W$. Si la tensión del lado primario del regulador fuera 15 V – algo deseable para los operacionales – en vez de 7 V, ¡podría significar la muerte del integrado!

Para solventar este problema de calor podíamos haber optado por una fuente de alimentación conmutada. Pero no lo quisimos, porque el diseño de una buena fuente de alimentación conmutada para aplicaciones de audio estaba

fuera de nuestras posibilidades y probablemente una sencilla fuente conmutada ‘mala’ causaría demasiado ruido e interferencias.

A pesar de esto y en nuestra opinión, la mejor solución fue la utilización de un regulador lineal, asegurando que la corriente se quedara dentro de determinados límites.

Durante algunas pruebas deducimos que no íbamos a necesitar los 2 A, porque nunca iban a estar encendidos todos los LED a la vez. Por eso fue posible utilizar un 7805 estándar.

A medida que avanzamos con el diseño de la placa impresa (hecha en Cadsoft Eagle) y en que queríamos una placa compacta por cada canal, nos dimos cuenta de que nunca podríamos hacerla (el ataque de la placa virgen) nosotros mismos. Por eso pedimos permiso al instituto para hacerlo a través del Elektor PCB Service, y lo obtuvimos.

Cuando finalizamos el diseño de la placa, solicitamos construirla para después ensamblarla. Por desgracia, el circuito no funcionó a la primera. En el símbolo de Eagle de un

operacional que utilizamos no estaba claro lo que era Vcc y lo que era la masa. A causa de esto volamos los cuatro operacionales antes de que supiéramos lo que estaba pasando exactamente. Una conexión incorrecta de la alimentación en la placa

impresa no es precisamente lo primero en lo que piensas después de haber verificado la tensión de alimentación.

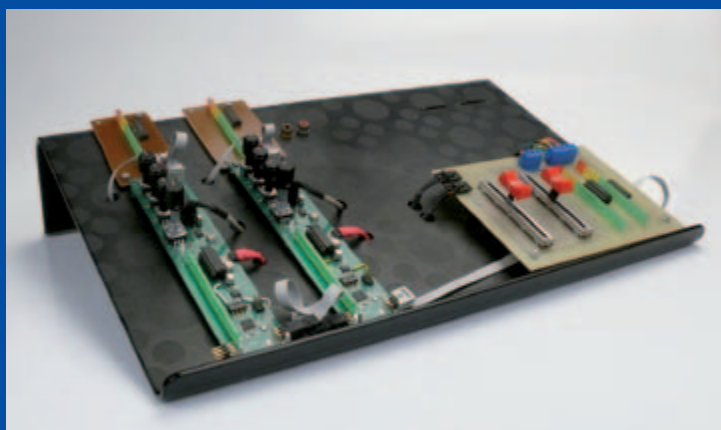
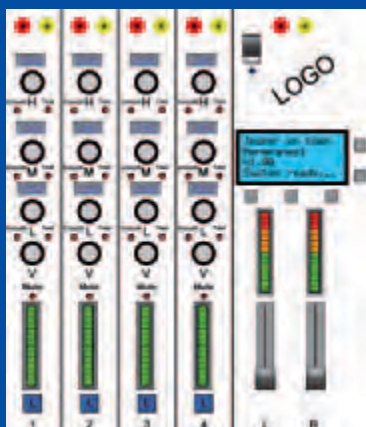
La **figura inferior** muestra el prototipo construido. La placa negra de plástico es un soporte barato de ordenador portátil que encontramos en IKEA y que sirvió perfectamente como base de nuestra instalación.

Inicialmente teníamos la idea de construir un mez-

clador de audio de 4 canales donde se hicieran todos los ajustes del LM1036 a través de un DAC y un codificador rotativo (con la representación del ajuste en el correspondiente display de 7 segmentos) y un potenciómetro deslizante motorizado para el volumen. Por desgracia no había lugar para esto en el presupuesto y tuvimos que reajustarlo hacia abajo con sólo un DAC y controlarlo a través de un codificador rotatorio con la ayuda de un ATmega88. Inicialmente obtuvimos del instituto un presupuesto de 100€. Lo sobrepasamos con creces a pesar de hacer frecuentes reajustes de nuestros planes.

Con todo fue una buena experiencia donde aprendimos alguna que otra cosa sobre electrónica y programación mientras trabajamos en algo que nos gustaba.”

(110601)



Detector de radiación verificado

Thijs Beckers (Redacción Holanda)

Con motivo del diseño del detector de radiación publicado en la edición de junio de este año ('Fotodiodo para medir rayos gamma', página 34 [1]), nos invitaron a comprobar nuestro sensor en el laboratorio de física nuclear de la universidad de Namur de Bélgica. El ingeniero de Investigación & Desarrollo Aurélien Nonet, nos había mandado un correo electrónico entusiasmado con el artículo donde se utilizaba un sensor barato (en la universidad se trabaja generalmente con sensores que junto con los accesorios rondan los 200.000€). Aurélien nos ofreció comprobar nuestro sensor con pulsos radioactivos calibrados y compararlo con sus detectores convencionales si fuera necesario. Así que me presenté junto con el autor Burkhard Kainka y su circuito de detección en el Laboratoire d'Analyses par Réactions Nucléaires [2]. Después de una calurosa bienvenida, disfrutamos de una amplia visita guiada por el laboratorio semienterrado – donde para mi sorpresa habían construido un completo acelerador de partículas de 2 GeV: el Altaïs (Accélérateur Linéaire Tandem pour l'Analyse et l'Implantation des Solides). Con este acelerador de partículas se disparan átomos a muestras para hacer pruebas del material. De esta forma se disparan, por ejemplo, átomos de nitrógeno a una muestra. Durante las colisiones se liberan varios tipos de radiación (principalmente rayos roentgen y gamma) que se miden con muchísima precisión. De ahí se obtiene la información del material disparado. Pero el acelerador también se utiliza para la investigación médica (la lucha contra el cáncer).

¡Y ahora a medir!

Después de la visita guiada se sacaron un número de sustancias radioactivas calibradas de la caja fuerte para ver lo que se podía hacer con nuestro (¡ridículo!) sensor. En primera instancia las pruebas no marchaban bien. Después de experimentar un poco, y ver diferentes muestras la cosa empezaba a funcionar. El contador del circuito aumentaba indicando que estaba midiendo radiación. La sensibilidad del sensor no resultó ser demasiado alta (lo que ya sabíamos). Nuestro circuito sólo reaccionaba

a muestras que generaban una energía suficientemente alta (expresado en electronvoltio, eV).

Había que proteger al sensor de cualquier forma de luz para la prueba de radiación. Una lata vacía de crema de manos sirvió perfectamente para eso. La placa, con el material radioactivo encima, se fijó con un trozo de goma moldeable y la tapa cerró para obtener un entorno de prueba completamente oscuro. Las primeras pruebas fueron positivas, aunque los sensores profesionales, con un coste de decenas de miles de euros, sean (por supuesto) mucho más sensibles.

Nuestro circuito se había equipado de una salida con toma coaxial, que ponía a disposición del mundo exterior las señales (amplificadas) del sensor. A través de esta toma coaxial pudimos conectar el circuito a los aparatos de medición del laboratorio de física nuclear. Ajustando y girando un poco los mandos (la tensión proporcionada por un sensor profesional es mucho mayor) pudimos generar una imagen en el PC. El software profesional que ahí se ejecutaba traducía las señales del sensor en valores legibles y gráficas donde se pudo observar las características del material radioactivo que estaba delante del sensor. ¡Todo el mundo estaba entusiasmado! ¡El sensor barato sí funcionaba!

Un ejemplo de resultados de la medición:

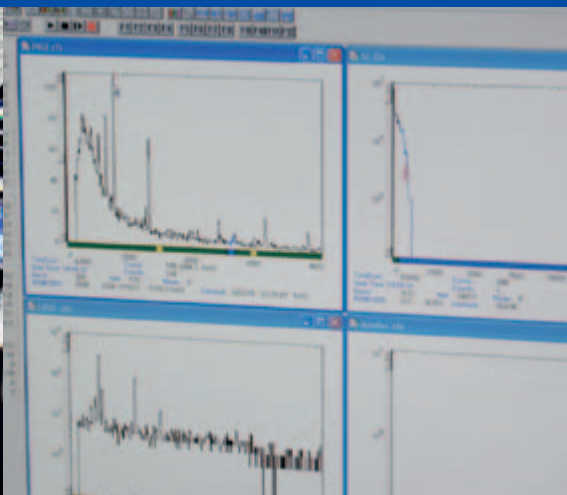
Medición de radiación gamma con un BPW34 como sensor.

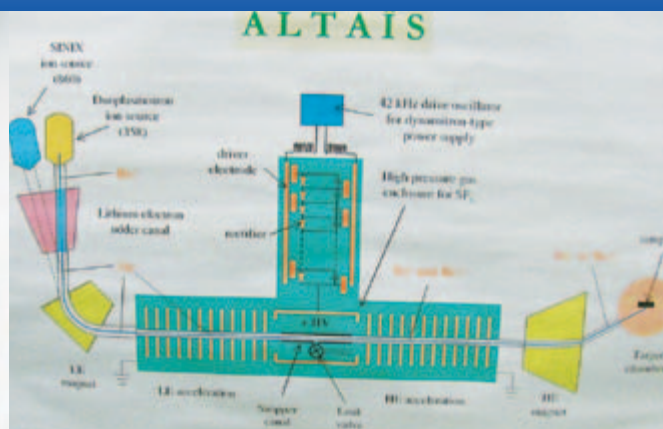
Muestra: 137 Cs (Cesio), 661 keV Gamma.

Resultado: Buena sensibilidad de la radiación gamma, traspasando también un encapsulado de aluminio. No surge un espectro de energía por que la unión PN es demasiado fina. La unión PN sólo absorbe una parte de la energía, el resto de la radiación sale por la parte trasera. Además de esto la dirección depende de la distancia recorrida en el sensor. Conclusión: El BPW34 sirve perfectamente para aplicaciones que contabilizan la radiación gamma.

Un segundo sensor

Como hubo muchas reacciones sobre el artículo publicado en junio, Burkhard siguió con sus experimentos y trajo un segundo tipo de sensor. Estaba basado en el BPX61. Con este sensor con-





siguió medir también la radiación alfa más débil. Nuestra visita a la universidad fue el momento perfecto para verificarlo con aparatos profesionales. Algunos resultados:

Medición alfa con el BPX61

Muestra: 244 Cm (Curio), 5,8 MeV Alfa.

Se colocó la muestra radioactiva junto al sensor en un encapsulado opaco. La amplitud del impulso registrada es una buena medida de la energía de las partículas que alcanzaba el sensor. Con esto el BPX61 es un detector alfa completo. Las pruebas con diferentes distancias entre sensor y muestra demostraban que cuando la distancia aumentaba hasta 5 cm la energía se reducía hasta 0.

Muestra: 239 Pu (Plutonio), 5 MeV

La muestra generaba 210 mV en la salida del amplificador de medición. A una distancia de 1 cm entre muestra y sensor generaba todavía 190 mV.

Conclusión: El BPX61 no solo es apto para aplicaciones de contabilización, sino también como espectroscopia alfa.

¿Medir radiación beta?

Lamentablemente no hemos podido contestar de forma unívoca a la pregunta de si se puede medir también radiación beta con sensores baratos. Normalmente el laboratorio no se dedica a esta forma de radiación y las muestras radioactivas disponibles generan relativamente poca radiación beta. No hemos podido detectar señales que procedan claramente de radiación beta, de modo que esta pregunta queda sin contestar.

Si tienes una idea o propuesta (o dispones de un laboratorio completo), ¡entonces no dudes en ponerte en contacto con nosotros! Para eso puedes utilizar la dirección de la redacción: redaccion@elektor.es.

Hacerlo uno mismo

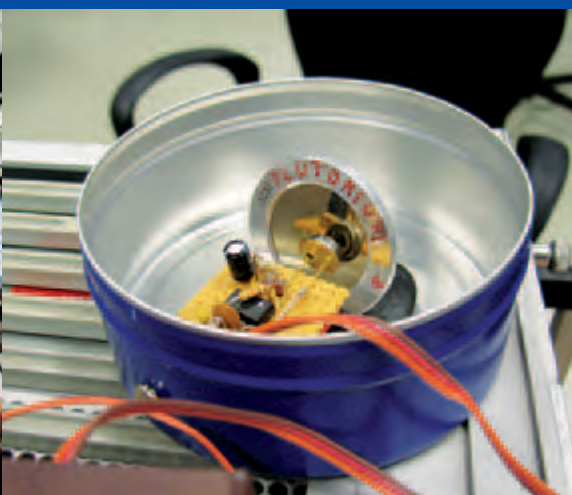
Si quieres empezar y no dispones de un laboratorio o de suficiente capacidad financiera para justificar un viaje a Japón, un reloj antiguo con agujas luminosas es, entonces, una buena alternativa. Incluso cuando las agujas ya no se iluminan, libera una cantidad más que suficiente de radiación. Un comentario importante: para detectar la radiación alfa hay que hacer un agujero en el cristal ya que dicha radiación es tan débil que el cristal la bloquea.

Por otra parte, tenemos la intención de publicar una versión mejorada del detector de radiación, con su placa impresa y una amplia descripción de los sensores. Estate atento a la revista en los próximos meses...

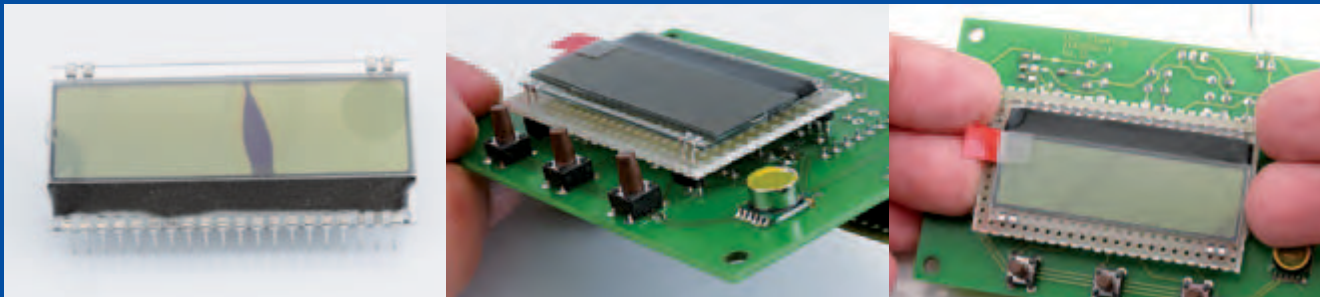
(110605)

Weblinks:

- [1] www.elektor.nl/110372
- [2] www.fundp.ac.be/en/sci/physics/larn/page_view/presentation.html



Útil, este display



Luc Lemmens y Thijs Beckers (Laboratorio Elektor/Redacción NL)

En el laboratorio reutilizamos muchas veces el mismo tipo de display para los proyectos de prototipos que requieren de uno. Y es que hay unos displays estándares que se utilizan con mucha frecuencia y sería tontería pedir al almacén uno nuevo para cada proyecto, mientras hay un montón en el armario de prototipos cogiendo polvo.

En fin, hay que reutilizar los displays. Esto no debe suponer ningún problema mientras el display este montado sobre una placa impresa decente (y fuerte) que tenga un conector. Montarle un conector es lo primero que hacemos en caso de recibir uno nuevo del almacén.

Sin embargo, hay displays donde no es tan fácil hacer esto, por ejemplo en los displays DOGM de Electronic Assemblies. Constata de un cristal con unos terminales finos para las conexiones eléctricas necesarias. De modo que no se trata de algo muy fuerte y no sirve para ser reutilizado. Hemos roto ya unos cuan-

tos intentando quitarlos de la placa impresa (ver foto) y seguramente no seamos los únicos.

Pero ahora tenemos una solución, aunque quizás suene un poco sencilla, lo importante es la idea: coloca una placa de pruebas entre el display y la placa (madre). No hace falta soldar el display sobre la placa de pruebas. Simplemente puedes traspasar los terminales del display por los agujeros de la placa de pruebas y asegurarte de que haya una buena conexión con la placa principal, generalmente los terminales tienen suficiente longitud para hacerlo. Cuando quieras quitar el display, sujeta la placa de pruebas en el frágil cristal del display. La placa de pruebas distribuye la fuerza y el display queda intacto. Es así de simple... También es posible soldar el display sobre la placa de pruebas. Y si fuera necesario puedes soldar también unos conectores decentes sobre la placa impresa. Si además haces coincidir las conexiones de los conectores con el diseño 'normal' de los terminales de un LCD, habrás obtenido otro módulo display universal.

(110506)

Bits rebeldes

Raymond Vermeulen (laboratorio de Elektor)

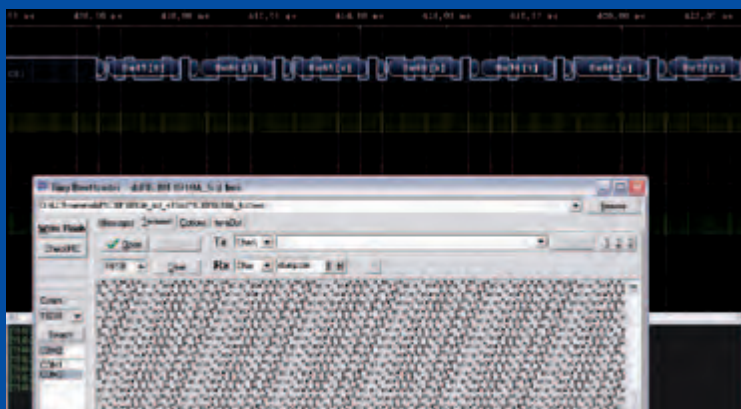
Algo extraño ocurrió durante las pruebas de un proyecto (ver imagen). El terminal mostraba basura, pero el analizador lógico mostraba correctamente 'Elektor' en ASCII. Este último indicaba también que el UART funcionaba a 4800 baudios en vez de los 19200 baudios que había programado – eso pensaba yo al menos –, por lo tanto, un factor cuatro de diferencia. El cambio que previamente había introducido en el código era la cuadruplicación de la frecuencia de reloj del dsPIC. La conclusión a la que llegué fue que no se había cambiado la frecuencia de reloj. ¿Pero por qué?

Tuve una ocurrencia, como no podía ser de otra forma, debajo de la ducha. En un proyecto de aficionado utilicé un Atmega32u4 con un cargador de arranque con la única limitación de que este no pudiera programar los bits de fusibles. “No será que...”, pensaba. Y efectivamente, el cargador de arranque que utiliza el dsPIC no puede programar bits de configuración. Por supuesto,

que los programadores experimentados lo saben de sobra, pero todo el mundo tiene alguna vez días malos...

(La solución fue la utilización de un programador de 'verdad', como es el ICD3).

(110613)



Inversor de Onda Senoidal con Corrección de Factor de Potencia

100 vatios, 24 VDC a 230 VACrms de señal senoidal pura de salida

Michael Kiwanuka (Reino Unido)

Los inversores de potencia son usados para generar líneas de alimentación de corriente alterna con tensiones de, por ejemplo, 230 VAC o 115 VAC en el campo, usando baterías de coche de alta capacidad de 12 V o 24 V. Estos equipos se suministran en una amplia variedad de potencias de salida (cualquier valor entre 15 y 1.000 vatios) y cualquier calidad de la tensión de salida AC (cualquier valor entre algo abominable y una señal senoidal pura). Algunos modelos hasta disponen de regulación de tensión de salida. Sin embargo, pocos combinan una corrección del factor de potencia (PFC) con 'una onda senoidal pura de salida'. Este es el motivo por el que este diseño sugerido aparece en este artículo, junto con un ligero fondo teórico.

En el circuito

En cuanto al esquema eléctrico del circuito de la **Figura 1**, IC2 y el cristal de cuarzo Q1, forman un reloj de alta frecuencia de 4,096 MHz. El divisor binario de 14 etapas, un CD4060, junto con IC3, un contador Johnson CD4017, divide la señal de frecuen-

cia básica en una onda cuadrada de 50 Hz. Para una salida de 60 Hz, debemos cambiar la frecuencia del cristal de cuarzo a un valor adecuado.

A continuación, la señal de reloj es procesada por IC5 e IC6.A, los cuales, junto con sus redes asociadas, forman una unidad de arranque suave. IC6.B, junto con su red con filtro doble en T, seleccionan la frecuencia fundamental de 50 Hz (o 60 Hz) que, por análisis de Fourier, es una onda senoidal pura.

El siguiente componente de la cadena es IC6.C, que actúa como disparador Schmitt transformando la señal unipolar básica de alta frecuencia en una onda cuadrada simétrica. Esta señal es integrada y mezclada con la señal senoidal de 50 Hz en el filtro "notch" formado por IC6.D.

IC13 y sus componentes asociados, forman un comparador que compara una señal DC con la salida del comparador para dar una señal de ancho de pulso modulado que,



Corrección de Factor de Potencia y la bombilla Osram

El factor de potencia de una bombilla de ahorro de energía Osram típica tiene un rango capacitivo de 0,6 (por ejemplo, encendida). Para una bombilla de 21 W, el factor de potencia (I/P) es $S1 = 21/0,6 = 35 \text{ W}$. El valor 'VA' de Q1 se puede calcular de forma vectorial como:

$$Q1 = \sqrt{(35^2 - 21^2)} = 28 \text{ VA}$$

Dejando un factor de potencia de = 0.9 después de la corrección, tendremos

$$S2 = 21 / 0,9 = 23,3 \text{ VA}$$

$$Q2 = \sqrt{(23,3^2 - 21^2)} = 10,1 \text{ VA}$$

De donde Qa (cantidad de potencia ser corregida) = 17,9 VA, en carga

Ahora, para calcular la inductancia del secundario del transformador tendremos:

$$L_s = V^2 / (2\pi \times 50 \times 17,9) = 9,45 \text{ H}$$

Para las medidas, $L_p = 12,32 \text{ mH}$.

$L_s = n^2 L_p = 9,49 \text{ H}$, para un n de 28, como se deduce de los conocimientos del número de vueltas para el transformador. De donde el transformador del inversor proporciona una corrección de factor de potencia de 0,9.

Nota. Los Proyectos de Lectores son reproducidos en base a la información suministrada únicamente por el(os) autor(es). La utilización de los estilos de Elektor en esquemas de diseño y en otras ilustraciones, o la disponibilidad de la descarga (software) del proyecto desde la página web de Elektor, no implican que el proyecto haya pasado por los Laboratorios de Elektor para verificar su funcionamiento.

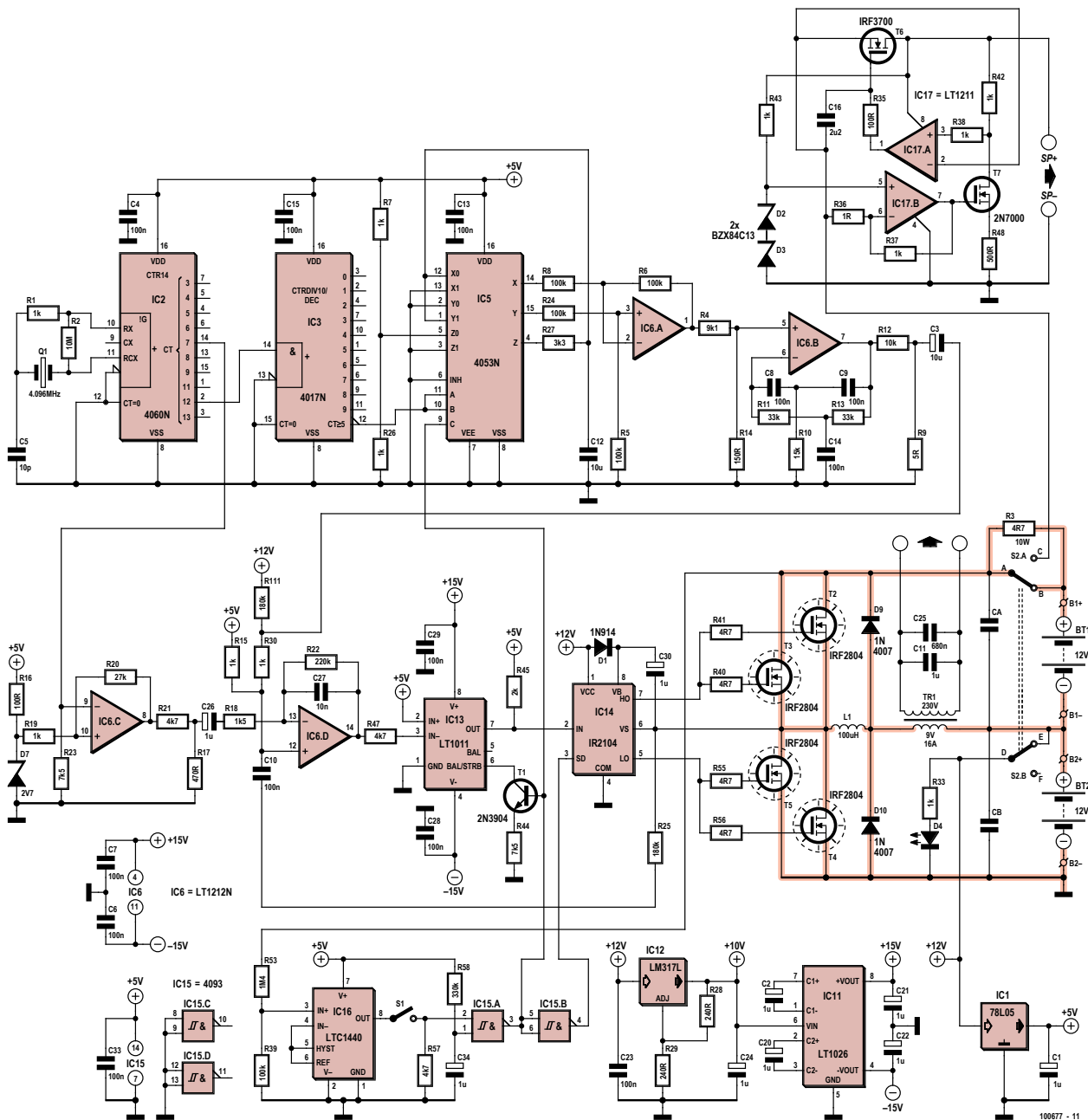


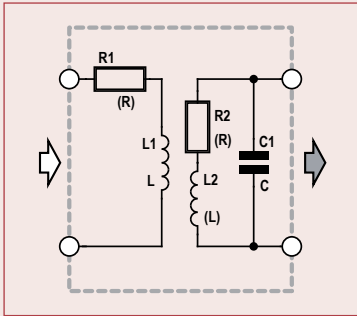
Figura 1. El esquema eléctrico del circuito del inversor de potencia es algo más elaborado, debido el uso de circuitos integrados y transistores estándar (de Farnell) que los circuitos integrados (CI) conversores dedicados a ello.

a su vez, controla a IC14, un controlador de semi-puente del tipo IR2104. El propio semi-puente está formado por los transistores FET de potencia T2-T3 y T4-T5, siendo este último opcional para cuando se requieran potencias de salida más elevadas.

La topología de semi-puente controla el primario de un transformador de potencia de 9 V y 16 A, que proporciona una tensión de salida de 230 V (o 115 V) rms, con un rango de potencia de 100 VA. Se inserta una inductancia de 100 μ H (Bourns 23000L; para mon-

taje vertical) en serie con el primario. Los condensadores C11 y C25 están conectados en los extremos del secundario del transformador para filtrar la alta frecuencia de conmutación. Hay que señalar que llamamos "bobinado primario" a la parte de 9 V

El filtro de salida



El filtro de salida está configurado para dejar pasar una banda de frecuencias más allá de la cual la señal se atenúa fuertemente. Esto se consigue conectando un condensador entre los extremos del bobinado secundario, como se muestra en el esquema eléctrico del circuito.

Podemos ver que la impedancia de entrada vista desde los terminales del primario viene dada por la fórmula:

$$Z_1 = R_1 + jX_1 + Z_2$$

$$Z_1 = R_1 + jX_1 + \frac{(\omega M)^2 R_2}{R_2^2 + X_2^2} - j \frac{(\omega M)^2 X_2}{R_2^2 + X_2^2}$$

Con el secundario resonando a la frecuencia ω_0 tenemos:

$$Z_1 = R_1 = jX_1 + \frac{(\omega_0 M)^2}{R_2}$$

Así, justo por encima de ω_0 , I_{pr} es proporcional a $1/\omega$. Ahora, consideremos a ω mucho mayor que ω_0

$$Z_1 = R_1 + j \frac{X_1 - (\omega M)^2 X_2^2}{R_2 + X_2} + \frac{(\omega M)^2 R_2}{R_2^2 + X_2^2}$$

En una segunda resonancia,

$$X_1 - \frac{(\omega M)^2 X_2}{R_2^2 + X_2^2} = 0$$

$$X_2^2 - \frac{(\omega M)^2}{X_1^2} X_2 + R_2^2 = 0$$

$$X_2 = \frac{(\omega M)^2}{2X_1} \pm \frac{1}{2} \sqrt{\frac{(\omega^2 M^2)^2}{X_1^2} - 4R_2^2}$$

Pero $X_1 = X_2(N_1/N_2)^2$ y, si recordamos la expresión para Q , así como la expresión de $L_1/L_2 = (N_1/N_2)^2$, tenemos que, para $Q \gg 1$:

$$X_2 = k^2 R_2 Q$$

Para una transformación de serie a paralelo tenemos que $R_2 = R_p/Q^2$, donde R_p es la resistencia del primario y R_2 la del secundario.

Bien $X_2 = 0$,

$$o \quad X_2 = \frac{R_p k^2}{Q}$$

$$o \quad X_2 = k^2 R_2 Q$$

Así, en una segunda transición, sustituyendo X_2 y remarcando que la parte imaginaria es cero,

$$Z_1 = R_1 + \frac{(\omega M)^2}{R_2 \times (1 + k^4 Q^2)}$$

Por lo tanto, I_{pr} es, aproximadamente, proporcional a $1/\omega^2$ para $\omega \gg \omega_0$, algo que es muy deseable.

y “bobinado secundario” a la de 230 V (115 V) y no al revés, como es lo habitual con los transformadores AC de tensión de red. La inductancia del secundario del transformador se duplica, al igual que el elemento de corrección de factor de potencia (PFC), cuando trabajamos con bombillas de ahorro de energía, las cuales tienden a tener un factor de potencia importante (ver **apartado correspondiente**).

En lo referente a los componentes adicionales que se pueden ver en el circuito, IC16 es utilizado como una red de detección de nivel para las baterías. IC17A, IC17B y los FET T6-T7 forman un cargador de baterías con una tensión de salida de 25–26 V. IC11 e IC12 forman una fuente de alimentación simétrica de ± 15 V para los amplificadores operacionales, mientras que IC1 es un regulador estándar de 5 V para los circuitos lógicos. Las puertas “trigger Schmitt” IC15A e IC15B forman

la señal de reinicio del encendido (POR). Los condensadores CA y CB son componentes “externos” a la placa debido a su tamaño. Su valor se obtiene de manera experimental. El prototipo del inversor funcionaba mejor usando dos condensadores electrolíticos Evox/Rifa de 22.000 μ F, de 25 V. El aluminio permite montar condensadores de tuerca. Las líneas más gruesas del esquema eléctrico indican las pistas, los puntos de montaje (“pads”) y los conectores adecuados para transportar las corrientes continuas (DC) de hasta 32 A.

Temas prácticos

La foto de la introducción muestra un primer prototipo del inversor que ha tenido desarrollos posteriores con el objetivo de llevarlo a una producción real. Los transistores FET de potencia del puente se montan sobre un radiador del tipo SFP100, que actúa como

tapa superior de la caja. Las dos tuercas que se ven en el panel lateral sirven para fijar los grandes condensadores electrolíticos CA y CB. El autor del artículo ha diseñado una placa de circuito impreso (PCI o PCB) cuyos ficheros de trabajo (serigrafía y pistas de cobre) pueden descargarse de [1], junto con los datos de taladrado para una caja del tipo KOH10-KOR2-160-ME-SL, de Fischer Elektronik. Las pistas de la PCB que soportan altas corrientes (ver esquema eléctrico del circuito) deben ser reforzadas soldando un hilo de cobre macizo de 1,5 mm² c.s.a. (aprox. 16 AWG) a lo largo de su recorrido.

(100677)

Enlaces en Internet

- [1] www.elektor.es/100677 (diseño de la PCB y datos de taladrado de la caja)

Placa polivalente para circuitos

Platino, la gran estrella de los segundos papeles

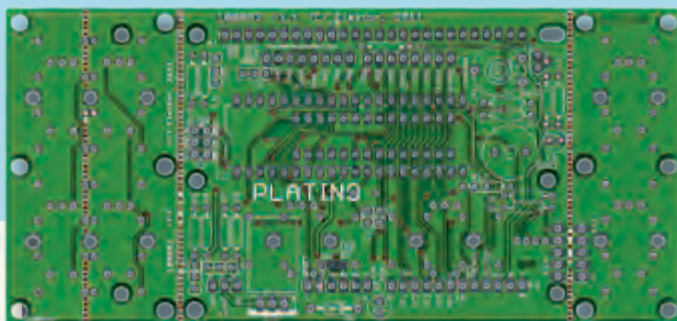


Grégory Ester y Clemens Valens (Francia)

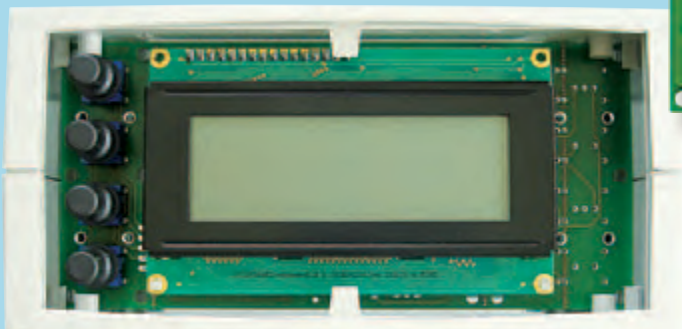
En los proyectos electrónicos, la placa de acogida a menudo juega sólo un papel de segundo plano. Sin ella es netamente más difícil poner en funcionamiento la realización de un montaje pero, después de todo, ¿quién se acuerda verdaderamente de la placa en la que descansa todo? Para corregir esta injusticia, decidimos darle el papel principal de este artículo a una placa. ¡Señoras y señores, gracias por esos aplausos a... Platino!

Biografía

Platino, nacido el 20 de noviembre de 2010 en los Países Bajos, es un circuito impreso polivalente para montajes basados en el microcontrolador AVR de 8 bits. Hermano pequeño del J²B [2], recibirá después de largos meses de preparación, a su primer microcontrolador AVR en junio de 2011. No completamente listo para enfrentarse a la jungla de la electrónica, decide



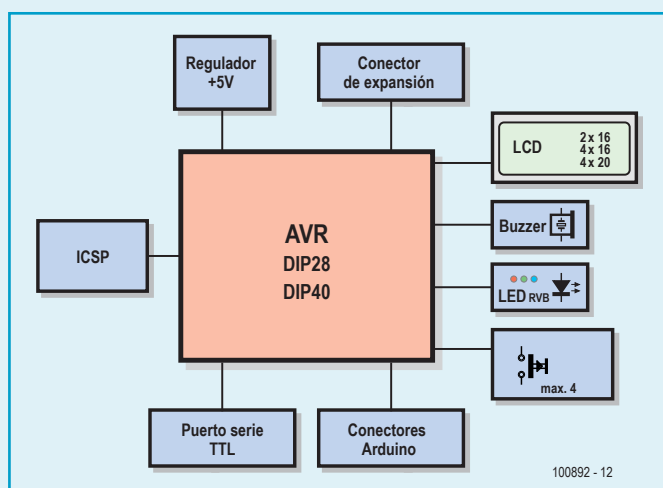
perfeccionarse tomando ejemplo de su ídolo, Arduino. Libre y despreocupado, su existencia es puesta "patas arriba" tras un encuentro fortuito con una caja Bopla. Flechazo inmediato, él le pide casarse con sus formas. Su unión ha sido celebrada en julio de 2011 y, preocuparos, quieren a muchos niños.



con microcontroladores AVR

Coqueta y con tendencia

La época en la que se vestía cada día del mismo modo, como lo hacen el pato Donald o Tintin, ha pasado ya hace más de un siglo, y Platino lo sabe muy bien. Una placa moderna se “viste” según las ganas y la ocasión. Cuando el tiempo es desapacible, Platino prefiere salir equipada con una gran pantalla LCD de 4 líneas de 20 caracteres, pero tan pronto lo permite la ocasión, Platino se equipa con una pequeña pantalla LCD negra, de 2 líneas de 16 caracteres. ¡Que esto desvele sus bellas pistas no le molesta, al contrario, Platino no es púdica!



Por más que se sea una bella placa, si la agudeza de espíritu y la sutileza de los sentidos no te acompañan, la gente se cansa rápido. Así pues, Platino se ha visto obligada a escoger a un microcontrolador. Como sus padres le han dicho siempre que no se deben dejar nunca las cosas a la mitad, Platino ha llevado este principio al extremo y ha decidido aceptar todos los micros AVR de 28 o de 40 terminales, del prestigioso diseñador Atmel, lo que corresponde, de hecho, con todos los micros AVR de 8 bits con encapsulado DIP.

¡Por una vez, el hábito hará al monje y serán los accesorios los que completarán ventajosamente el comportamiento, ya muy avanzado, de Platino, a la que le encanta disfrazarse! Habitualmente, Platino saca su colección de pulsadores, colocando a veces hasta cuatro, a la izquierda, a la derecha o debajo de su pantalla. Platino ama a sus pulsadores, ya que los puede equipar con tapas de longitud, color y formas diferentes. Es verdad que son un poco más caros, pero cuando se trata de ampliar sus posibilidades, Platino no tiene en cuenta sus gastos. Otro accesorio favorito de Platino es el codificador rotativo que puede poner en lugar de dos (incluso tres) pulsadores. ¡Puede implantarse incluso dos a la vez!

Platino adora ser el centro de interés. Así, ha invertido en un

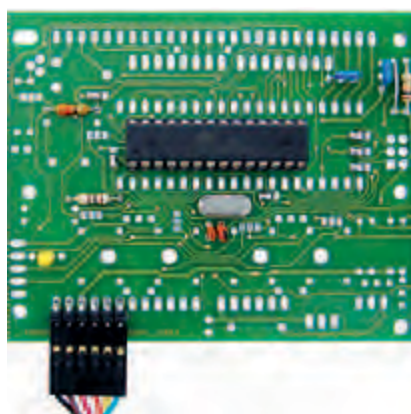


zumbador que le permite atraer todas las miradas cuando quiere dejarse oír. Para evitar romper su voz, lo utiliza siempre con una resistencia limitadora de corriente (R2).

En respuesta a una crisis precoz de los cuarenta, Platino se ha equipado con un LED tricolor. Es capaz de deslumbrarnos emitiendo una luz perfectamente blanca, adaptando los valores de las resistencias R1, R8 y R9 al modelo de LED RVB escogido. ¡También puede ser totalmente verde fogoso o rojo cólera, ¡Platino tiene su carácter!

Imitadora

Desde pequeña, Platino admiraba a Arduino, la estrella italiana del “prototipado” rápido a la que estudiaba bajo todas las perspectivas. Impresionada sin llegar a quedar cegada, Platino decide imitar los lados buenos de su ídolo y mejorar ciertas debilidades. Así, se equipa con conectores de extensión (K4 a K7), que le permiten hacer gala de los mismos caparazones (“shield”), a semejanza de su modelo, pero también se equipa de conectores de extensión con un paso más convencional (K1, K2, K9).



A diferencia de Arduino, y como considera como ocasionales los enlaces serie, Platino puede vivir sin adaptador USB/TTL. Según Platino, siempre podrá recurrir a un cable FTDI externo, si lo necesita. ¿Arrogancia juvenil? Tal vez, pero también menos caro. ¡Pues sí que es maligna esta Platino! Por otra parte, este cable servirá para programar su micro desde el IDE Arduino, ya que Platino lo hace todo exactamente igual que Arduino. Ella asegura el reinicio de su micro desde el IDE sin botón-pulsador de reset, gracias a la resistencia R13, que, cosa sorprendente, puede también ser un

condensador de 100 nF (Arduino nunca se expresó claramente sobre este tema, por lo que Platino puede trabajar con las dos opciones). Por supuesto, todo esto funciona sólo si el micro de Platino ha sido programado previamente con un cargador de arranque compatible Arduino y una versión reciente del IDE (actualmente v0022) [3]. Es K3 quien permite esto, gracias a un programador AVR estándar.

Arduino ha sido criticado a menudo por sus capacidades, a veces demasiado limitadas. Por ello Platino ha decidido no fijarse en un solo modelo de microcontrolador. Con un ATmega168 o un ATmega324, Platino puede jugar el papel principal en un gran número de situaciones. Y para los realizadores más exigentes, Platino dispone de un as muy convincente: ¡puede ser equipado con un ATmega1284!

Incluso si Platino es capaz de hacer gala de un LED tricolor, puede también contentarse con un simple diodo LED como su ídolo. Para ello, conecta sencillamente R8 a PB5 con la ayuda de JP14. Así, el LED parpadea cuando se carga su micro por el puerto serie. El ejemplo Blink, incluido en el IDE Arduino, también funciona sin modificación.

(100892)

El guardarropa de Platino

Equiparse correctamente es importante para Platino. He aquí lo que ella cuenta sobre sus componentes preferidos:

Resistencias

« Las resistencias son importantes en la medida en que ellas me permiten limitar las corrientes o aplicar tensiones. Me gusta elegir las de 47 Ω para R2 y R3, una de 100 Ω para R13, una de 4,7 k Ω para R11 y un poco por facilidad, las de 10 k Ω para R4 a R7, R10 y R12. R1, R8 y R9 son más difíciles, ya que hay que conectarlas a un diodo LED tricolor. Para ello, las de 470 Ω son siempre un buen compromiso.»

« Utilizo a P1, mi resistencia ajustable horizontal, para ajustar el contraste de mi visualizador. Tengo uno de 10 k Ω .»

Condensadores

« A menudo los condensadores son un poco olvidados, pero yo les concedo, a pesar de

todo, una cierta atención. Para C1 y C2, que acompañan al cristal de cuarzo de 16 MHz, me gusta utilizar mucho los de 22 pF. Las perturbaciones de RF me irritan, por lo que le llevo siempre los condensadores C3 a C6. Generalmente, 100 nF es una buena elección, pero para C4 prefiero escoger uno de 10 nF. Esto me ayuda a ampliar la gama de las frecuencias atenuadas. Su paso no tiene mucha importancia, 2,5 mm o 5 mm, para mí es igual. Si pongo a IC3, tengo que poner también a C8 y C9. C8 asegura la estabilidad de IC3 y 1 μ F hace el apaño. C9 es un poco la misma cosa, pero a menudo se aconseja elegir un valor diez veces más grande. Así pues, uso uno de 10 μ F. Dos deben soportar, como mínimo, tensiones de 16 V, C9, a veces, incluso más. Su paso debe ser de 2,5 mm.»

Inductancias

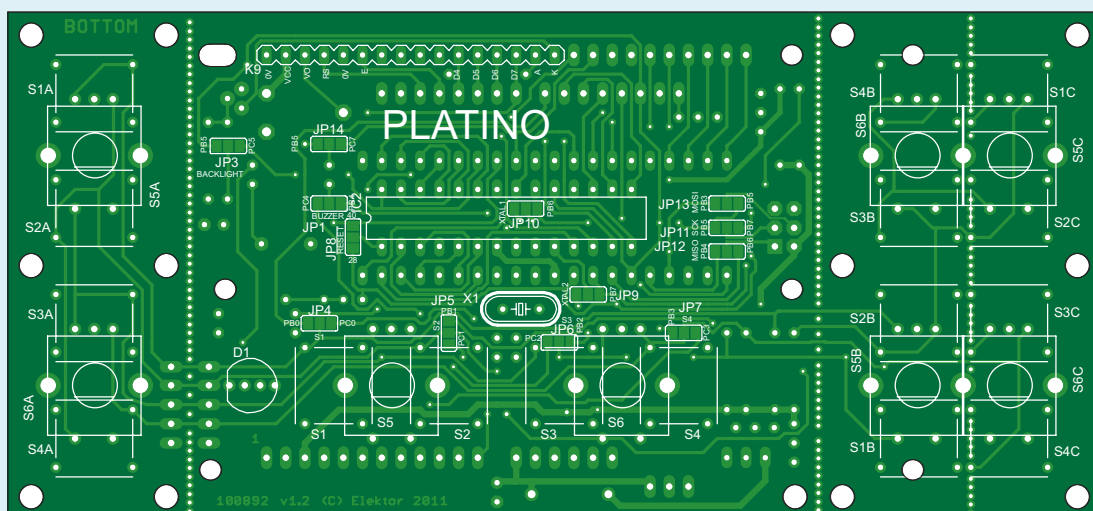
« Tengo sólo una: L1. Ella me sirve para mantener limpia la tensión de alimentación del convertor analógico/digital del microcontrolador. C5 la

mantiene en esta tarea. Me ofrecieron una de 10 μ H y la mantengo, incluso aunque un puente en hilo funciona también en la mayoría de los casos.»

Semiconductores

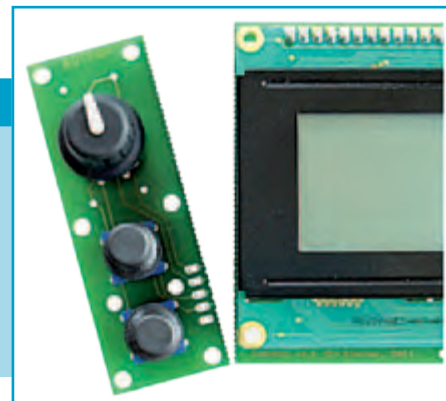
« Nunca pongo a IC1 e IC2 juntos, aunque puede ser que sea posible, habría que intentarlo... Para las salidas habituales, prefiero utilizar a IC2, es más fácil de encontrar y menos caro. Así, tengo la posibilidad de elegir entre un ATmega48, un ATmega88, un ATmega168 o un ATmega328 que monto sobre mi cara de soldadura. IC1, que hay que montar por la cara de componentes, puede ser un ATmega164, un ATmega324, un ATmega644 y hasta un ATmega1284, que sirve más bien para las grandes ocasiones. Utilizo sólo los modelos que soportan un reloj de 20 MHz.»

« Cuando necesito controlar la retroiluminación de mi visualizador LCD, monto a T1. Verdaderamente no tengo ninguna preferencia, siempre y cuando este transistor NPN pueda



Infiel

Platino ama su espaciosa cajita Bopla sin, por ello, privarse de los servicios de otras cajas. Platino tiene un corazón de alcachofa y, por otra parte, posee un gran número de agujeros de fijación, pero Platino no queda por ello menos desprotegida, se adapta pues voluntariamente y fácilmente sobre otras partes, desembarazándose de unas protuberancias desmontables, siguiendo las zonas punteadas. Gracias a sus conectores K10 y K11, ¡tiene incluso la posibilidad de aprovecharse de sus botones-pulsadores a distancia!



conmutar unas centenas de mA. Un BC547C va muy bien. Puedo también poner a un puente en lugar de T1 (colector/emisor), el paso es idéntico.»

« A veces no tengo demasiada confianza en la alimentación externa de +5 V. En ese caso me sirvo de IC3, un regulador de tensión de +5 V. No hay necesidad de hacerlo difícil, un viejo 7805 me viene perfectamente, incluso si hoy día ya se puede hacer mejor. También pongo a D2, un 1N5817 si la tensión de entrada no es muy elevada o un 1N4001 cuando dicha tensión está por encima de, digamos, 9 V. Con D2 e IC3 soporto sin problemas tensiones de entrada de hasta 18 V y estoy protegido contra las inversiones de polaridad. Acepto también ser alimentado con un cable adaptador USB/TTL. En ese caso, no me alimento por Vin, ya que si no, IC3 podría causar problemas. Los frioleros probablemente prefieren no implantar el regulador en esta configuración.»

« D1 es mi LED tricolor de cuatro terminales. Kingbright hace modelos que me van muy bien,

como la serie L-154A4SURKQBxxxx. COM-09264 de Sparkfun es válido también.»

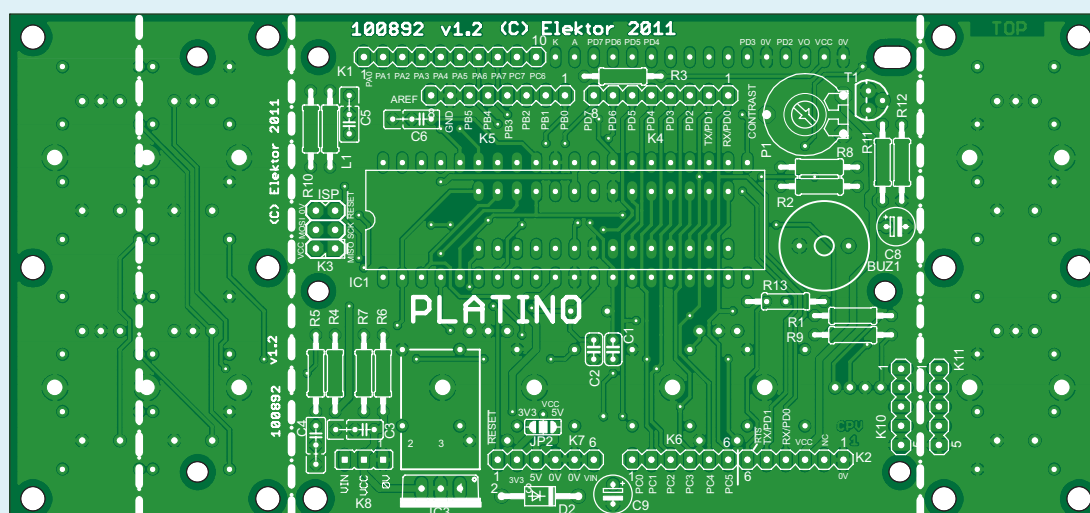
Varios

« Mi cristal de cuarzo, X1, es un modelo estándar de 16 MHz, que me lo coloco en mi cara de soldaduras para evitar el contacto con ciertas pistas. Mi zumbador BUZ1, con un diámetro de 12 mm, tiene un paso de 6,5 mm. En la parte de conectores, tengo la elección. Son barras divisibles con un paso de 2,54 mm y de una sólo fila, excepto para K3, que es un modelo macho de 2 x 3 contactos. K4 a K7 son modelos hembras que deberán permitir, llegado el caso, acoger un shield. K1 y K9 pueden ser macho o hembras, tulipas o espadines, todo depende de la aplicación. K1 posee 10 contacto, K4 y K5 tienen 8, K6 y K7 tienen 6 y K9 posee 16. K2 es un modelo macho de 6 contactos (me gusta acodado, es práctico) y K8, macho también, tiene sólo 3. Finalmente, uso a veces zócalos para IC1 o IC2. Para IC2 hace falta un soporte DIP de 28

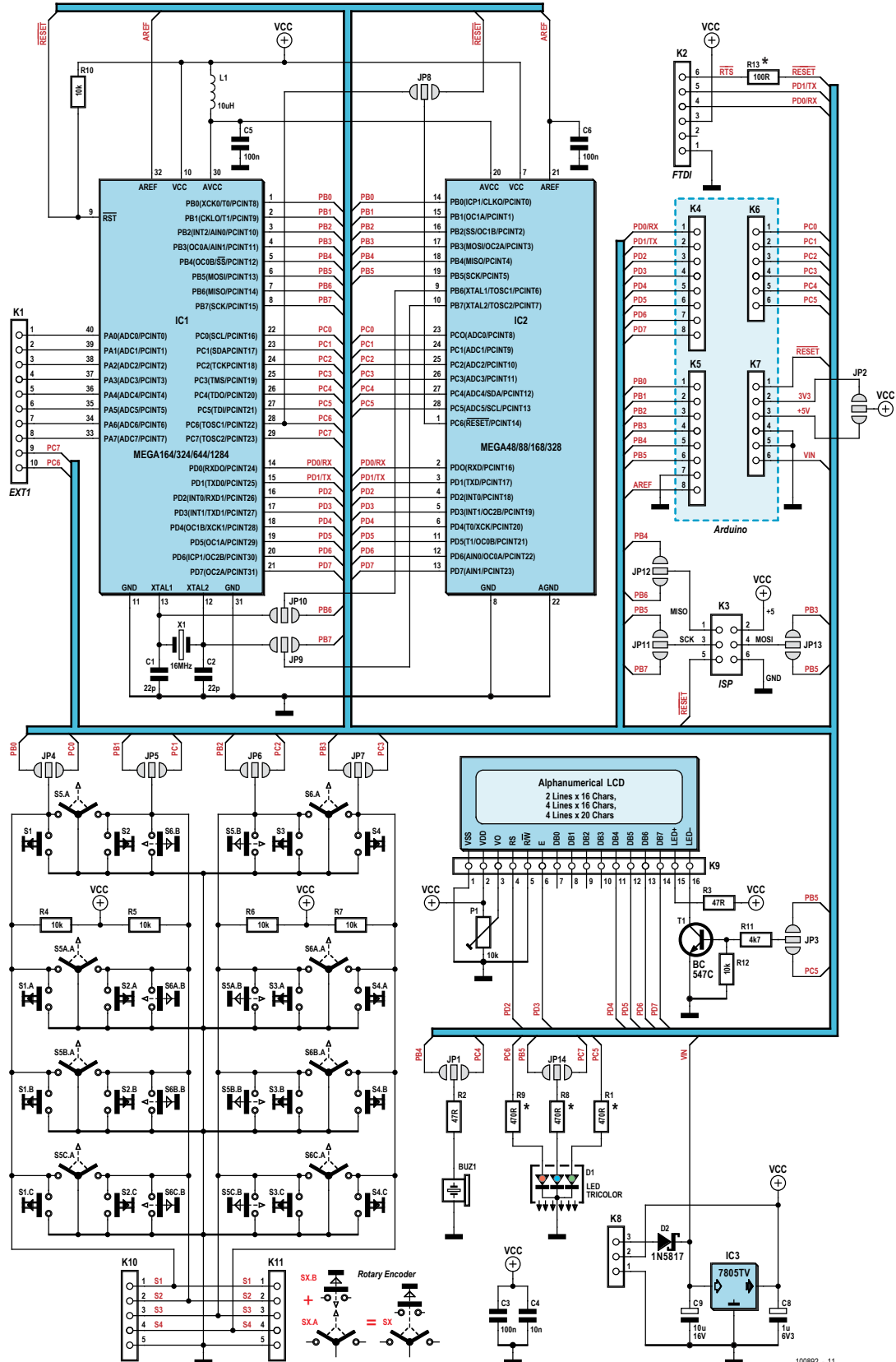
terminales, con una anchura de solamente 7,62 mm, y para IC1 un zócalo DIP estándar de 40 terminales.»

« ¡Ah sí, mis botones-pulsadores! Pues bien, son los 3FTL6 de Multimec, disponibles en Farnell. Mis codificadores rotativos son los Alps EC12E2424407 (con botón-pulsador) o los EC12E2420404 (sin botón-pulsador), también en Farnell.»

« Tengo varios visualizadores de diferente tamaño, 2 x 16, 4 x 16 o 4 x 20. Me van bien casi todos, con la condición de que tengan un conector de 14 ó 16 contactos en línea, arriba y a la izquierda de su pantalla y que sus terminales estén distribuidos en el mismo orden que sobre el esquema. El controlador integrado en el LCD deberá ser compatible con las librerías de gestión del visualizador que utilizamos, un HD44780 es un estándar.»



El segundo papel de este artículo lo interpreta el esquema eléctrico de Platino.



¡Platino pronto en rodaje en element14!

La familia

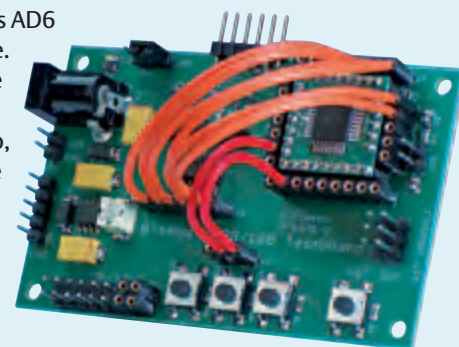
Platino no niega sus orígenes y muestra orgullosamente su pertenencia a la descendencia ATM18 BASCOM-

AVR/AVR-GCC. Cuando Platino acoge un micro de 28 terminales, no dispone de las señales AD6

y AD7 de su primo pequeño ATM18 [4] y, por lo tanto, no es completamente compatible.

Pero si estas señales se muestran indispensables para una ocasión especial, basta con que Platino reemplace su micro por un modelo de 40 terminales. Probablemente habrá que reconfigurar algunos puertos o desplazar a algunos hilos conductores, ya que este micro, más grande, ofrece un puerto A que el micro pequeño no tiene, aunque la situación puede ser salvada.

En homenaje a una rama alejada la gran familia de placas de desarrollo de microcontroladores AVR, Platino ha procurado que el controlador de su visualizador LCD sea compatible con las librerías de programación de Mikroelektronika, que utilizan por defecto el puerto D en el modo de 4 bits.



A pesar de algunas extravagancias, Platino queda como una placa accesible, cómoda y modesta que desea gustar a un gran público. Es por esto por lo que se la ama.

Enlaces en internet

[1] Platino: www.elektor.es/100892

[2] J²B: www.elektor.es/110274

[3] Arduino: www.elektor.es/080931

[4] ATM18: www.elektor.es/atm18

Función	Puente	Mi elección
El zumbador BUZ1 sobre PB4 o PC4	JP1	<input type="checkbox"/> PC4 <input type="checkbox"/> PB4
Selección de la tensión de alimentación de un <i>shield</i> Arduino: 5V o 3,3V.	JP2	<input type="checkbox"/> 3,3V <input type="checkbox"/> 5V
Control de la retroiluminación a través de PB5 o PC5	JP3	<input type="checkbox"/> PB5 <input type="checkbox"/> PC5
Atención, elegir aquí a PB5 inhibe su selección por JP14.		
S1 conectado a PB0 o PC0	JP4	<input type="checkbox"/> PB0 <input type="checkbox"/> PC0
S2 conectado a PB1 o PC1	JP5	<input type="checkbox"/> PC1 <input type="checkbox"/> PB1
S3 conectado a PB2 o PC2	JP6	<input type="checkbox"/> PC2 <input type="checkbox"/> PB2
S4 conectado a PB3 o PC3	JP7	<input type="checkbox"/> PB3 <input type="checkbox"/> PC3
Permite conectar PC6 de IC2 (DIL28) a / Reset.	JP8	<input type="checkbox"/> PC6
Permite conectar PB7 de IC2 (DIL28) al cristal de cuarzo. Usar siempre en combinación con JP10.	JP9	<input type="checkbox"/> PB7
Permite conectar PB6 de IC2 (DIL28) al cristal de cuarzo. Usar siempre en combinación con JP9.	JP10	<input type="checkbox"/> PB6
SCK-ISP:		
a conectar a PB5 si se utiliza a IC2	JP11	<input type="checkbox"/> PB5 <input type="checkbox"/> PB7
a conectar a PB7 si se utiliza a IC1		
MISO-ISP:		
a conectar a PB4 si se utiliza a IC2	JP12	<input type="checkbox"/> PB4 <input type="checkbox"/> PB6
a conectar a PB6 si se utiliza a IC1		
MOSI-ISP:		
a conectar a PB3 si se utiliza a IC2	JP13	<input type="checkbox"/> PB3 <input type="checkbox"/> PB5
a conectar a PB5 si se utiliza a IC1		
A conectar a PB5 para «compatibilidad» Arduino (ver también a JP3), si no a conectar sobre PC7.	JP14	<input type="checkbox"/> PB5 <input type="checkbox"/> PC7

Curso de audio DSP

Parte 4: probando el hardware

Alexander Potchinkov (Alemania)

En las entregas anteriores de la serie hemos aprendido como tratar señales digitales de audio con un DSP como procesador, los entornos de desarrollo de software describiendo sus funciones y la tarjeta DSP presentada en este curso. En esta ocasión cubriremos el software de desarrollo y las pruebas del hardware. Además, podremos elegir entre dos software tradicionales de diseño con componentes separados y dos debuggers distintos, así como un entorno de desarrollo integrado. En las pruebas del hardware nos serviremos de un pequeño programa DSP, que por supuesto pondremos a disposición de nuestros lectores gratuitamente.



Suite56 y Symphony-Studio son dos entornos de desarrollo de software (SDE). Debemos decidir cual de ellos queremos utilizar, ya que se basan en distintos adaptadores para conectar el DSP (y el programador), y su enfoque y complejidad varía notablemente, y con ello lógicamente el tiempo de diseño. Bueno, vayamos al grano: los programas del curso y los desarrollados por los lectores funcionan igualmente, sin depender del SDE que hayamos utilizado. Los programas independientes que componen Suite56, en Symphony-Studio son plug-ins. Ciertamente, ambos SDEs se diferencian en su funcionamiento, pero no en su utilidad, pues se basan en los mismos programas. Una de las diferencias (trivial sin embargo en nuestro curso) es la programación de los DSPs de doble núcleo de la familia DSP5672x, que en Symphony-Studio puede hacerse directamente. En adelante presentamos ambos SDEs y mostramos las diferencias con el primer programa de test a partir del código fuente `tst_dsp.asm`, desde su programación en ensamblador a la carga en el propio programa en el DSP. En la segunda parte de este artículo se analiza dicho programa de test y otros cuatro más. Si el lector no quiere tomar una decisión

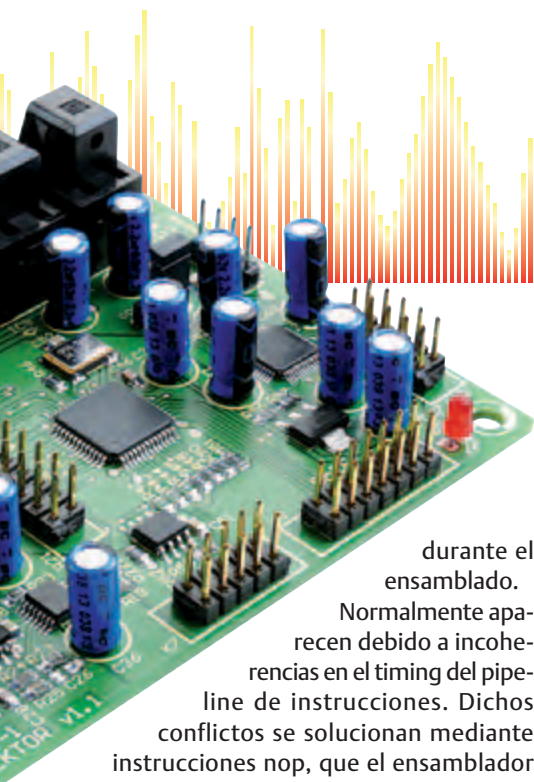
inmediatamente y podría preferir utilizar Symphony-Studio, puede probar primero ambos programas sin hacer uso del adaptador, sólo editando, ensamblando, simulando y haciendo uso del debugger (integrado). Luego, a la hora de conectarlo tendremos que tomar una decisión invariablemente, ya que esto determina el adaptador utilizado.

Suite56

El software Suite56 (DSP56300_TOOLS.exe de aprox. 8,9 MB, siendo un archivo autoextraíble) está formado de componentes individuales antiguos, como el ensamblador `asm56300.exe`, el simulador `sim56300.exe` (o `gds56300.exe` en su versión Windows) y el debugger `gui56300.exe` con un adaptador del puerto paralelo. Si no nos hace falta ningún adaptador de este tipo, podemos utilizar otro debugger cualquiera en lugar del incorporado en Suite56. Dado su “intuitivo funcionamiento”, podemos descargar el atractivo debugger `evm30xw.exe` (con adaptador USB), que no es de Freescale sino de Domaintec, desde la sección de descargas de su web [1]. En adelante damos por hecho que se utilizarán el ensamblador y el simulador

incluidos en Suite56 y dicho debugger de Domaintec. Las ventajas de Suite56 y este debugger son su transparencia y el escaso tiempo de aprendizaje. Los inconvenientes es que tiene una “edad considerable” y el hecho de que el debugger probablemente ya no funcione en un sistema operativo de 64 bits. Sin embargo el autor, convencido de poder obtener resultados más rápidamente, utilizó ambos componentes de software por separado.

Tras hacerse con el software de Freescale y Domaintec, procedió a la sencilla instalación de ambos. El programa debugger de Domaintec ha de ejecutarse con la opción `-cx`, donde `x` corresponde al número de puertos COM (virtuales) dados por el propio sistema Windows, y estando el adaptador conectado. Si no tenemos conectado ningún DSP, podemos lanzarlo con la opción `-D`, en el modo demo, o sea, para probarlo. En el primer paso, ensamblamos en la ventana de `cmd` con la llamada `asm56300 -a -b -l tst_dsp` el primer programa de test y obtenemos el archivo `cld tst_dsp.cld`, el código objeto y archivo de texto (generalmente bastante útil). Este último se trata del código, con el nombre `tst_dsp.lst`. Podemos ignorar los warnings que aparezcan



durante el ensamblado.

Normalmente aparecen debido a incoherencias en el timing del pipeline de instrucciones. Dichos conflictos se solucionan mediante instrucciones nop, que el ensamblador introduce por sí mismo en el programa DSP. Si tenemos experiencia suficiente en la programación de DSPs, podemos meter instrucciones nop “con sentido”. El DSP no necesita nada más que el código objeto (Object-Code) para funcionar. Finalmente el ejecutamos el debugger con la siguiente serie de comandos:

```
force r      (Reseteo del DSPs)
load tst_dsp.cld (Carga del código objeto en el DSP)
go 0         (Implementar el programa con P(rogramm) C(ounter)=0)

force b      (Poner el DSPs en el modo debug)
disp x:$200  (Mostrar el contenido de la X-Ram a partir de la dirección $200)
disp y:$200  (Mostrar el contenido de la Y-Ram a partir de la dirección $200)
```

Y eso es todo. La **figura 1** muestra el entorno del debugger tras implementar el último comando. En la mitad izquierda tenemos los mensajes con el programa DSP y los comandos del debugger, así como los registros DSP-IO. En la mitad derecha se muestra el contenido de las memorias X e Y solicitadas, y los registros DSP. A la hora de mostrar dichas memorias y registros, hemos optado por la representación fraccional mediante [FRA]. Opcionalmente también es capaz de representar en números enteros, al igual que hexadecimales o binarios, o incluso de forma gráfica.

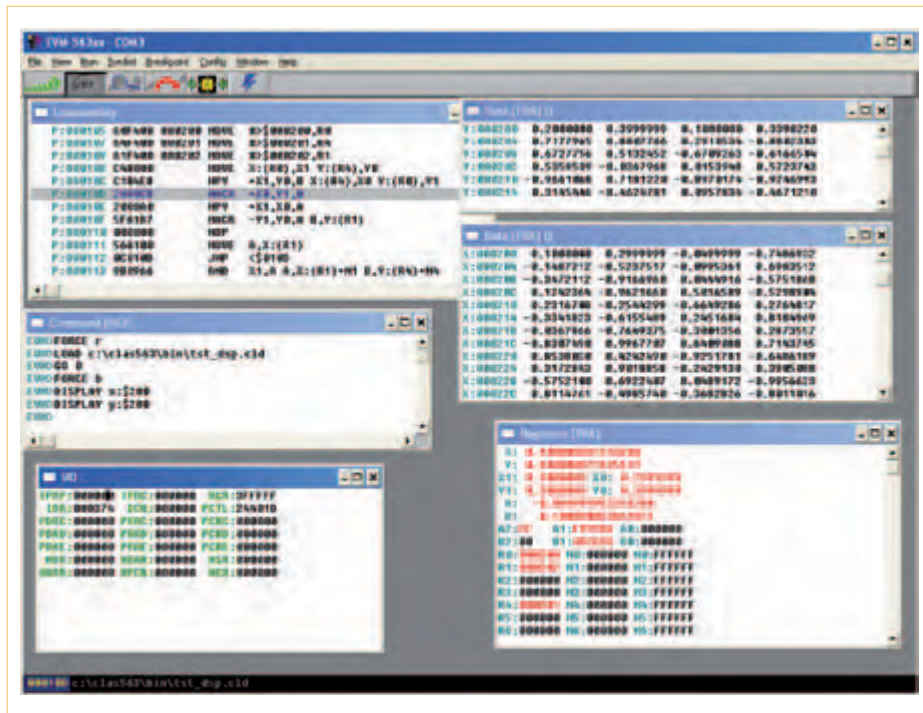


Figura 1. Entorno de usuario del debugger de Domaintec.

Alternativamente, en el programa también puede configurarse el DSP de modo que el debugger pueda acceder a él (con la instrucción en ensamblador *debug*). Lo mismo pasa desde el debugger, con el comando “force b”.

Symphony-Studio

Con Symphony-Studio disponemos de un ensamblador, simulador y debugger junto con un compilador en C, más integrado. Symphony-Studio se basa en el entorno de desarrollo de código abierto para programas en Java, *Eclipse*. Al contrario que Suite56, se trata de un producto actual. Eclipse es utilizado por otros fabricantes de procesadores, y este motivo resulta tan útil por sí solo, familiarizarse con este entorno significa hacerlo con el estándar de la industria. La gran ventaja de Symphony-Studio es la posibilidad de desarrollar en equipo, incluyendo sincronización y control de versiones. Aparte, nada impide que funcione en sistemas Windows de 64 bits. Un inconveniente es la complejidad de su entorno,

con numerosas configuraciones. Con una configuración errónea, el software no reacciona de ninguna forma, con lo que no parece un programa especialmente transparente. Aparte, algunos de los pasos a seguir hasta configurarlo no son en absoluto tan obvios.

Un entorno en Eclipse se basa en componentes específicos, vistas, editores y perspectivas. Para obtener una descripción informativa nos remitimos a Wikipedia, habiendo filtrado algo la información y su redacción (para que se ajuste a Symphony-Studio). Las vistas (Views) son pequeñas ventanas que muestran distintas tareas. Algunas vistas en Symphony-Studio son por ejemplo el navegador o la representación de la estructura de directorios del proyecto. Estas vistas pueden ordenarse a voluntad mediante “drag & drop”. Por ejemplo, a modo de tabs (tarjetas de registro), que se activan con un clic, a modo de ventanas permanentes o en *fast views*, ordenadas como símbolos a lo largo de una barra según queramos (y sólo se ocultan tras hacer clic en su correspon-

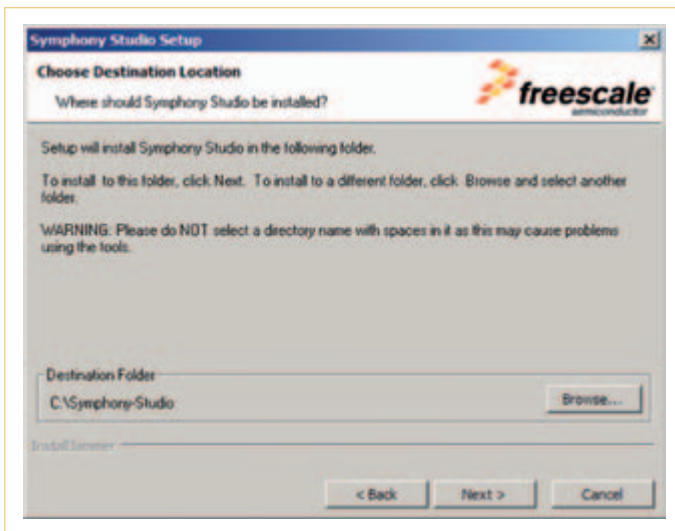


Figura 2. Directorio de instalación.



Figura 3. Opciones de instalación.

diente símbolo). En los editores se muestra el código fuente con las peculiaridades de la sintaxis. Las perspectivas son disposiciones específicas de las listas del menú, los símbolos, las vistas y los editores. Son altamente configurables; las “constelaciones” de cada usuario pueden guardarse y cargarse. Symphony-Studio incluye dos perspectivas, la de **C/C++** para la organización de proyectos y ensamblado del código fuente, y la de **Debug**, en la cual se integran simulador y debugger. Podemos cambiar entre ambas

perspectivas mediante un “selector” (literalmente “interruptor”) en la parte superior derecha del entorno de usuario. Tanto el simulador como el debugger son herramientas de “debug” propiamente dichas, la única diferencia es la existencia de un DSP en el momento. Esto tiene todo el sentido del mundo, de hecho, para el simulador no existe una perspectiva propia. En los pasos siguientes explicamos la instalación de Symphony-Studio y su uso con un programa de ejemplo. Los pasos aquí

dados son igualmente ejemplos, ya que pueden seguirse otros alternativos. Necesitaremos cierto tiempo hasta encontrar nuestro propio camino. Freescale ofrece un manual de instrucciones [2] y varias notas de aplicación [3] para seguir formándonos posteriormente.

Paso a paso

Tras registrarnos, Symphony-Studio puede descargarse de la web de Freescale como un archivo de unos 55 MB de nombre SYM-

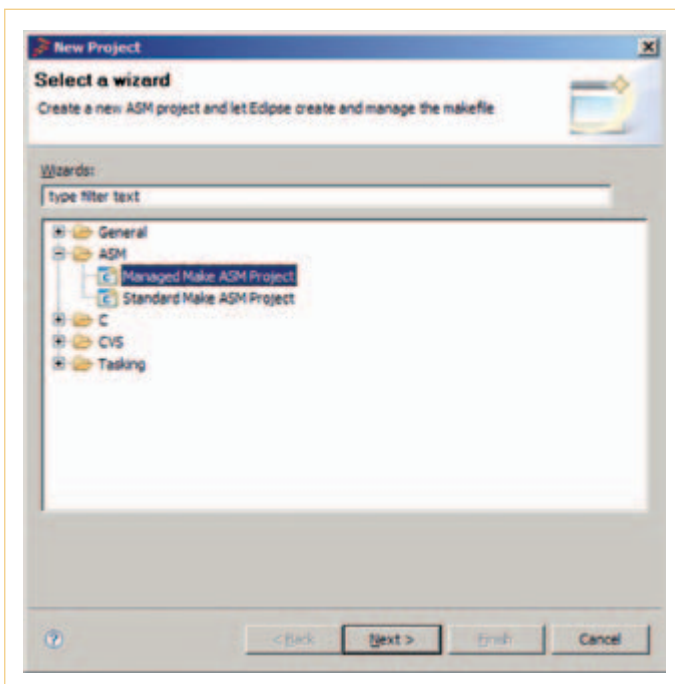


Figura 4. Selección del tipo de proyecto.

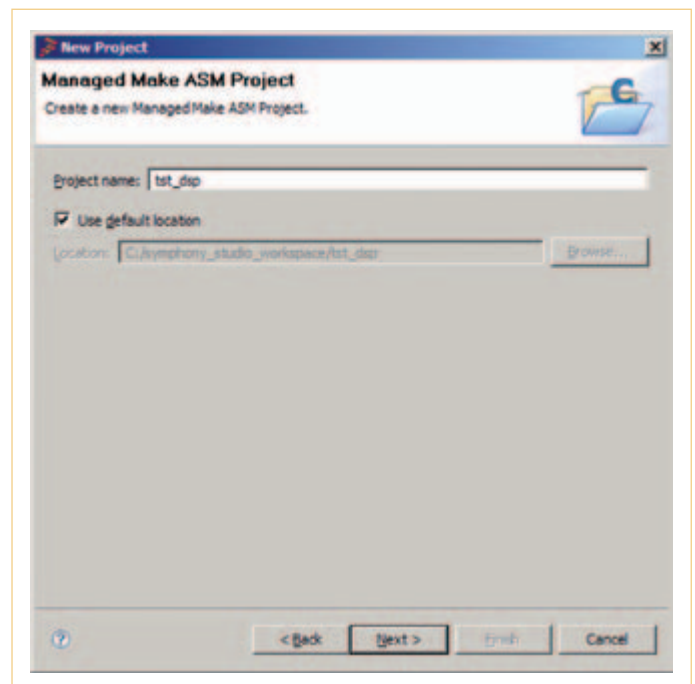


Figura 5. Poniendo nombre al proyecto.

PHONY_STUDIO_IDE.zip, y finalmente instalarlo. Ya que Symphony-Studio se basa en el entorno Eclipse, tendremos que disponer de *Java Runtime Environment* JRE en su versión 1.5. En caso de no disponer de dicho JRE, podemos hacernos con él en [4]. Las **figuras 2 y 3** muestran dos ventanas de diálogo de la instalación. Especialmente tendremos que instalar el driver FTDI (¡no desmarque la casilla de la ventana mostrada en la figura 3!), porque el adaptador USB accede gracias a ellos. La primera vez que lo ejecutamos aparecerá una ventana de bienvenida, en la parte derecha hemos de hacer clic en el icono de *workbench*. Ahora tenemos que hacer clic en el borde superior derecho, sobre *Open Perspective* y después seleccionar la perspectiva C/C++.

El software en Eclipse se crea **siempre** a modo de **proyectos**. Un proyecto de este tipo es lo primero que tenemos que iniciar y acceder. Por ello, hacemos clic en *File -> New -> Project -> Managed Make ASM Project*. La **figura 4** muestra la ventana de *New Project*. Le damos al proyecto un nombre, por ejemplo el del programa de test, como se muestra en la **figura 5**. Ahora especificamos

el tipo de proyecto. La **figura 6** muestra la configuración correcta para nosotros. Nos encontramos en el modo *Build Automatically*, lo que significa que cada vez que cambiemos algo en el código fuente, y hagamos clic en *Save*, el programa se ensamblará automáticamente. El siguiente paso es establecer el directorio del código fuente, haciendo clic en *File -> New -> Sourcefolder*. En la **figura 7** se muestra el nombre del subdirectorio del proyecto, que hemos llamado sencillamente *src*. Ahora ya tenemos el subdirectorio. En el explorador de Windows, pegamos nuestro código fuente *tst_dsp.asm* con *Paste* en el directorio creado previamente *...\tst_dsp\src* en la ventana *Project (View Project)*. El código fuente se abre con *Open File* en un editor. Si hacemos clic en *File* y luego en *Refresh*, se iniciará el ensamblado de nuestro código. En lugar de eso, también podemos editar nuestro código por la marcha (pro forma) y lanzar el ensamblador haciendo clic en *File -> Save*. A la vez se crea un archivo con el código, que podemos cargar con el editor desde *Project-View*. La **figura 8** muestra el entorno de la perspectiva C/C++ con el código fuente inclu-

yendo las palabras clave marcadas con colores y el archivo del código *tst_dsp.lst*. En el marco izquierdo podremos ver la estructura de directorios de nuestro proyecto *Project-View*, así como el subdirectorio *Debug*, del cual hablaremos en adelante.

Desde la perspectiva *Debug* podemos simular, cargar el código en el DSP y hacer el debugging. En el *Im SDE* se utilizan plugins como el simulador y el debugger, las llamadas *External Tools*, las cuales son programas que dependen del entorno Eclipse. Para poder hacer uso de ambas *External Tools*, tendremos que estar conectados. Aparte, se establece un camino para la transferencia de los datos y se configura la herramienta. Para la transferencia de los datos nos servimos de un puerto TCP/IP. La configuración depende del software conectado. Symphony-Studio reconoce la tarjeta Soundbite de Freescale, que debido a una “excepción”, pertenece a la familia de procesadores especiales de señal DSP56371 y DSP56300, de la cual el DSP56374 forma parte. Para configurar el debugger con nuestro propio hardware seguimos los siguientes pasos:

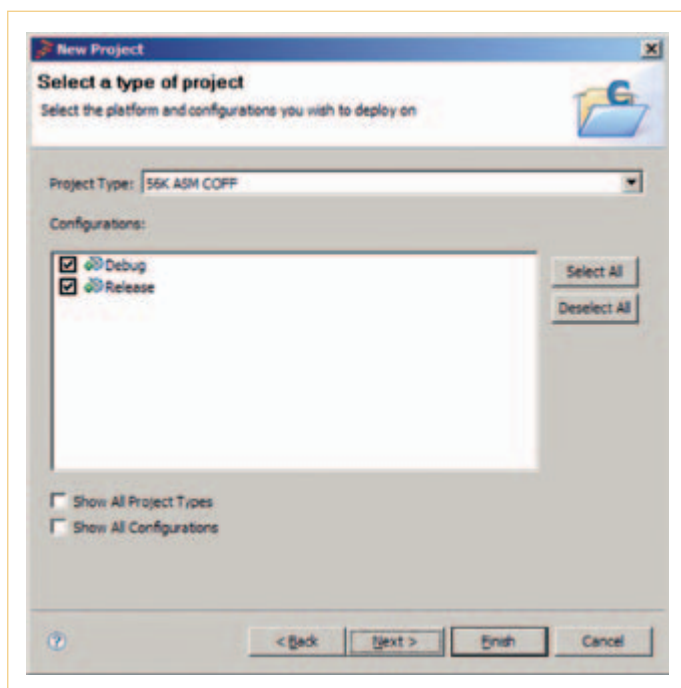


Figura 6. Tipo de proyecto.

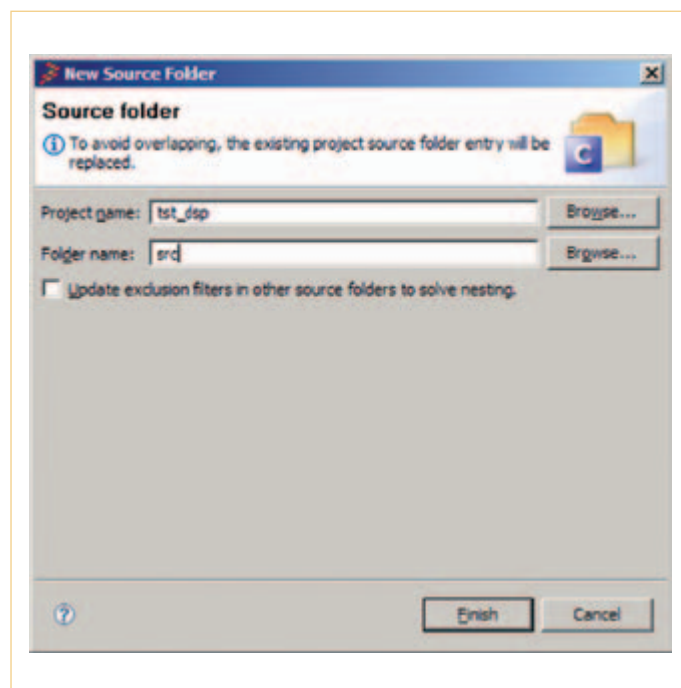


Figura 7. Configurando el proyecto.

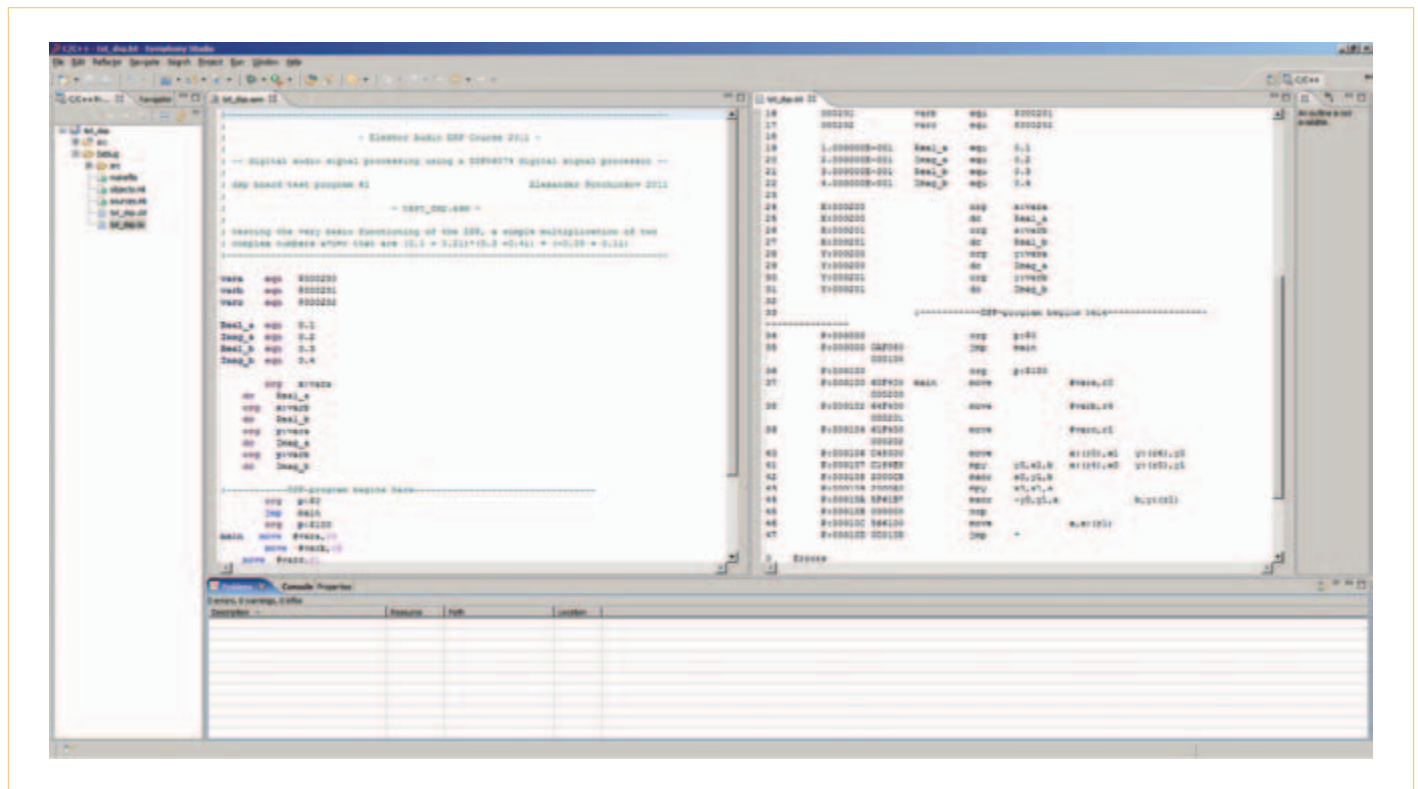


Figura 8. Ver proyecto, código fuente y normal en la perspectiva de C/C++.

1. Pasamos a la perspectiva *Debug* (esquina superior derecha en el entorno de usuario) y seguimos el menú desplegable *Run -> External Tools -> External Tools*.
2. En *External-Tools*, seleccionamos el menú *OpenOCD GDB Server*.
3. Lanzamos el menú *New Launch Configuration*, que aparece como una hoja con el signo "+" en amarillo. Alternativamente, esta configuración también puede hacerse mediante un doble clic en *OpenOCD GDB Server*.
4. En la lista de selección de dispositivo, optamos por el *DSP56300* y *soundbite* en la lista de dongles.
5. Conectamos el hardware con el adaptador al PC. Windows se tomará cierto tiempo hasta enlazar el driver.
6. Una vez el hardware haya sido reconocido y esté listo para ser utilizado, hacemos clic en el botón *Run* para iniciar el debugger. Tras el inicio, veremos una línea de estado con la información: *openocd.c:82 main(): Open On-Chip Debugger... no puede haber ningún mensaje de error*. Si aparece alguno, la mayoría de las veces lo mejor es volver al paso 1 y empezar de nuevo, hasta establecer la conexión.
7. Tras iniciar el debugger tendremos que seleccionar el proyecto que queremos depurar. Nos vamos a la perspectiva *C/C++* y hacemos clic en el directorio de proyecto *tst_dsp* an. Reconoceremos fácilmente la selección por el color (azul).
8. Nos volvemos a la perspectiva *Debug*.
9. Para hacer el debugging seleccionamos el punto del menú *Run -> Debug*. Como procesador (véase la **figura 9**) optamos por el *Freescape 563xx* y ajustamos una nueva configuración de *Debug* con el botón de *New Launch Configuration* ("hoja de papel" con el signo "+" en amarillo). Esta configuración también puede hacerse mediante un doble clic en *Freescape 563xx*.
10. Se creará una nueva configuración de *Debug* con el nombre del proyecto actual.
11. Todavía falta grabar el programa al DSP, aparecerá el mensaje de error "Program not specified" bajo el título "Create, manage and run configurations". El código objeto *tst_dsp.cld* se puede seleccionar en el ordenador mediante el botón *Browse* o simplemente en *Search Project*.
12. Haciendo clic en *Apply* se guarda la configuración actual y en *Debug* se iniciará el debugger, lógicamente.
13. El programa puede ponerse ahora en el modo debug mediante *Run -> Resume* y detenerse con *Run -> Suspend*. Los puntos de ruptura (Breakpoints) pueden fijarse o eliminarse haciendo doble clic en el marco izquierdo de la vista *disassembly*. La vista del registro del procesador es la herramienta más importante para el debugging. Haciendo clic en el signo "+" pueden abrirse los distintos grupos de registros.
14. En *Run -> Step Into* o la tecla *F5*, el programa DSP puede ejecutarse paso a paso, para comprobar el resultado de las instrucciones individuales y el contenido de los registros.
15. Una vez hayamos terminado, podemos cerrar todos los grupos de registros con una sola orden. La vista de registros puede moverse haciendo algunos clics, o dejarla abierta.
16. Para repetir el debugging podemos acceder otra vez a la configuración de "External Tools" así como la de "Debug". Obviamente, dichas configuraciones se encuentran en los puntos "External Tools" y "Debug" y en sus iconos correspondientes.

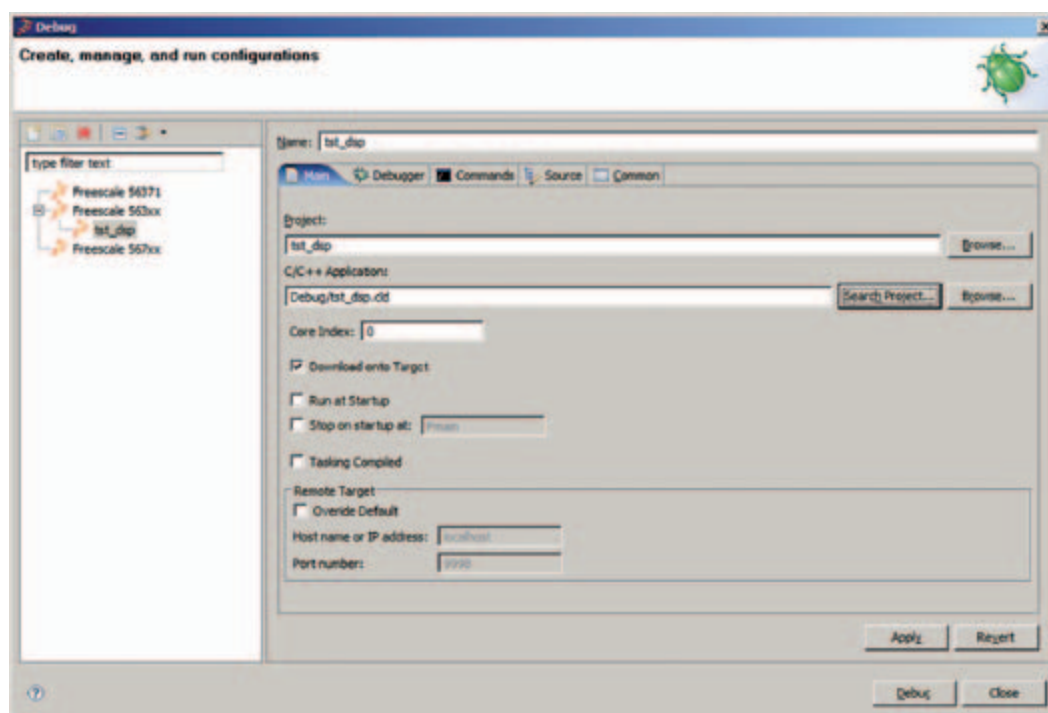


Figura 9. Configurando el debugger.

17. Tras terminar la sesión de debug, hemos de “cortar” la conexión Open-OCD, ya que de otra forma no sería posible iniciar de nuevo el debugger. Esto se hace mediante *Terminate*.

Para lanzar el simulador seguimos los siguientes pasos:

1. Cambiamos a la perspectiva de Debug (esquina superior derecha del entorno de usuario) y acudimos al menú desplegable *Run -> External Tools -> External Tools*.
2. Seleccionamos el modelo *Simapi GDB Server* en *External-Tools*.
3. Hacemos clic en el menú *New Launch Configuration* (“hoja de papel” con el signo “+” en amarillo). Alternativamente, esta configuración también puede hacerse mediante un doble clic en *Simapi GDB Server*.
4. Hacemos clic en *Run*.
5. En la vista *Console* aparece la línea de título “DSP56720 Simulator [SIMAPI GDB Server]” con una entrada para directorio y un botón de Stop rojo, con el cual podemos detener este servidor del simulador.

6. Ahora, para utilizar el simulador, hemos de seguir los mismos pasos que para el debugger (a partir del 7).

Adaptador para los lectores de Elektor

El concepto *Dongle* en Symphony-Studio corresponde a nuestro adaptador. En el paso 4 de arriba, al configurar el debugger se seleccionaba *soundbite*, que también se utiliza para el adaptador USB del autor. En caso de que se utilizase otro adaptador hemos de dejar constancia de ello (los adaptadores aptos se describieron en el segundo artículo). El autor dispone de dos adaptadores USB diseñados por él mismo, uno para Suite56 con debugger de Domaintec y otro para Symphony-Studio, que se implementa como un *soundbite dongle*.

El adaptador para Suite56 de Domaintec del autor consta de un integrado FTDI para convertir de USB al formato serie μ C y un microcontrolador OTP (one time programmable microcontroller) de la familia 68HC05, que por un lado establece comunicación bidireccional mediante RS232 con el PC y por otro en formato serie ONCE mediante cinco pines. Con el integrado de FTDI del tipo FT232BL es posible gestionar un puerto COM virtual en el PC vía USB. En este caso,

si el sistema operativo no dispone de ningún driver VCP (VCP: Virtual Com-Port) tendremos que descargarlo desde la web del fabricante [5] e instalarlo posteriormente. Ya que el autor no está seguro de las peticiones de este adaptador que pueden surgir por parte de los lectores, ha puesto a su disposición algunas tarjetas y microcontroladores ya programadas. Si la demanda fuese alta, habría que pensar en fabricarlas a mayor escala.

El adaptador para Symphony-Studio del autor utiliza un conversor USB paralelo FTDI sin microcontrolador alguno, lo cual lo simplifica algo, ya que el integrado de FTDI requiere escasos periféricos. El autor puede construir un número determinado de adaptadores, y suministrarlos a los lectores de Elektor que los requieran.

Si está interesado en uno o dos adaptadores, puede contactar directamente con el autor [6]. Como ya se dijo en el segundo artículo, existen numerosos adaptadores de distintos fabricantes. Incluso el “viejo conocido” adaptador de puerto paralelo, con unos ligeros ajustes caseros puede conectarse y utilizarse con Symphony-Studio. Sin embargo, debemos asumir que para utilizar un conversor USB a puerto paralelo el PC ha de disponer de

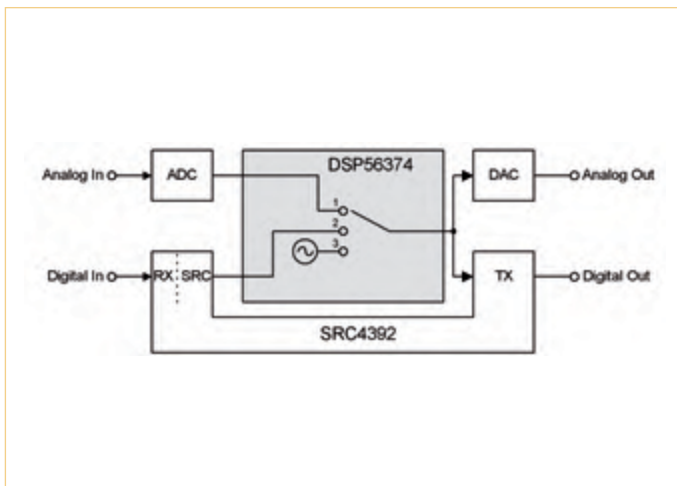


Figura 10. Itinerario de las señales de audio.

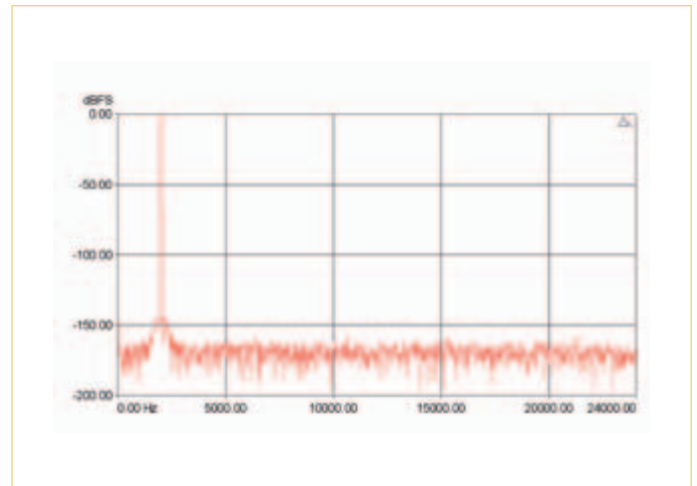


Figura 11. Espectro de la salida digital en el test 5.

uno de estos puertos de impresora antigua, y que podría no ser apto para nuestro adaptador.

Por último, ten en cuenta que el interfaz OnCE/JTAG desarrollado por Elektor se publicará en una próxima edición. Este interfaz puede usarse en combinación con Symphony Studio para programar la tarjeta DSP. El interfaz se suministrará ensamblado al igual que la tarjeta DSP.

Probando el hardware

La tarjeta ya está montada y con la fuente de alimentación múltiple funcionando (el consumo es algo mayor de 130 mA). Ahora hemos de probar las cuatro unidades del circuito de tratamiento de señales: tanto el ADC y DAC para el tratamiento de señales analógicas, como el SRC y el DSP para el de las digitales. El DSP es el master de audio para las cuatro unidades del circuito. Ciertamente esto sí que simplifica el circuito y la programación del DSP, pero no sirve de nada sin el programa DSP. Hemos de considerar esto también para probar el hardware, por ello, hemos desarrollado una serie de cinco programas DSP. La numeración de los programas no establece un orden obligatorio, pero digamos que sí lo sugiere.

Para testear el hardware, el mejor aparato de medida es un analizador de audio con puertos analógicos y digitales. Ya que un analizador de este tipo no siempre estará al alcance de todos los lectores, nos serviremos de una técnica más fácil haciendo uso del PC, a pesar de que los resultados de las medidas no sean comparables con la precisión que ofrece una unidad de este tipo. Resultan particularmente útiles:

- Un reproductor de CD o DVD como fuente analógica y digital de la señal. En Internet podemos encontrar archivos Wave con señales de prueba para utilizar con CDs/DVDs. Necesitamos ante todo oscilaciones sinusoidales con distintos (pero definidos) niveles, que podamos obtener desde un editor de audio y grabar los datos al formato deseado. De este modo podremos introducir señales analógicas en el ADC o digitales en el SRC.
- Un editor de audio, por ejemplo el comercial y profesional Wavelab, o uno gratuito como Audacity, como osciloscopio analógico y digital en el dominio del tiempo (forma de onda) o de la frecuencia (análisis de la FFT). Wavelab dispone de ambos tipos de osciloscopio online. Con el resto de editores tendremos que hacerlo offline, es decir, pasar el test tras grabar a un archivo Wave. En Internet hay otros osciloscopios de audio online gratuitos, capaces de utilizar la tarjeta de sonido del PC. Para probar el DACs también podemos utilizar un osciloscopio convencional, aunque identificar las distorsiones puede resultar difícil, especialmente si se trata de un osciloscopio digital que disponga de análisis de FFT.
- Una tarjeta de sonido USB puede conseguirse por un coste bastante bajo. También puede utilizarse para obtener señales de prueba. Para poner a prueba el SRC es recomendable utilizar una señal de test con una frecuencia de muestreo distinta que la especificada para el resto de programas del DSP, de 48 kHz. Por este motivo (acoplamiento asíncrono) tam-

bién hemos sugerido utilizar un reproductor de CD o DVD como fuente de la señal.

Los programas de test (ver la **tabla 1**) pueden descargarse todos juntos, con algunos archivos necesarios en la página web de Elektor [7]. Dichos archivos adicionales contienen códigos necesarios para el programa, como por ejemplo las rutinas de Interrupt-Service para los puertos de audio, definiciones, la secuencia de bytes para programar el SRC o las señales senoidales para los programas de prueba 2 y 4.

Programa de prueba 1, `tst_dsp.asm`, test del DSPs

Este programa calcula el producto de dos números complejos $a * b = c$ con los valores $(0, 1 + 0, 2i) * (0, 3 + 0, 4i) = (-0, 05 + 0, 1i)$. En las direcciones de memoria \$200, \$201 y \$202 almacenamos las componentes reales en X-Ram y las imaginarias en Y-Ram según el orden a, b y c. Una vez transferido el programa al DSP y comprobado su estado con el debug, podremos leer en los registros del DSP y las áreas de memoria del debugger lo siguiente: $x0=0.3$, $x1=0.1$, $y0=0.4$, $y1=0.2$, $a=-0.0499999...$, $b=0.1000...$, $r0=\$200$, $r1=\$202$, $r4=\$201$, $x:\$200=\$0CCCCD$ (corresp. 0,1), $x:\$201=\266666 (corresp. 0,3), $x:\$202=\$F9999A$ (corresp. -0,05), $y:\$200=\$19999A$ (corresp. 0,2), $y:\$201=\333333 (corresp. 0,4), $y:\$202=\$0CCCCD$ (corresp. 0,1).

Programa de prueba 2, `tst_dac.asm`, test del DACs

Este programa trabaja con la configuración número 3 de la **figura 10** y obtiene dos

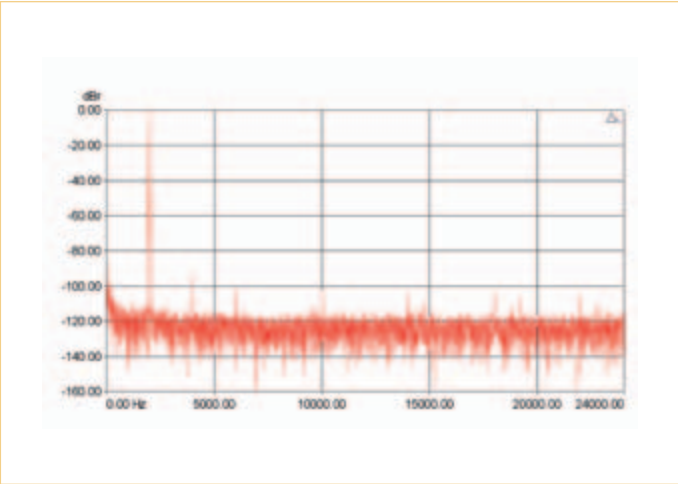


Figura 12. Espectro de la salida analógica en el test 5.

Tabla 1. Archivos adicionales necesarios para los programas de prueba.

<code>src4392.tab</code>	Secuencia de bytes para la programación del SRC
<code>sin1k192.tab</code> , <code>sin2k192.tab</code>	Señales senoidales, 1 kHz y 2 kHz, frecuencia de muestreo de 192
<code>ivt.asm</code>	Entradas del vector de interrupciones, Audio-Interrupts
<code>esai4r2t.asm</code>	Audio ISR, 4 canales de entrada, 2 canales de salida
<code>mioequ.asm</code>	Nombres claros y útiles para las direcciones DSP-IO

señales sinusoidales de 1 kHz para el canal izquierdo y 2 kHz para el derecho, cada una con una modulación de 0.5 FS. En la salida analógica han de detectarse ambas señales. Aquí podemos utilizar bien un osciloscopio o unos auriculares de alta impedancia, en los cuales es mejor limitar el volumen de los altavoces con una resistencia en serie. Para ensamblar este programa tendremos que ajustar la configuración del proyecto según los archivos de “Include” necesarios. Resulta más fácil si creamos una carpeta propia para todos estos archivos de Include en el directorio de proyecto. Lógicamente, este directorio lo hemos llamado “include”. Esto también puede hacerse desde el explorador, pero lo hemos hecho igual que hicimos con el directorio src. Ahora copiamos los archivos a este directorio (véase la tabla 1). El directorio tiene que asignarse todavía mediante un “Refresh”. Podemos ignorar los mensajes de error que parpadean en la consola. En la pestaña “C/C++ Projects” (perspectiva de C/C++) buscamos el subdirectorio “src” y hacemos clic con el botón derecho en “test_dac.asm”. Después hacemos clic en “Properties” y seleccionamos “C/C++ Build”. Después nos vamos a la pestaña “Tool Settings”, seleccionamos “Options”, bajamos con el scroll hasta “Include File (-I)”. Hacemos clic en el signo “+” verde y buscamos el directorio adecuado con el “File System”. Ahora sólo hace falta excluir del ensamblado los tres archivos asm. Buscamos dichos archivos en el directorio de Include (nuevamente en la pestaña “C/C++ Projects”) y hacemos clic derecho sobre ellos. Después vamos otra vez a “Properties” y luego a “C/C++ Build”. En “Active Resource configuration” marca-

mos la casilla “Exclude from build”. Una vez hayamos hecho esto con los tres archivos asm, veremos en la consola que el ensamblador ha obtenido el archivo “tst_dac.cld”.

Programa de prueba 3, `tst_adc.asm`, test del ADCs

Tras probar el funcionamiento del DAC con el programa 2, podemos pasar al ADC. El programa funciona con la configuración número 1 de la figura 10, en la cual la señal del ADCs pasa al DAC. Una señal de audio obtenida en el puerto de entrada analógico, por ejemplo mediante un generador de audio, ha de poder ser transmitida al puerto de salida analógico.

Programa de prueba 4, `tst_src1.asm`, test del SRC, salida

Ahora podemos testear el SRC, que es el responsable de ambos puertos de audio digital. Este programa trabaja con la configuración número 3 de la figura 10 y obtiene dos señales sinusoidales de 1 kHz para el canal izquierdo y 2 kHz para el derecho, cada una con una modulación de 0.5 FS. Con éste primero resetearemos y programaremos el SRC. Ambas señales sinusoidales obtenidas mediante el DSP han de estar accesibles en las salidas tanto analógica como digital. Para probar el puerto de salida digital se recomienda utilizar un PC con interfaz digital de audio (tarjeta de sonido USB) y un editor Wave.

Programa de prueba 5, `tst_src2.asm`, test del SRC, entrada y salida

Este programa trabaja con la configuración número 2 de la figura 10, mediante la cual

la señal de entrada se pasa a la salida y se convierte mediante el DAC. Una señal digital de audio (por ejemplo procedente de un reproductor de CD) ha de poder ser detectada en la salida tanto analógica como digital. Las figuras 11 y 12, obtenidas mediante un analizador de audio (dScopeIII de Prism Sound), muestran el espectro de una oscilación digital de 2 kHz en las salidas analógica y digital. Si los archivos de los programas de prueba y los de la tabla no están en el mismo directorio, tendremos que reconfigurar las características de “Include” en el programa DSP. Una vez hayamos realizado los cinco tests podemos dar paso a nuestros propios programas.

(110004)

Enlaces

[1] www.domaintec.com/ftp/domtech/e30x_331.zip

[2] www.freescale.com/files/dsp/doc/user_guide/DSPSTUDIOUG.pdf

[3] http://cache.freescale.com/files/dsp/doc/app_note/AN3754.pdf

[4] <http://java.sun.com/javase/downloads/index.jsp>

[5] www.ftdichip.com/Drivers/VCP.htm

[6] signum.dsp@gmx.de

[7] www.elektor.es/110004

Hacer malabares con bits de audio

DSP de audio para aplicaciones de construcción propia

Harry Baggen (redacción NL)

Los aficionados al audio limitan, generalmente, su afición al dominio analógico dado que las posibilidades de experimentar uno mismo con lo digital son muy limitadas. Hay muy pocos aparatos asequibles con los que realizar todo tipo de tratamientos digitales de audio e incluso empezar con DSP requiere un conocimiento profundo de esta materia. Con los módulos de miniDSP se pueden realizar todo tipo de funciones de audio fácilmente sin necesidad de profundizar en el tratamiento digital de audio.

Todavía ha de pasar mucho tiempo en el mundo del audio hasta que se extinga la construcción propia. Aún hay muchísima gente que diseña/construye por sí misma sus amplificadores y altavoces o modifican aparatos existentes. A pesar de que el material fuente (la música) y el reproductor son normalmente digitales, hay bastantes posibilidades de hacer arreglos al antojo de cada cual en las siguientes etapas analógicas. Sin embargo, se está notando que se están realizando cada vez más tratamientos de audio a nivel analógico, desde procesadores de señal hasta amplificadores digitales. Este es un mundo bastante complicado para un aficionado medio donde casi no hay posibilidad de intervenir uno mismo. A pesar de ello, cada vez es más interesante profundizar en las posibilidades que ofrecen las técnicas digitales. Piensa, por ejemplo, en un filtro de separación digital para un sistema activo de altavoces, donde puedes configurar sin problema la frecuencia de corte, el ángulo de la pendiente, el tipo de filtro y el retardo entre los driver.

Si bien es cierto que hay algunos sistemas en el mercado que ofrecen la posibilidad de experimentar uno mismo (por ejemplo el módulo de amplificación AS2.100 de Hypex, que contiene dos etapas digitales de potencia y un DSP para el filtrado en la ecualización, o el módulo semiprofesional DCX2496 de Behringer), la oferta en éste ámbito es muy limitada.

La joven empresa miniDSP, afincada en Hong Kong, cuenta con unos ingenieros que, reunidos, tuvieron la idea de lanzar al mercado productos digitales de audio flexibles y asequibles tanto para aficionados como para fabricantes de audio. Su empeño ha sido que la parte de programación de estos productos sea lo más simple posible. Eso significa, en la práctica, que compras una placa con un DSP y luego optas por una de las aplicaciones de software que ofrecen. El software se configura mediante un cómodo interfaz de usuario en el PC. Sólo hace falta introducir una frecuencia de separación o un ángulo de una pendiente de filtro y el DSP hace exactamente lo que has introducido. No se requiere conocimiento de DSP ni de programación y (casi) tampoco de técnica digital de audio.

La base de todo el sistema es la placa miniDSP, una placa de 7,5 x 7,5 cm que contiene un DSP y varios conectores. Le puedes añadir una placa de E/S con varias entradas y salidas digitales (miniDIGI) y una placa con cuatro etapas digitales de potencia (miniAMP). Todas las placas tienen las mismas dimensiones y se pueden montar fácilmente una encima de otra e interconectarlas. Con la ayuda de los denominados "plugin" puedes determinar la funcionalidad de la placa DSP, es decir, el software que grabas en el DSP. De momento

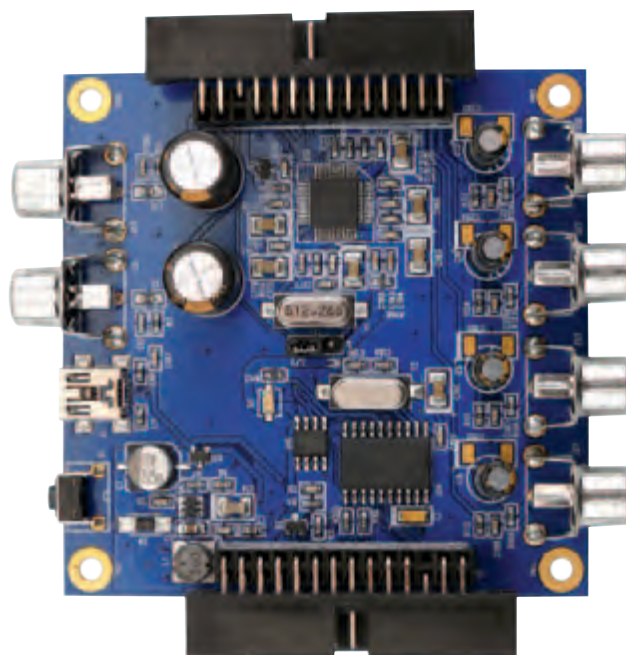
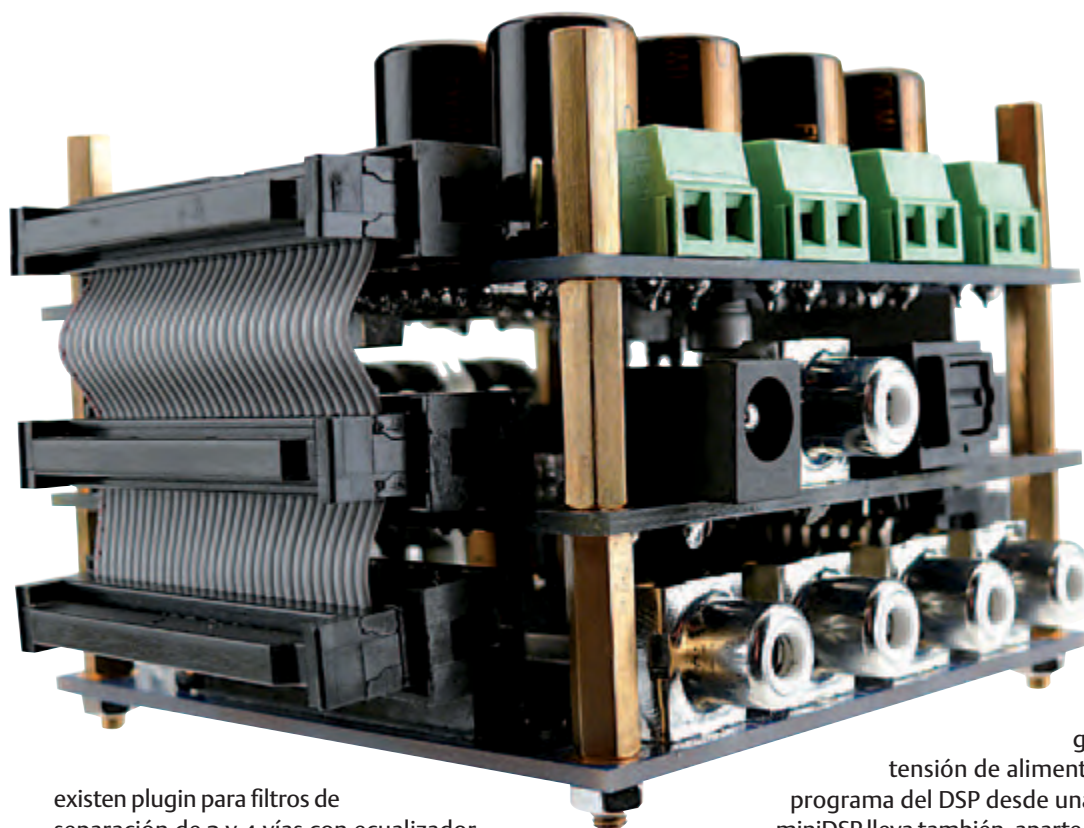


Figura 1. La placa miniDSP contiene dos entradas y cuatro salidas analógicas.



existen plugin para filtros de separación de 2 y 4 vías con ecualizador (en una versión sencilla y una más amplia) y un mezclador con un ecualizador de 31 bandas. Se está trabajando intensivamente en otros plugin. Los plugin funcionan en combinación con Adobe Air que hay que instalar primero en tu ordenador. Hacer doble clic en el plugin descargable es suficiente para instalarlo.

Las diferentes placas

Echemos primero un vistazo al hardware. El núcleo de la placa miniDSP (**figura 1**) es un ADAU1701 de Analog Devices. Este procesador ha sido especialmente desarrollado para aplicaciones de audio. La mayoría de los cálculos se realizan en modo de doble precisión de 56 bit para obtener un resultado preciso. El integrado contiene también, aparte del DSP, convertidores estéreo A/D y D/A de 24 bits que funcionan según el principio sigma-delta, que se encargan de que haya un gran alcance dinámico. Al conectar la

tensión de alimentación se carga automáticamente el programa del DSP desde una EEPROM serie de la placa. La placa miniDSP lleva también, aparte del DSP y la EEPROM, un PIC18F14K50 que se encarga de la comunicación entre el ordenador y el DSP a través de un interfaz USB. La placa se puede conectar fácilmente con otros componentes de audio a través de conectores RCA (2 entradas, 4 salidas). En la placa se ha utilizado condensadores electro-líticos especiales de audio de Nichicon para que haya una mínima influencia de las señales analógicas. En los lados de la placa hay tiras

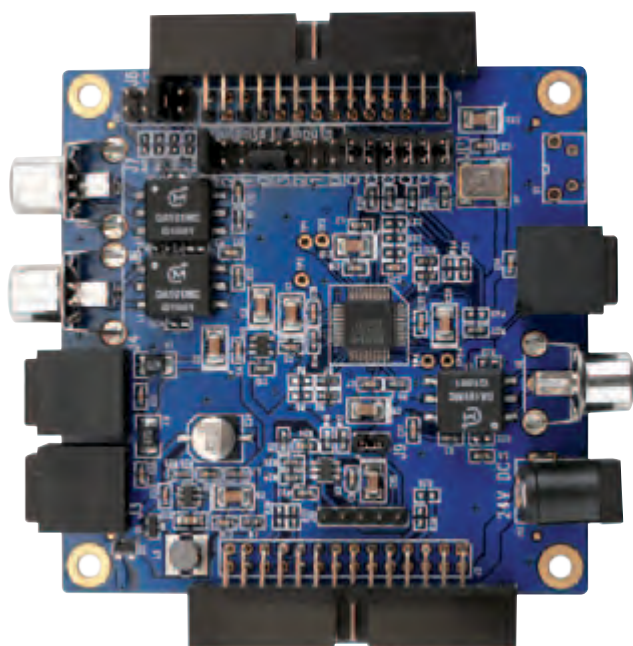


Figura 2. La placa miniDIGI contiene un convertidor de velocidad de muestreo y ofrece una buena cantidad de entradas y salidas digitales.

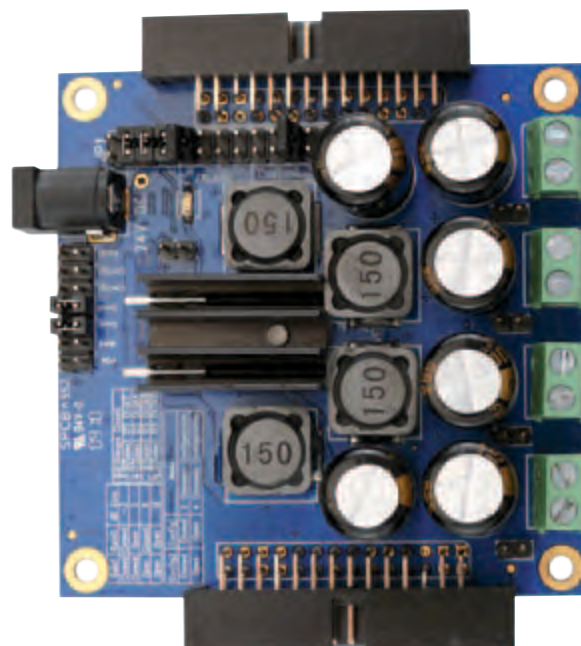


Figura 3. La tercera placa, la miniAMP, contiene un amplificador clase D en un integrado con cuatro canales de salida que también se puede configurar como puente.



Figura 4. La estructura del software. Desde este resumen puedes seleccionar cualquier bloque funcional.



Figura 5. Aquí vemos la parte del filtro de separación. Puedes configurar el punto de corte, el tipo de filtro y el ángulo de la pendiente. Igualmente se puede configurar un tipo de filtro paso banda.

de pines para pasar todas las señales pertinentes a las demás placas miniDSP. También hay la posibilidad de conectar un potenciómetro con el que se puede regular el volumen.

Quien requiera entradas y salidas digitales (por ejemplo para conectar directamente un reproductor de CD, o para pasar directamente la señal tratada en forma digital a otros aparatos), tiene todo tipo de opciones con la placa miniDIGI (figura 2). Aquí encontrará dos entradas coaxiales y dos entradas ópticas y una salida coaxial y una óptica. La placa se vende con o sin transformador de separación para las conexiones coaxiales. El núcleo de esta placa consta de un convertidor de velocidad de muestreo (SRC43821 de TI) que convierte todas las velocidades de muestreo hasta 216 kHz a 48 kHz con la que funciona el DSP. Mediante un puente se puede configurar cual de las 4 entradas se utiliza y con el otro determinar qué señal se conecta con las salidas digitales (por ejemplo la señal de entrada o la señal tratada por el DSP). También se puede utilizar la placa para quitar el jitter de la señal de entrada S/PDIF.

La última de las placas actualmente disponibles es la miniAMP (figura 3). Esta placa contiene cuatro amplificadores digitales que se pueden conectar en puente para obtener una versión estéreo de más potencia de salida. El integrado que contiene el amplificador de potencia es un TAS5704 (igualmente de TI). Puede suministrar una potencia de audio de 4 x 10 W en 4 Ω o 2 x 20 W en 8 Ω en una configuración de puente. Tiene un alto rendimiento (90%) gracias al diseño como amplificador clase D y le basta con una refrigeración de un disipador pequeño colocado directamente sobre la placa. Los condensadores electrolíticos de las salidas son del tipo Nichicon, pero en la configuración puente no son necesarios y puedes puentearlos. El TAS5704 recibe la señal de audio de forma digital, por lo tanto, no se necesita utilizar los convertidores D/A del DSP. Aunque la potencia de salida de ésta placa no es muy elevada, es ideal para experimentar ya que entonces dispone de cuatro salidas. Conecta las unidades de los altavoces directamente a las regletas a través de unos cables y podrás evaluar lo bien que suena la configuración del DSP realizada.

Todas las placas tienen las mismas dimensiones y se pueden conectar fácilmente una encima de otra usando distanciadores. Para las interconexiones entre las placas, se suministran unos cables planos.

La tensión de alimentación para la placa miniDSP se puede obtener a través del conector USB presente. En las placas miniDIGI y miniAMP hay un conector para un adaptador de red. Para las miniDSP y miniDIGI la tensión de alimentación puede estar entre 4,5 y 24 V, la placa miniAMP requiere 12...24 V.

Conectar y probar

Recibimos, para esta prueba, un set completo que consta de miniDSP, miniDIGI y miniAMP. Optamos por una versión DSP con un margen de entrada de 2 V, de modo que podamos conectar un reproductor de CD directamente (también hay una versión con una sensibilidad de 0,9 V, que puedes proveer fácilmente de un divisor de tensión en caso de ser necesario). Las placas (ver foto de portada) se han interconectado a través de cables muy cortos. Con este set completo puedes hacer cualquier cosa, pero en la práctica sólo se compra lo que se necesita para una aplicación determinada. Si quieres, por ejemplo, proveer a una caja infinita de un filtro digital de separación con entradas analógicas y utilizar las etapas de potencia existentes, entonces es suficiente la placa miniDSP. Si quieres también unas entradas digitales, entonces necesitas además una placa miniDIGI. Y si quieres experimentar regularmente con todo tipo de altavoces, entonces la placa miniAMP te resultará un accesorio muy útil.

Las placas se suministran completamente ensambladas. Sólo hace falta conectar una fuente de alimentación y listo. Bueno listo, después de haber configurado todos los puentes de las placas. Especialmente la placa miniDIGI contiene una gran cantidad de configuraciones de puentes, entre las que se encuentra la interconexión correcta de las señales I²S entra las placas. Esto puede resultar un poco confuso en primera instancia. Afortunadamente hay disponible documentación con diferentes configuraciones de ejemplo en el sitio Web del miniDSP, donde se indica exactamente que puente hay que colocar para una determinada combinación o aplicación. Además los desarrolladores nos comunicaron que se está mejorando y ampliando continuamente la documentación. Incluso durante nuestro primer periodo de prueba aparecieron continuas actualizaciones del software y de la documentación.



Figura 6. En cada ecualizador paramétrico del plugin de dos vías se pueden realizar hasta un máximo de seis correcciones (frecuencia, valor, factor Q). Aquí se ha realizado una sencilla amplificación de los bajos.

Como ejemplo de software optamos, en esta prueba, por un filtro de dos vías con ecualizadores paramétricos incorporados, que es un perfecto proyecto de prueba para tal sistema.

En primera instancia conectamos el módulo completo a nuestro Audio Precision System II para la verificación del funcionamiento y la realización de algunas mediciones. El módulo se alimentó de una fuente de alimentación estabilizada potente (tensión de alimentación 24 V).

Para tener una idea de la influencia del DSP sobre una señal analógica de audio, se configuró primero todos los filtros en software en “recto” y aplicamos una señal digital a través de la placa miniDIGI. Luego medimos la distorsión en las salidas analógicas del DSP. Esta fue muy baja, unos 0,005% a 1 kHz y con casi 100% del máximo. Este valor casi no cambió cuando se conmutó a las entradas analógicas del DSP, es decir cuando también los convertidores A/D se encontraban en el bucle. La relación señal/ruido se quedó en algo más de 90 dB (lineal), lo que concuerda con el valor que indica AD para el ADAU1701.

El amplificador clase D de la placa miniAMP funciona con una frecuencia de cerca de 400 kHz (cambia un poquito con la frecuencia de muestreo suministrada), el restante de esta frecuencia tiene en las salidas una amplitud aceptable de cerca de 100 mV_r. La distor-

Publicidad

Crea sistemas electrónicos complejos en minutos utilizando Flowcode 4



Diseña – Simula - Descarga



Flowcode es uno de los lenguajes de programación gráfica más avanzados del mundo para microcontroladores (PIC, AVR, ARM y, ahora también, dsPIC/PIC24). La gran ventaja de Flowcode es que permite a aquellos que tienen poca experiencia crear sistemas electrónicos complejos en minutos. El interfaz gráfico de desarrollo de Flowcode permite a los usuarios construir un sistema electrónico completo en pantalla, desarrollar un programa basado en diagramas de flujo estándar, simular el sistema y generar el código hexadecimal para microcontroladores PIC, microcontroladores AVR y microcontroladores ARM.



¡NUEVO!
Flowcode 4 para
dsPIC/PIC24

Convencete tu mismo.
Versión de demostración,
más información y pedidos en
www.elektor.es/flowcode

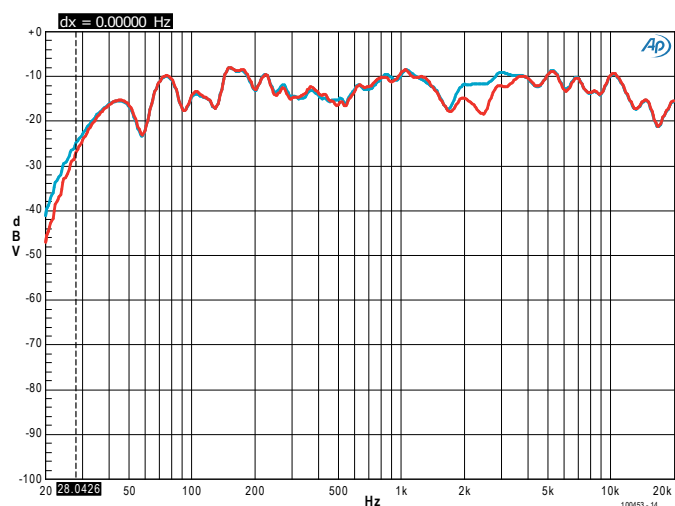


Figura 7. Medición MLS donde se configuró el retardo entre woofer y tweeter en pasos pequeños y se midió al mismo tiempo el desarrollo cerca de la frecuencia de corte. Rojo = sin retardo, azul = retardo de 0,2 ms, unos 7 cm.

sión armónica en la configuración “single-ended” asciende a unas 0,07% a 100 Hz y 1 kHz/1 W/8 Ω . La potencia de salida con 1% THD y una tensión de alimentación de 24 V resultó ascender a 14,5 W con 4 Ω y 8 W con 8 Ω . Esto se aproxima mucho a las especificaciones de TI. No hemos medido la configuración en puente, pero seguro que ascenderá a 20 W con 8 Ω , tal y como menciona la hoja de características de la miniDSP.

La práctica

Después llegó el momento de jugar con el software y ver cómo realizar un circuito práctico. Después de la instalación de Adobe Air runtime (disponible de forma gratuita en el sitio Web de Adobe) y la descarga del plugin de dos vías del sitio Web de miniDSP, sólo hace falta hacer doble clic sobre el fichero .air descargado para que a continuación se ejecute automáticamente la instalación y se creen también los accesos directos. Sólo necesitamos interconectar la placa DSP con el PC a través de un cable con mini conector USB y ya podremos empezar a experimentar.

Después de iniciar el programa, aparece una pestaña “Audio Settings” con un resumen en forma de diagrama de bloques con la estructura del software. En la parte inferior de la pestaña “System Settings” encontramos algunas cosas comunes, como son la selección del tipo de señal de entrada (analógica o digital), la activación de un regulador de volumen externo y la posibilidad de guardar y cargar configuraciones (en formato xml). De esta manera podemos guardar todas las configuraciones y posteriormente recuperarlas fácilmente. También hay un botón para volver a las configuraciones por defecto. Finalmente, hay una tercera pestaña con la que llegas al sitio Web de miniDSP. Todos los plugin tienen la misma estructura.

Cuando haces clic sobre uno de los bloques funcionales que hay debajo de la pestaña “Audio Settings” aparecen las configuraciones correspondientes. En el plugin filtro de cruce de 2 vías, puedes configurar la atenuación de la señal de entrada en el primer bloque. Luego hay por cada canal, un ecualizador paramé-

trico con 6 puntos de configuración. A continuación llegamos a la parte del filtro de cruce en donde puedes seleccionar por cada driver el punto de corte, el tipo de filtro y el ángulo de la pendiente. Los tipos de filtros disponibles son Bessel, Linkwitz-Riley (12, 24 y 48 dB/oct) y Butterworth (6...48 dB/oct). Además puedes ampliar cada filtro a uno del tipo de paso banda, por ejemplo para proteger un woofer pequeño contra frecuencias excesivamente bajas o para combinar un sistema de dos vías con un subwoofer ya existente. Después de esta sección llega un ecualizador paramétrico de 6 bandas para realizar correcciones individuales por cada driver. En el último bloque se puede configurar la atenuación y el retardo por cada driver. Este último es especialmente importante en configuraciones Linkwitz-Riley, ya que en este caso el centro acústico del woofer y del tweeter tiene que quedar exactamente el uno encima del otro. La corrección es bastante elevada, hasta 7,5 ms o sea 2,5 m.

Después de realizar los ajustes necesarios puedes pulsar el botón verde “Synchronize” para que el software se conecte con la placa miniDSP (a través del cable USB). La configuración realizada se envía a la placa miniDSP y se guarda. También se actualiza el firmware de DSP en caso de que fuera necesario. Después de sincronizar desaparece el botón verde, lo que parece en primera instancia algo extraño. ¿Entonces no se puede sincronizar otra vez? Sí, después de una sincronización se envían todos los cambios posteriores directamente a la placa miniDSP, hay, por así decir, una conexión permanente. Así puedes experimentar a gusto y cambiar todo tipo de cosas. Puedes, por ejemplo, proveerlo de una señal de música y oír directamente el efecto que tiene una determinada adaptación sobre el sonido. Durante los experimentos con una caja infinita de dos vías pudimos adaptar así el nivel del tweeter e incluso jugar con el retardo entre woofer y tweeter. ¡Una posibilidad fantástica para experimentar con sistemas de altavoces! En el momento en que se cierra el programa del PC, se guarda automáticamente la última configuración en la memoria de la placa miniDSP.

En el momento en que hayas terminado con los experimentos, podrás integrar las placas en la caja infinita, junto con una fuente de alimentación y los amplificadores de potencia. Como ya hemos comentado anteriormente, la placa miniAMP es muy práctica para hacer experimentos, pero para el sistema de altavoces final es mejor optar por algún amplificador de potencia más potente (y cualitativamente mejor).

Muy útil

Con el sistema miniDSP puedes construir fácilmente un circuito de audio en muy poco tiempo que sería mucho más difícil de realizar con componentes analógicos. Piensa, por ejemplo, en un filtro de separación activo con una corrección de duración y varias correcciones de frecuencias. Con este sistema es pan comido. No requiere ningún conocimiento de DSP, pero, por supuesto, sí el conocimiento necesario y experiencia en el área de técnicas de filtros y altavoces. Pero estos los necesitabas de todas formas si quisieras diseñar sistemas de altavoces tú mismo. El método para adaptar configuraciones directamente desde tu PC funciona

perfectamente y te incita a probar muchas más cosas que inicialmente tenías en mente. Después de todos los experimentos se puede crear fácilmente una versión definitiva construyendo la placa miniDSP junto con una fuente de alimentación y las etapas de potencia correspondiente en la caja infinita. Y si después de un tiempo la configuración no parece ser la más adecuada, puedes conectar el PC para realizar alguna corrección.

A pesar de las amplias posibilidades, el sistema miniDSP es muy asequible. Para la mayoría de las aplicaciones tienes suficiente con la placa miniDSP y la puedes conseguir por \$99 más \$10 para el plugin. Una sola placa capaz de controlar dos cajas infinitas de dos vías. Sin embargo en la mayoría de los casos, es más práctico construir una placa miniDSP en cada caja, donde cada placa tenga además suficiente potencia de cálculo para un filtro de cuatro vías.

Las calidades de audio de la placa miniDSP son bastante buenas, pero nos podemos imaginar que algunos aficionados al audio no estén aún satisfechos y prefieran utilizar otros convertidores A/D y D/A a los ejemplares integrados en el DSP. También se puede hacer

esto con esta placa utilizando las entradas y salidas I²S de la placa miniDSP. Y así hay muchas más configuraciones y posibilidades... Sin embargo, lo más importante es que con estas placas puedes empezar a trabajar fácilmente con el tratamiento digital de audio. ¡Y además por precios muy atractivos!

(100453)

Enlace en Internet: www.minidsp.com

miniDSP, rev A (0,9 V) o rev B (2,0 V): \$ 99

minDIGI rev A (sin transformador S/PDIF): \$ 55

minDIGI rev A (con transformador S/PDIF): \$ 60

miniAMP: \$ 60

Audio plugin: \$ 10 de unidad

Además hay disponibles varios paquetes de combinación

Publicidad

Elektor OSPV

Open Source Personal Vehicle

El vehículo autobalanceado Elektor OSPV está basado en la misma idea y tecnología que el exitoso ElektorWheellie.

En el diseño hay sólo una diferencia: ¡es para uso interior!

Se conduce fácilmente, es ligero y plegable, hecho en código libre y además tiene un bonito aspecto.

En primera instancia el OSPV está pensado para el desplazamiento de personas, pero... no hace falta que siga siendo así. Podrías inventar otras aplicaciones que varían desde una carretilla eléctrica hasta una útil ayuda para las compras. Esta es la ventaja del código abierto.



Especificaciones principales:

- Dimensiones: 120 x 47 x 47 cm
- Peso: 25 kg
- Carga máxima: 90 kg
- Motores: DC 2 x 200 W
- Ruedas: PU, 14 cm de diámetro
- Transmisión: correa dentada HDT
- Velocidad máxima: 15 km/h
- Radio de acción: 8 km

El kit incluye de motores de tracción DC de 200 vatios, dos baterías AGM plomo-ácido de 12 V, cargador de batería, dos ruedas de Poliuretano de 14 cm, carcasa, palanca de control y placa de control con placa de sensores completamente montadas y comprobadas.

Art.# 110320-91 • 1095,00 €

*IVA incl., gastos de envío excl.

Más información y pedidos en www.elektor.es/ospv

La Máquina del Caos:

Cálculo Analógico Redescubierto (2)

Maarten H. P. Ambaum y R. Giles Harrison (Departamento de Meteorología, Universidad de Reading, UK), Jan Buiting y Thijs Beckers (Laboratorios de Elektor)

El ordenador analógico, que intentamos describir en la entrega anterior, fue construido a partir de módulos de cálculo independientes de multiplicación, integración, suma y escalado, combinados para representar el sistema de ecuaciones de Lorenz de 1963 (referencia parte 1). Los circuitos de los módulos están basados en gran parte en las sugerencias y suposiciones de Peyton y Walsh en, *Electrónica Analógica con Amplificadores Operacionales: Un Libro Fuente de Circuitos Prácticos*, donde podemos encontrar más detalles de sus funcionalidades. Hemos encontrado el uso de placas de prueba muy útiles para este proyecto, pero también hemos hecho una versión con soldaduras que se transporta mejor.

Aproximación Modular al Caos

La **Figura 1** proporciona una visión general de cómo se combinan los módulos de cálculo, desde el punto de vista del camino de la señal, mientras que las **Figuras 2a a 2g** proporcionan los esquemas eléctricos individuales para cada uno de los módulos de cálculo que se necesitan. A continuación se da un resumen de su funcionamiento, pero antes de nada vamos a ver:

El Diagrama de Bloques (Figura 1). Esta figura muestra la combinación de los módulos de cálculo requeridos para formar un ordenador analógico completo. Los triángulos representan módulos de función, cada uno de ellos con un conjunto de señales de entrada y una única salida.

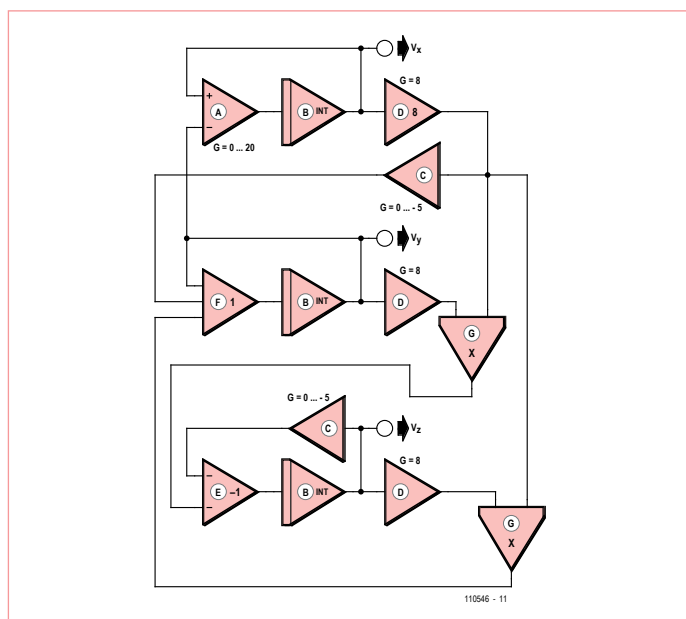


Figura 1. Diagrama de bloques de la Máquina del Caos. Cada función mostrada se corresponde con uno de los siete circuitos básicos de las Figuras 2a a 2g resolviendo las ecuaciones de Lorenz.

Los símbolos (+) y (–) son usados para indicar las entradas no inversoras e inversoras, respectivamente, mientras que la (×) es la multiplicación. Un delgado rectángulo a la entrada del triángulo indica un integrador. Los amplificadores escalados de ganancia 8 (identificados como: ‘D’) son incluidos para asegurar que las tensiones de cada hilo no superan el valor máximo establecido de ± 10 V, en cualquiera de los amplificadores operacionales de las etapas de entrada y del circuito multiplicador.

Las tensiones de salida están disponibles en tres nodos marcados como V_x , V_y , V_z , para su uso con un osciloscopio que disponga del modo de pantalla 'xy' y de modulación (intensidad) del eje 'z'.

Este modelo tiene un control (en el módulo 'A') para variar el número Prandtl (parámetro σ en las ecuaciones de Lorenz) entre 0 y 20, para mostrar los diferentes regímenes de las ecuaciones de Lorenz.

A: Amplificador Diferencial (Figura 2a). Las tensiones de entrada V_1 y V_2 son almacenadas por una etapa con amplificador operacional dual, A1, restadas en el amplificador operacional de la etapa A2.A y, a continuación, amplificada por una etapa amplificadora inversora A2.B, con ganancia G (de hasta $\times -20$). Función: $V_{out} = G (V_1 - V_2)$.

B: Integrador Inversor (Figura 2b). La tensión de entrada V_1 , referida a la señal de masa, es almacenada por la etapa con el amplificador operacional A2.A y, seguidamente, integrada por la etapa A2.B. Un encapsulado con un amplificador operacional dual proporciona ambos amplificadores. Función: $V_{out} = \int V_1 dt / (3,3 \times 10^{-4} s)$

C: Amplificador Escalado Inversor (Figura 2c). La tensión de entrada V_1 , referida a la señal de masa, es almacenada y aplicada a una etapa amplificadora inversora con ganancia G variable (de hasta $\times -5$), controlada por el potenciómetro de 100 k Ω . Función: $V_{out} = -GV_1$. Para la mariposa de Lorenz, pon $G=-3.5$ en el módulo C de arriba y $G = -2.7$ en el módulo C de abajo.

D: Amplificador Escalado no Inversor (Figura 2d). La tensión de entrada V_i , referida a la señal de masa, es almacenada y amplificada por una etapa no inversora con ganancia fija de 8. Función: $V_{out} = 8V_i$.

E: Amplificador Sumador Inversor (Figura 2e). Las tensiones de entrada V_1 y V_2 son almacenadas y sumadas en una tercera etapa inversora. Un encapsulado con un amplificador operacional dual proporciona ambos amplificadores y un encapsulado adicional, la etapa de suma. Función: $V_{out} = -(V_1 + V_2)$.

F: Amplificador Sumador no Inversor (Figura 2f). Cada una de las tres tensiones de entrada, V_1 , V_2 , y V_3 son almacenadas y, a continuación, sumadas en una etapa sumadora. Se pueden usar dos encapsulados con amplificadores operacionales duales. Función: $V_{out} = V_1 + V_2 + V_3$.

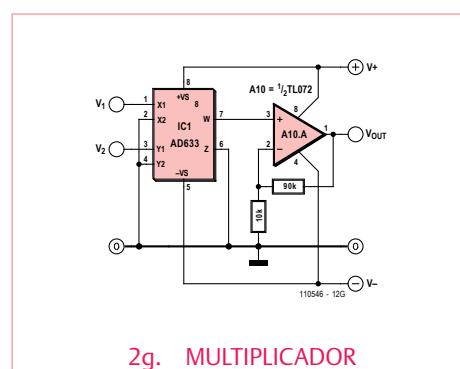
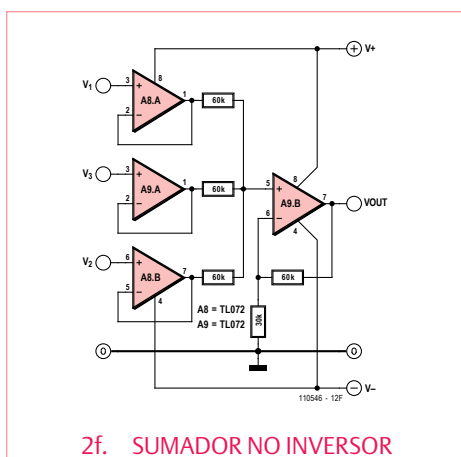
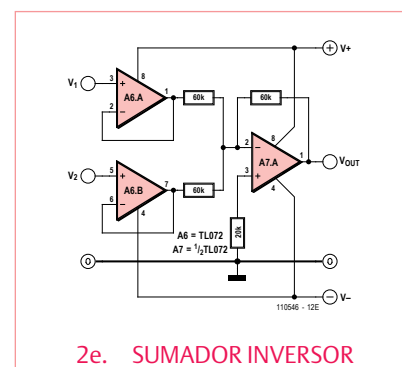
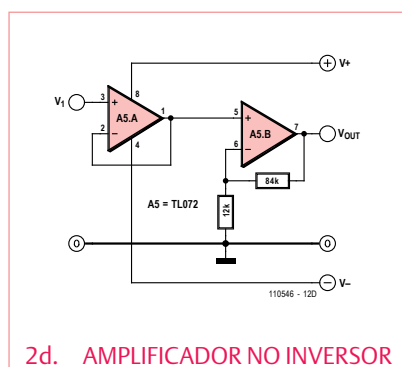
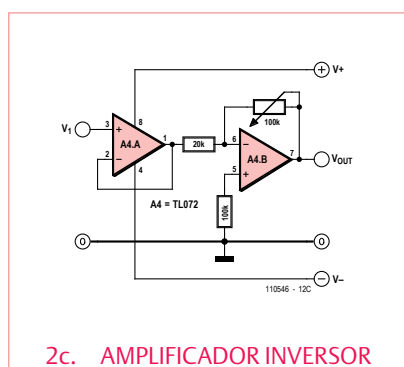
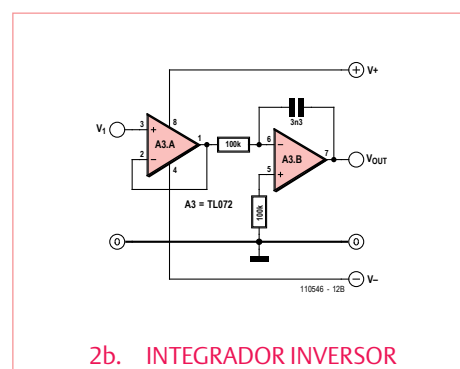
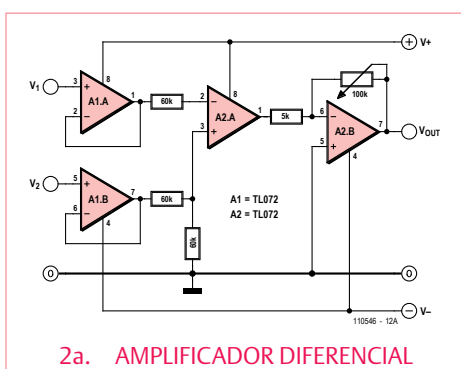


Figura 2. Vista general de todas las funciones matemáticas requeridas para la Máquina del Caos, realizada usando amplificadores operacionales en su mayor parte.

El Caos en la Teoría

La conclusión básica del trabajo de Lorenz consiste en que incluso si sabemos las condiciones iniciales con total exactitud, el error en la condición inicial especificada crece rápidamente, de modo que, incluso después de un corto periodo de tiempo, no podamos predecir los detalles del movimiento (la dependencia sensible de las condiciones iniciales es una de las propiedades que definen el caos).

El Caos real

Jan Buiting, Editor.



Los artículos de “Retrónica” cumplen normalmente con el conocimiento y la aprobación velada de gente de Elektor con inclinaciones más técnicas y con la leve sorpresa o la sonrisita rara del resto del personal, viendo y escuchando equipos antiguos, metidos y sacados de los sótanos húmedos de la Casa medieval de Elektor. Sin embargo, cuando el rumor era que *“El caos había bajado sobre las páginas de Jan en la edición de septiembre de 2011”* muchos empleados se desilusionaron al ver una entrega de “Retrónica” con un formato esmerado y un contenido sólido, y ningún caos u otro tipo de desorden en la misma.

Interesados en experimentar el caos verdadero, algunos aficionados murmuraron y, después, sugirieron construir realmente el Generador de Caos y así lo hicimos, donde ‘nosotros’ \approx [Thijs Beckers \pm Jan Buiting] de los “Chaosfree Desks”, una facción pequeña y tranquila dentro de la editorial de Elektor y un grupo de laboratorio.

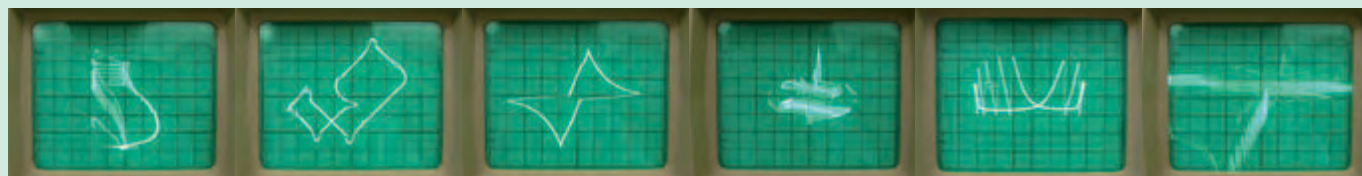
Las funciones matemáticas que esperanzadoramente permiten al generador comportarse caóticamente, fueron conectadas al circuito, así como los componentes y, finalmente, los módulos que se interconectan con hilos (señales así como fuentes de alimentación). El equipo funcionó bien, produciendo imágenes extrañas en nuestro osciloscopio Hameg, en el modo x-y (por desgracia, todos los modelos con modulación z no funcionaban correctamente). Ajustando los dos mandos y, de vez en cuando, introduciendo una pequeña capacidad parásita con nuestros dedos bajo algunas placas, fuimos capaces de producir formas extremadamente complejas que iban desde caballitos de mar a bandas de Möbius, cadenas de ADN, con forma de epsilon y hasta modelos de negocio bastante adecuados para el grado de doctor. En la Escuela de Económicas de Londres (LSE) el equipo de contabilidad y marketing obtuvo su MD. A Dilbert y al profesor Bill Phillips les habría gustado. Por ejemplo, actuando sobre el potenciómetro del módulo “A” (LSE: “aumentando los recursos de ventas de la compañía XYZ”) más allá de un cierto nivel crítico, se provocó que la imagen completa del osciloscopio se amotinase (LSE: “esta organización altamente creativa”), para cambiar de forma (LSE: “adaptando sus presupuestos”) haciendo, a continuación, un desplazamiento de la corriente continua (DC) en la pantalla del osciloscopio (LSE: “entra en la búsqueda de otros desafíos”) y, finalmente, provocando un rebote hacia atrás en la pantalla, lo que parecía un vórtice violento, no como el que se produce en los servicios de un avión (LSE: “depreciación repentina de activos”). ¡Tan voluble como los mercados financieros modernos!

Algunas de las mejores imágenes caóticas se muestran en este apartado, así como las filmadas para un videoclip que podremos visualizar muy propio en el Canal de YouTube de Elektor (los detalles serán anunciados en línea). Todas las imágenes aquí mostradas son probablemente debidas a la saturación de los amplificadores operacionales.

Las señales de sonido tomadas en las salidas eran tan impresionantes y extrañas como Maarten y Giles mencionaron en la parte 1 de este artículo. ¡Tenéis que oírlas para creerlas y todo el mundo está invitado a comprobarlo en el acontecimiento de Elektor Live!, este mes de noviembre.

Para algunos el generador es un “chisme” (aparato) al que no podemos resistirnos a manipular y ajustar por medio de los dos mandos, para conseguir formas aún más locas en la pantalla del osciloscopio. Para otros, es una realización seria de funciones matemáticas complejas, un potente DSP de un microcontrolador PIC que tendrá como reto producir resultados similares, es decir, de la misma forma visual y con la misma velocidad. Es verdad que el uso práctico de la máquina es limitado, en el mejor de los casos, con algún consuelo de que el tiempo es el sistema caótico reconocido más grande que conocemos (con algunos puntos de trabajo en la Casa Elektor que, felizmente, compiten por el segundo lugar).

En este punto, desafiamos a nuestros lectores a investigar si sus potentes programas de simulación electrónica de 32 y 64 bits y sus ordenadores personales están a la altura de la circuitería del Generador de Caos, encantadoramente simple, mostrada aquí. Si esto no funciona, o si estamos cansados de informes de error que aparecen (¡División por Cero!), jenviadnos la mejor aplicación del Generador de Caos en la que podáis pensar! O un “applet” de Caos para el iPhone o Android que permita a empresarios de rango superior usar esto en el tren (hay dinero para hacerlos).



el caos en vivo

Prueba el Generador del Caos en Elektor Live! 2011

División de Audio & Retro de Elektor
Eindhoven, Holanda, 26 de Noviembre

G: Multiplicador (Figura 2g). Se ha usado un circuito integrado función (del tipo AD633) para determinar el producto de dos tensiones de entrada, con una etapa no inversora adicional que contribuye con una ganancia de $\times 10$, para establecer una tensión escalada V_0 de 1 V. Función: $V_{out} = V_1 \times V_2 / V_0$.

Sugerencias para el montaje

En la Universidad de Reading, los componentes para el prototipo fueron ensamblados sobre placas de experimentación, sin soldaduras, usando una placa de experimentación independiente para cada módulo funcional. Todas las etapas del circuito estaban alimentadas con una tensión de alimentación común de ± 15 V, con una tensión de masa convencional de 0 V central. Los amplificadores operacionales usados fueron del tipo OP97. Este amplificador operacional (AO) está disponible en encapsulado con versión sencilla (OP97) o versión dual (OP297), los cuales pueden ser combinados para reducir el número de circuitos integrados necesarios, al mismo tiempo que preservamos la independencia de los módulos. Los números de los terminales hacen referencia a los encapsulados de circuitos integrados DIP estándar de los OP97 y OP297. Las etapas de multiplicación usan un circuito integrado de función matemática, el AD633, el cual utiliza las mismas fuentes de alimentación bipolar.

La réplica de la Máquina del Caos, construida en el Laboratorio de Elektor, usa amplificadores operacionales (AO) TL072 de taladros pasantes por la única razón de que estaban disponibles. Además, algunos de los valores teóricos de resistencias, como 60 k Ω , fueron sustituidos por sus equivalentes reales más próximos, buscando en cajones de componentes. Para añadir alguna estética al proyecto, los módulos fueron sujetos a un poste central hecho con separadores de PCs superpuestos y girados para parecer los escalones de una escalera prácticamente redonda, ver **Figura 3**. Se utilizó una Placa de Prototipo Universal de Elektor ('Elektor Universal Prototyping Board size-1' o UPB-1), para construir todos los módulos y darles una apariencia de uniformidad. Las placas fueron etiquetadas y, algunas de ellas, duplicadas para permitir otras funciones matemáticas y establecer otras configuraciones.

Agradecimientos: Este proyecto fue promovido entre diferentes talleres interdisciplinarios de artistas y científicos, conducidos por el artista Charlie Hooker, de la Universidad de Brighton de Reino Unido, en el Departamento de "Fine Art". El ordenador analógico fue construido en los laboratorios del Departamento de Meteorología, por Stephen R. Tames, en la Universidad de Reading, Reino Unido.

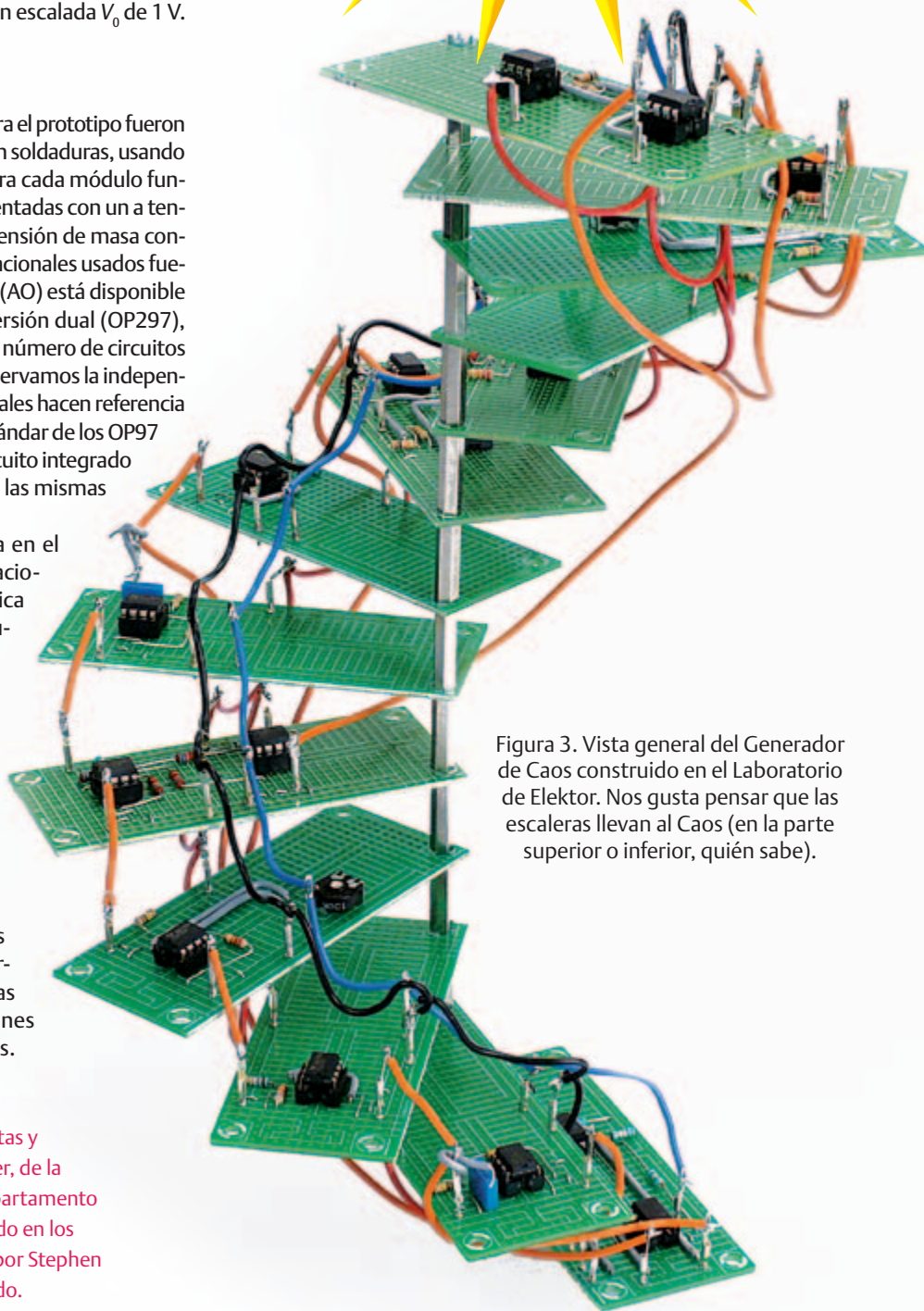


Figura 3. Vista general del Generador de Caos construido en el Laboratorio de Elektor. Nos gusta pensar que las escaleras llevan al Caos (en la parte superior o inferior, quién sabe).

(110546)

Retrónica (Recuerdos de electrónica) es una columna mensual que cubre equipos electrónicos antiguos, incluyendo diseños legendarios de Elektor. Se agradecen contribuciones, sugerencias y peticiones; por favor, enviad un correo electrónico (email) a redaccion@elektor.es.