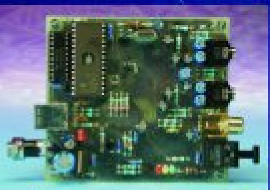
# ELEKTOR ELECTRONICS

MAY 2003
£3.45

THE ELECTRONICS & COMPUTER MAGAZINE

www.elektor-electronics.co.uk



**USB Audio Recorder**

**DMX 3-in-1**

**Countdown Timer**

**Low-Cost LCD Controller**

**Two-Eyed LED Lamp**

**Printing PCB Artwork**

# UNIVERSAL XA DEVELOPMENT BOARD

**for 16-bit XA micros**

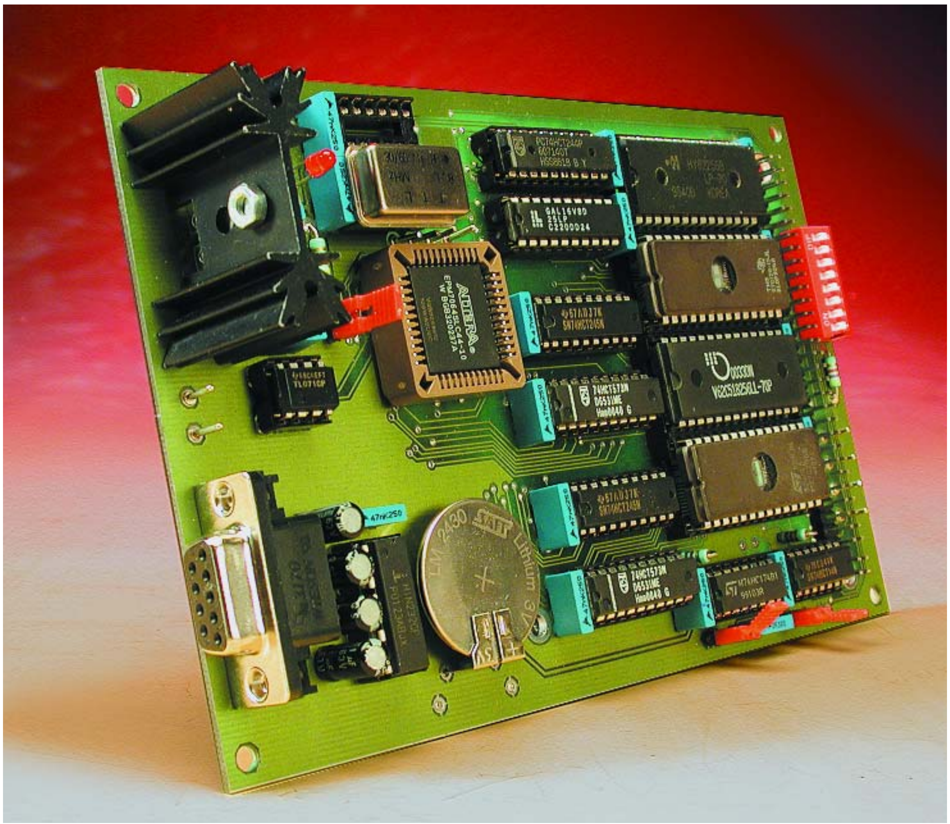
9 770268 451104 05

# Universal XA Development Board (1)

## For 16-bit XA microcontrollers, with PC/104 interface

Design by B. Bouchez

The Philips XA series of microcontrollers has been available since 1996. These microcontrollers, which were originally presented as 16-bit versions of the famous 80C32, are more than just successor versions. They have much more to offer than their 8-bit cousins.



In the beginning, the eXtended Architecture (XA) family of microcontrollers had only one member: the XA-G3 (introduced in 1996), with 32 KB of internal memory. In response to repeated requests, less expensive types with smaller amounts of memory were introduced in 1997, consisting of the XA-G1 (8 KB) and the XA-G2 (16 KB). The 'G' in these type numbers refers to the general-purpose nature of these models (with the same pinout as the 80C32).

In 1997, Philips also introduced the XA-S3 (the 'superchip'), which features many on-board peripherals (see **Table 1**). At that time, there was also talk of an XA-SCC model featuring a non-multiplexed address/data bus, as well as an XA-D3 model with a DeviceNet interface (DeviceNet is an industrial bus based on CAN 2.0B).

As far as we know, the XA-SCC model never made it to production. The XA-D3 was succeeded by the XA-C3, where the 'C' indicates the presence of a CAN 2.0B interface. This opened the market for other applications based on this bus (such as CANOpen).

The first microcontrollers in this family were given 'P51XAxxxx' type numbers, while more recent models are called 'PXAxxxx (without the '51'). This change is more significant than might appear at first glance, since the XA core has been partially upgraded to boost its performance, while retaining machine-code compatibility with previous versions.

Quite recently, the XA family has also been extended with models having on-board Flash memory. These are the XA-4 models, which

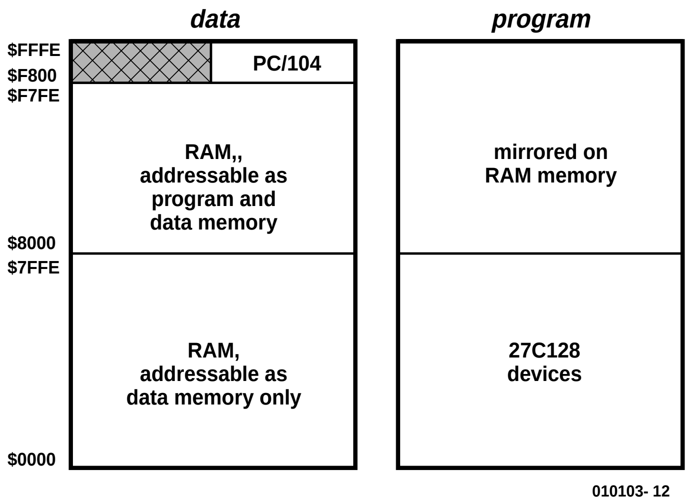



Figure 1. Memory structure of the XA development board.

are based on the XA-xx49 model (the XA-3 models have UV EPROM or are ROMless, which means that they do not have any internal memory).

## Powerful instruction set

The XA family was in principle presented as a 16-bit successor to the 80C32, but XA microcontrollers cannot execute the binary code of their 8-bit cousins. Philips preferred to completely revise the instruction set and offer translation of 80C3x software to XA software at the source code level. This means that programs written for 80C3x microcontrollers must be recompiled before they can be used with XA microcontrollers. This allows the XA core to run much faster than its 8-bit relative using the same source code (without optimisation).

With regard to code optimisation, an important consideration is that the XA family has a register-based architecture. Instead of having only a single accumulator (as in the 80C32x models), which leads to innumerable load and store operations, the XA family has sixteen 16-bit registers labelled R0 through R15, which are fully equivalent at the instruction-set level. The ALU can thus use all of these registers for its computations. If an existing 8-bit application is rewritten to run on an XA microcontroller, it will run many times faster because it needs a lot fewer machine cycles.

Besides this, the XA family can use each of these registers as a data pointer register (memory is partitioned into 64-KB pages). By contrast, the 80C3x family has only the one 16-bit DPTR register, which makes swapping data in RAM a real headache for programmers.

We could say even more about the various capabilities of these microcontrollers, but that would far exceed the scope of this magazine. We recommend that you download the data sheet for the XA-G3 (or the XA-C3) from the Philips website, along with the XA User's Guide, which is *the* reference document for anyone who wants to know how the XA core works.

Anyone fortunate enough to have access to the paper versions of the Philips handbooks should without question order Data Handbook IC25, which bears the title *16-bit 80C51XA Microcontrollers*.

In order to understand the operation of the development card described here, it is important to understand its memory structure. This is illustrated in **Figure 1**.

All data are formatted as 16-bit values, which means that only paired (even-valued) addresses are used. Executable code can be stored in RAM starting at address $8000, since this memory region is accessible for addressing both code and data.

## Table 1: Comparison of XA microcontrollers and the 80C32.

| | 80C32 | XA-G3 | XA-C3 | XA-S3 |
|---|---|---|---|---|
| Core: | 8-bit | 16-bit | 16-bit | 16-bit |
| Architecture: | interrupt | registers | registers | registers |
| 16 bit data/program pointers | 1 16-bit pointer<br>2 8-bit registers | 16 16-bit registers<br>(+page register) | 16 16-bit registers<br>(+page register) | 16 16-bit registers<br>(+page register) |
| Bus: | 8-bit data<br>16-bit address | 8/16-bit data<br>20-bit address | 16-bit data<br>20-bit address | 16-bit data<br>24-bit address |
| Internal ROM (program code): | 4–32 KB | 32 KB | 32 KB | 32 KB |
| External program memory: | 64 KB max. | 1 MB max. | 1 MB max. | 16 MB max. |
| Internal data memory: | 256 bytes | 512 bytes | 1 KB | 1 KB |
| External data memory: | 64 KB max. | 1 MB max | 1 MB max | 16 MB max |
| Clock frequency: | ≤ 32 MHz | ≤ 30 MHz | ≤ 30 MHz | ≤ 30 MHz |
| Clock cycles per instruction | :12–36 | 3–24 | 3–24 | 3–24 |
| On-board peripherals: | 1 UART<br>3 timers | 2 UARTs<br>3 timers | 1 UART<br>1 SPI port<br>1 CAN 2.0B port<br>3 timers | 2 UARTs<br>1 I2C port<br>3 timers<br>8-channel ADC<br>5 PCA I/O |
| Watchdog: | no | yes | yes | yes |

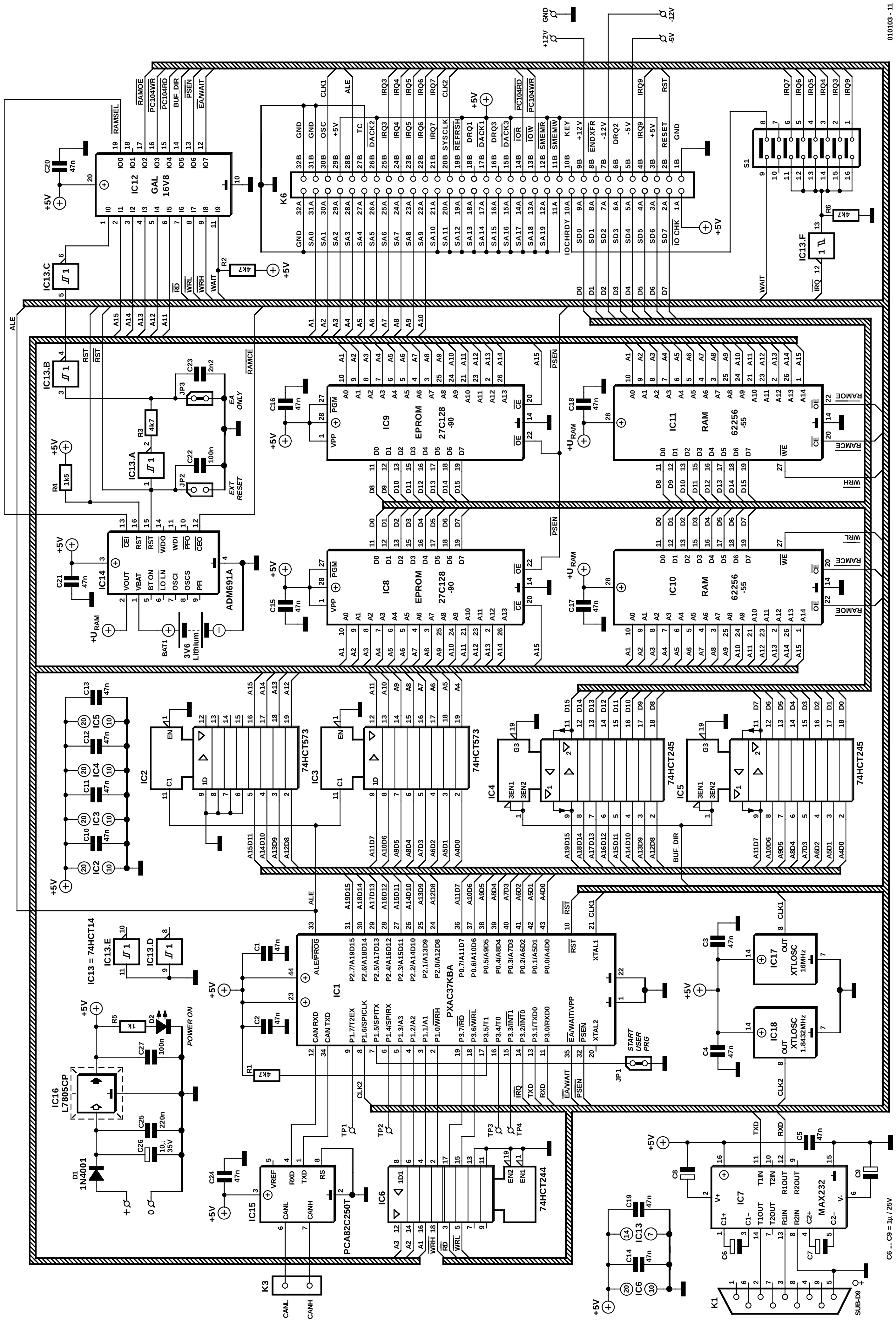



Figure 2. The classic architecture of microcontroller, memory and peripherals can be immediately recognised in the schematic diagram.

## Schematic diagram

A glance at the schematic diagram in **Figure 2** clearly shows that this development board can certainly be regarded as a 'professional' tool. Although the schematic diagram may appear complex, it is not particularly difficult to understand.

The heart of the development board is naturally an XA microcontroller (IC1), which is configured to use 16-bit data and 20-bit addressing. This design is suitable for use with XA-G3 and XA-C3 microcontrollers, without any need for modifications.

The microcontroller is clocked by a DIP oscillator (IC17) connected to the XTAL1 input. Any desired frequency can be used, as long as it is within the allowable range for the microcontroller.

In order to make the data transmission rate of the serial port independent of the primary clock frequency, the monitor configures Timer 2 as the clock source for the serial port. This explains the presence of a second oscillator (IC18). If you don't use the monitor (for instance, if you use your own program), or if you simply don't need a serial port, you can dispense with the second oscillator and release line P1.6 for other uses.

With regard to the serial interface, a short description suffices. IC7 is a 'classic' MAX232, which converts TTL levels into V24 levels and *vice versa*. This IC has been described countless times in *Elektor Electronics*, so we do not have to say anything more about it here.

The address bus here is 20 bits wide, which corresponds to the default startup value of the XA microcontroller. If you examine the schematic closely, you will see that the A0 address line is missing; it is replaced by the $\overline{\text{WRL}}$ line with the 16-bit data configuration. The addresses are available at ports P0 and P2, multiplexed with the data.

There are two 74HCT573 address demultiplexers (IC2 en IC3), which are driven by the ALE line of the XA IC. IC3 and IC3 also buffer the address lines, which is necessary due to the large number of ICs loading the address bus (not to mention the PC/104 interface). The buffers also protect the microcontroller in the event of a problem on the bus (such as a short circuit during an experiment).

It should be noted that the A1 and A3 lines are not multiplexed by the XA IC. This allows burst addressing, to be used, which reduces the number of required clock cycles. As with the other address lines, a buffer (IC6) protects the microcontroller.

The data bus is buffered by IC4 and IC5, for the same reasons as for the address bus. The control signals ($\overline{\text{RD}}$, $\overline{\text{WRH}}$ and $\overline{\text{WRL}}$) are buffered by the second half of IC6.

Readers familiar with the 80C3x will have already seen that the pinout of the XA ICs shows many similarities to that of its 8-bit cousins (except for the 16-bit bus). However, it should be noted that there are certain subtle differences, which are nevertheless quite important.

## Bus configuration

The first difference relates to the bus widths, which can be configured dynamically. XA microcontrollers can work with a data bus width of 8 or 16 bits and an address bus width of 12, 16, 20 or 24 bits. Naturally, the hardware environment of the microcontroller must match the selected configuration, since the allocation of the pins to the various bus lines depends on the configuration. Here the development card has been designed for a 16-bit data bus, so the microcontroller must start up in this mode, but how does it know which mode to use?

Port P3.5 (T1/BUSW) handles this task. After a reset, the XA IC tests the value on this port. If it is 0, the microcontroller switches to the 8-bit mode. Otherwise, with a value of 1 on the input of P3.5, the XA IC is configured in 16-bit mode. For this reason, pin P3.5 is tied to +5 V via R1. Once the microcontroller is running, the T1 pin can be used as a normal I/O port, but it is important that it has a logic 1 value during the reset phase. Note that the XA-C3 also includes the BUSW function, but it only runs in the 16-bit mode!

## External memory usage

The second difference relates to the $\overline{\text{EA}}$/WAIT line. Just as for the 80C3x, the $\overline{\text{EA}}$ line indicates to the microcontroller that the executable code is entirely located in external memory ('EA' stands for 'external addressing'). The level on the $\overline{\text{EA}}$/WAIT pin is read by the XA IC during the reset phase. If it is a logic 0, the microcontroller looks for the program code exclusively in external memory (in this case, EEPROMs IC8 and IC9), even if the microcontroller is an XA-G37 or XA-C37 with internal ROM. Naturally, the level on this pin *must* be 0 for an XA-C30 or XA-G30, since these types do not have any internal program memory.

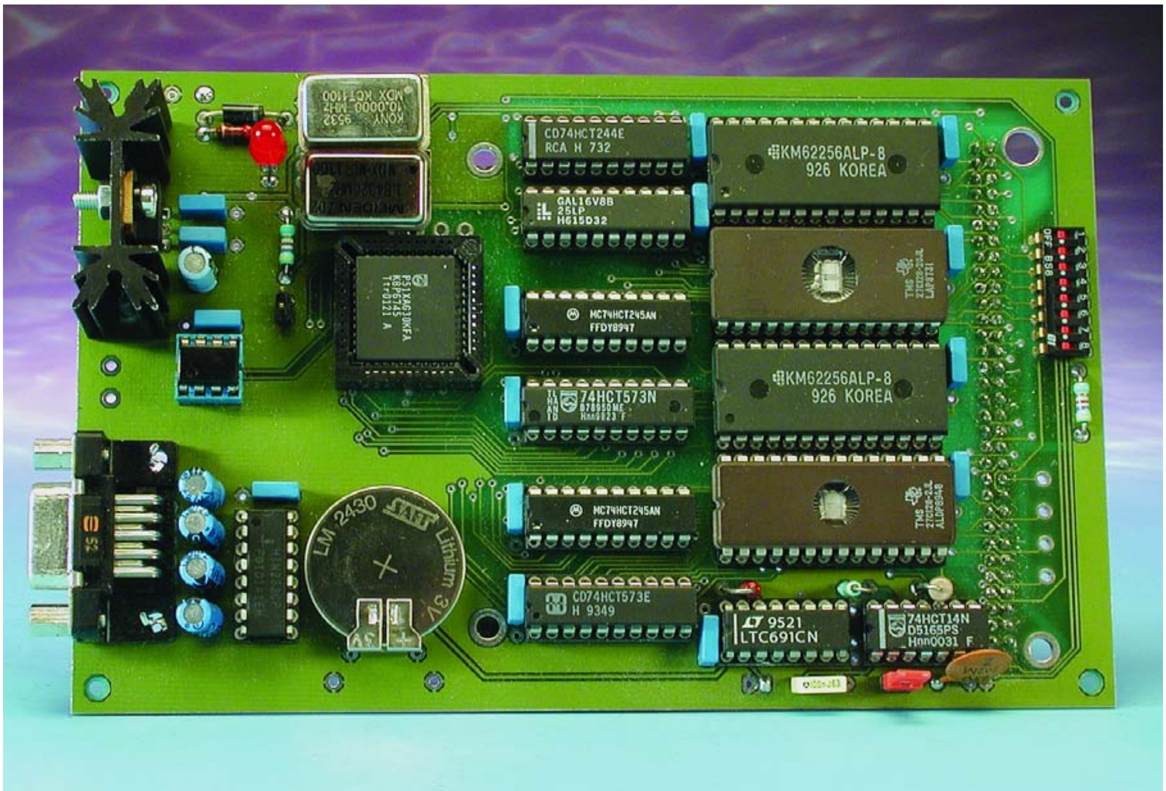
If $\overline{\text{EA}}$ has a value of 1 during the reset phase, the XA IC will start executing the program stored in its internal ROM. The external memory will not be used unless the called address is outside the address range of the internal memory (refer to the description of the Bus Disable function in the Philips documentation).

The problem with the XA is that the $\overline{\text{EA}}$ and WAIT functions share the same pin. The latter function, which is not present in the 80C3x, allows the XA processor to be 'frozen' during an access to a slow peripheral. The 80C3x was never blazingly fast with regard to bus speed, so problems with slow peripherals were practically unknown with such microcontrollers.

In contrast to their 8-bit cousins, XA microcontrollers have an instruction pipeline, which allows instructions to be fetched in a single clock cycle (opcode fetch). At maximum speed, the duration of one bus cycle for an XA microcontroller clocked at 30 MHz is barely 33 ns (or 66 ns if an ALE cycle is necessary). Very few peripheral devices can support such short access times. In order to avoid any problems with slow peripherals, the speeds of external accesses can be configured using the BTRH and BTRL registers. (Note: with an XA-C3 device, these registers must be initialised to standard values. The MIFBTRH and MIFBTRL registers should be used instead, with a compression factor between 1 and 5.)

It would be a shame to reduce the overall speed of a system just because one of its peripherals is too slow. This is where the WAIT function comes into play. When accessing the external bus, the XA processor tests the value of the $\overline{\text{EA}}$/WAIT pin. If it remains at 0, the bus cycle is executed normally according to the speed configured in the BTRH and BTRL registers. On the other hand, if the WAIT pin has a value of 1, the bus cycle is frozen as long as the pin remains at this level. It can thus be prolonged indefinitely for the slowest peripheral, while still permitting full-speed access to fast peripherals.

This approach is extremely effective, but it




has one drawback. Consider a system using an XA-C37 or XA-G37 (with internal EPROM). To cause the program stored in internal memory to be executed, the level on the $\overline{\text{EA/WAIT}}$ pin must be 1 during the reset. As long as the XA processor only accesses internal memory, everything is fine, since the level on the WAIT pin does not affect internal accesses. However, as soon as the XA processor attempts to access external memory, it will test this pin, and since it has a level of 1, the cycle will be prolonged indefinitely (because the level on $\overline{\text{EA/WAIT}}$ is 1). The system will thus be stuck.

Since the development board can also accommodate XA models with internal EPROM, an small circuit built around IC13 (gates A, B & C) is necessary to hold the level on the $\overline{\text{EA/WAIT}}$ pin at 1 for several milliseconds following a reset, which is the interval during which the XA processor tests the level on this pin. After this interval expires, the level on this pin returns to 0 in order to avoid interference with the WAIT function. Incidentally, we would like to point out that it is not essential to use the WAIT function in a system, since the same effect can be achieved in software using the WAITD bit.

If the microcontroller is a ROMless type, or if it is desired to use external addressing with an XA-C37 or XA-G37, jumper JP3 must be fitted to hold the $\overline{\text{EA}}$ line at a 0 level during the reset.

## The PC/104 bus

We haven't completely lost the WAIT signal, since it is passed through the GAL chip (IC12). In our case, this function is tied to the IOCHRDY line of the PC/104 bus, which is specifically intended to be used for controlling slow peripherals. Here we would like to point out that it is essential that any PC/104 card installed in the system maintains the IOCHRDY line at a 1 level during the reset phase, since otherwise the operation of the $\overline{\text{EA/WAIT}}$ function will be impaired (any other behaviour would be abnormal and not in conformance with the PC/104 specifications, but we have already seen such a situation in one of our systems, so consider yourself forewarned). If your have any doubt regarding the behaviour of a particular PC/104 card in this regard, the IOCHRDY signal can be isolated using DIPswitch S1-8.

## Jumpers

With regard to jumpers, we should say something about JP1, which is connected to P1.5. This does not relate to any special feature of the XA family, but instead serves as a simple expedient to allow the capabilities of the development board to be extended. Whenever the monitor (written by us) starts up the XA-G3, it tests the level on this line. If it is high (due to the internal pull-up resistor in the XA IC), the monitor takes control and enters the dialogue mode in order to process external commands.

If JP1 (START USER PROG) is fitted (P1.5 tied to ground), the monitor disables itself and jumps immediately to the user reset vector, which we have placed at address $8000 (which means in RAM), in order to start executing the user program in the autonomous mode. We would like to emphasize that if the development board is fitted with EPROMs that do not include our monitor, but instead contain a dedicated application, jumper JP1 serves no function, and port P1.5 can be used in the normal manner.

JP2 is not a jumper, although the schematic shows it as a jumper. Instead, it provides a connection point for an external reset signal (EXT RESET). The LT691 (IC14) does not provide this capability. Incidentally, the LT691 data sheet recommends putting a 100-Ω resistor in series with the reset switch – not to limit the short-circuit current, but to prevent oscillations. Although this resistor is not shown in the schematic diagram, we strongly recommend that you use it if you decide to fit an external reset switch.

Note that an external reset switch connected to JP2 will only affect the XA microcontroller, but not any PC/104 card connected to the system. In order to effect a complete reintialisation (XA and PC/104), it is

## Table 2:

**Principal features of the XA Development Board.**

– Compatible with XA-G3x and XA-C3x microcontrollers
– Compatible with EPROM versions (XA-G37 / XA-C37) and ROMless versions (XA-G30)
– 16-bit data bus
– Monitor in EPROM
– 32 Kword 16-bit program/data memory region in EPROM
– 32 Kword 16-bit data region
– 32 Kword 16-bit data/program region for debugging
– PC/104 interface for I/O access, allowing standard PC/104 cards to be connected (for digital and analogue I/O, network interfaces, etc.)
– Code can be executed from RAM
– Page-0 addressing (16-bit) for maximum XA core speed
– Can be used as a development board or an expansion card
– Battery-backed RAM

necessary to switch the power off and back on. This is a peculiarity of the LT691 that we have to live with.

## Using the PC/104 bus

Now we come to the PC/104 interface. Since XA microcontrollers are not x86 microprocessors, it is not possible to implement the full functionality of the PC/104 specification. Here we only have access in the I/O mode, in this case to the first 2048 bytes. PC experts will doubtless point out that only the first 1024 bytes are used in PCs, but there are PC/104 cards available with a larger address range, which explains our choice.

Since the XA microcontroller does not have a specific I/O addressing mode, it accesses the expansion bus via a memory window located between addresses $F800 and $FFFF in the data region of the processor's address space. Access to the PC/104 bus is controlled by the $\overline{\text{PC104WRT}}$ and $\overline{\text{PC104RD}}$ signals, which are generated by the GAL chip (IC12). Note that only the least significant byte is transferred to the PC/104 bus!

The development board does not support DMA access (due to the absence of suitable capabilities in the XA microcontroller), but it can handle interrupts IRQ2 (sometimes called IRQ9) through IRQ2 from the PC/104 bus. Although the XA chip has two inputs for external interrupts ($\overline{\text{INT0}}$ and $\overline{\text{INT1}}$), only one of these is assigned to the PC/104 bus. Switch S1 is used to select a particular interrupt line.

This approach means that only one PC/104 card can be installed with interrupt support (the number of cards that can be installed without interrupt support is limited only by the capacity of the bus drivers). The polarity of the interrupt signals on the PC/104 bus is the opposite of what is conventional for XA microcontrollers, so gate IC13F sets things right.

It is not possible to access the PC/104 bus in memory mode. This is not particularly important, since most PC/104 expansion cards use only I/O mode. Nevertheless, you should verify this before fitting a PC/104 card to the development card, in order to avoid potential problems.

Our final comment with regard to the PC/104 bus is that the development card only provides +5-V power on the connector. Some PC/104 cards also require +12 V, –5 V and –12 V. Since these voltages are not present on the development card, connectors K4 and K5 are provided for connection to an external power supply if necessary.

## Reset circuit

Let's go back to the LT691 (IC14), whose main function is generating the power-on reset pulse for the card. For this purpose, it has an internal comparator that indicates whether the supply voltage has dropped below 4.75 V. If this happens, the Reset and $\overline{\text{Reset}}$ signals are automatically generated and maintained until the voltage again rises above 4.75 V. In practice, the reset pulse is maintained for 35 ms after the voltage stabilises.

IC14 also contains a highly useful power supply switchover circuit, which can be used to maintain the RAM supply voltage using the 3.6-V lithium battery BAT1. This circuit is automatically activated whenever an out-of-range supply voltage is detected.

IC14 also inhibits the control signals for the RAM during the reset phase, in order to prevent corruption of the content of the RAM. This allows program code or data to be saved in the RAM, even in the absence of external power. Since the monitor checks the P1.5 line on startup, the system can be made fully autonomous without any need for specific code for this purpose in EPROMs IC8 and IC9. This is very useful during program development. Naturally, if the card is used for a specific application (with an application program in EEPROM), data for this application can also be saved in the RAM.

The LT691 family also includes other functions, such as an integrated watchdog and power-fail inputs and outputs, but they are not used here. Finally, we would like to point out that this IC is available from several different manufacturers under various type numbers, such as ADM691, LT691, MAX691, etc.

## CAN bus and power supply

IC15 is a TTL/CAN converter, which is only necessary if the board is fitted with an XA-C3 microcontroller (the XA-G3 does not have a CAN interface).

Finally, we come to the power supply, which is certainly not the least important part of the schematic. Since the board only needs a single stabilised 5-V supply, little more is needed than a simple 7805 voltage regulator. Diode D1 is included to provide reverse-polarity protection, so it should not be mistakenly be assumed that the circuit can be powered from an AC source. A mains adapter supplying at least 9 V DC should be connected to K2.

We would also like to emphasize that numerous decoupling capacitors are fitted to the board (one per IC), and that these capacitors are essential. Although the schematic shows a value of 47 nF for these capacitors, a value of 100 nF can be used without hesitation, and this value may be more readily available.

This concludes the description of the circuit. The construction and use of the development board will be described next month in Part 2 of this article. To whet your appetite, **Table 2** provides a condensed summary of the principal features of the XA development board.

(010103-1)

# DMX 3-in-1

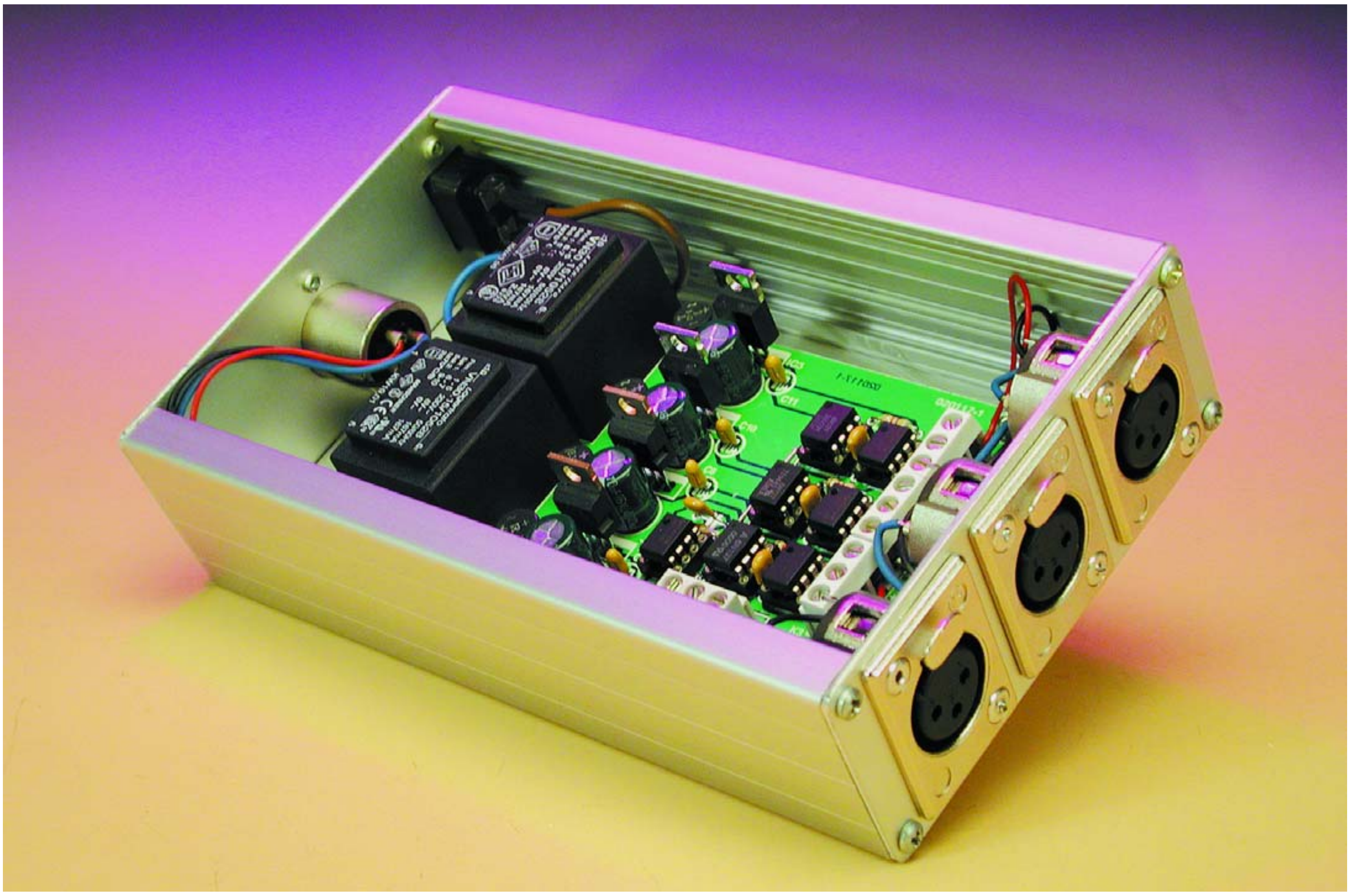
## Splitter, Isolator, Repeater

Design by J. Mack

**This project dubbed 'DMX 3-in-1' is an active splitter that distributes a single DMX signal across three outputs. In addition it features full electrical isolation between input and outputs, while any of the three outputs can act independently of the others.**

The DMX 3-in-1 splitter/isolator/repeater can be used with all equipment that supports DMX512, and will help to increase its reliability in actual operation. The unit will also prove useful for quick re-cabling of equipment, because it allows the (physically) shortest link to be created.

An example may help to illustrate the point: let's assume there's DMX equipment at the left-hand as well as the right-hand side of the stage.



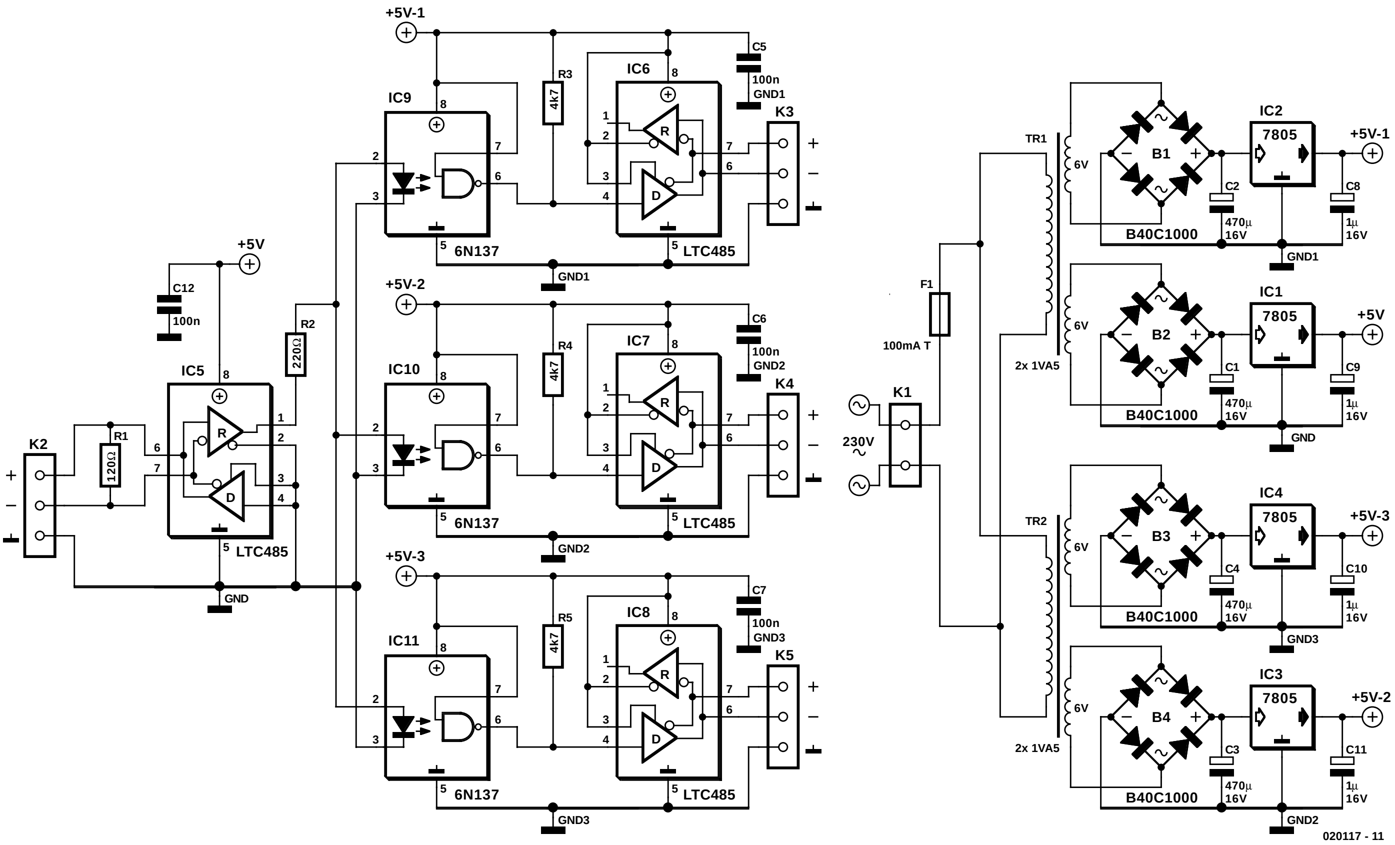



Figure 1. The circuit diagram of the DMX 3-in-1 comprises four electrically isolated subsections.

Without the splitter, it would be awkward to feed one cable to one side only, then take it back all the way to the origin, then on to the other side, and then from equipment to equipment. Arguably, this is not the best solution to distributing DMX signals across equipment.

The DMX splitter makes the cabling job much easier. Only one cable is required at each side, where it is used to interconnect all equipment.

## Repetitia Placet

A quick glance at the circuit diagram in **Figure 1** shows that the DMX 3-in-1 consist of three identical sub-circuits. Resistor R1 at the input of the circuit acts as a terminator ('terminating resistance') for the DMX bus, which, as you may know, works in compliance with well established RS485 standards. The DMX signal is first converted to TTL by level converter IC5. The LTC485 from Linear Technology is an interface component that translates RS485 signals into TTL (receiver, R), and vice versa, TTL into RS485 (driver, D). In this circuit the transmitter section of the chip has been disabled. The datasheets of this interesting chip may be found at
www.linear-tech.com/pdf/485ff.pdf

Behind IC5, the signal is split across three identical circuits. High-speed optocouplers type 6N137 guarantee electrical isolation. Resistor R2 limits the current through the LEDs inside the optocouplers. The optocouplers are enabled by means of a High level at pin 7. Pull-up resistors (R3, R4, R5) are required at their outputs which are of the open-collector (OC) variety.

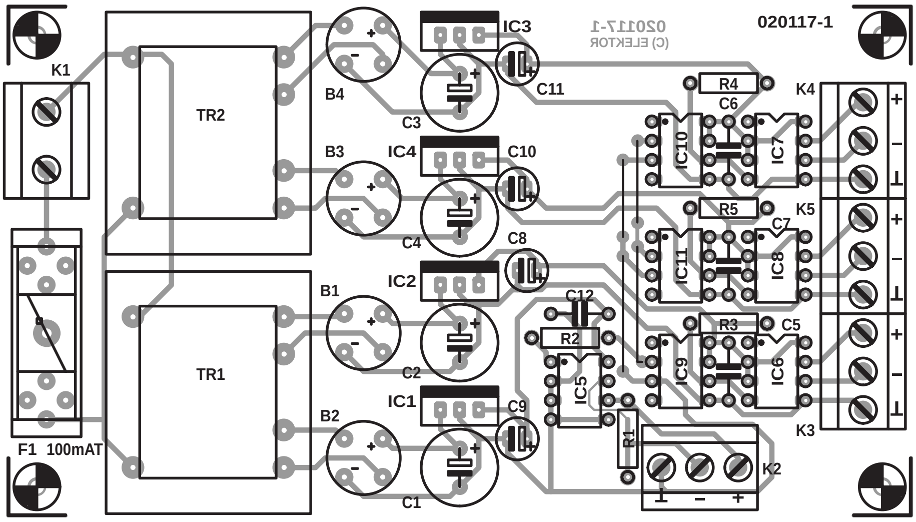
Next, the signal once again meets an LT485. This time, however, only the transmitter (D) is employed. A small difference may be noted with respect to IC5: Because of the inverting action of the optocouplers, the outputs of the LTC485's are transposed to prevent unwanted inversion.

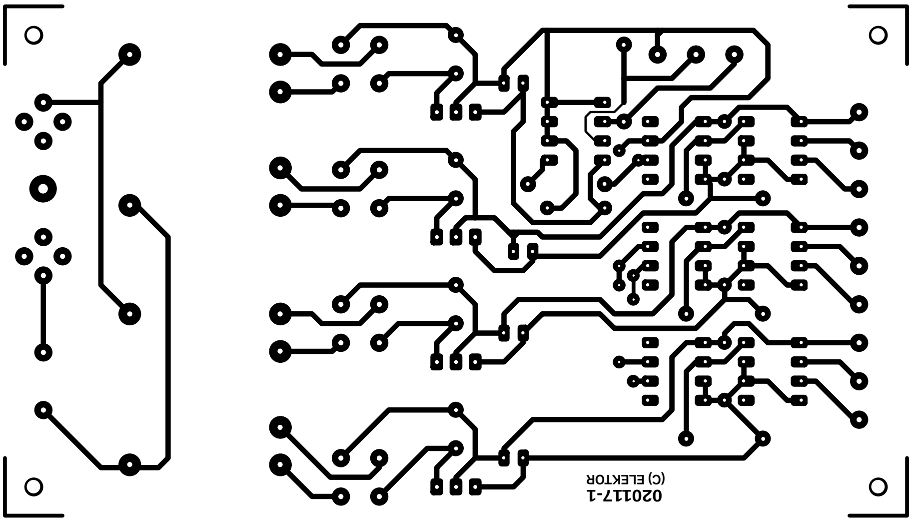
The tripled DMX signal leaves the circuit via connectors K3, K4 and K5. The electrical isolation that exist between the three outputs will prove invaluable when errors occur. An error on one DMX line will only cause one output to fail instead of the complete unit — the other outputs will continue to operate normally!

The power supply is seen in the right-hand section of the circuit diagram. Because we are dealing with four electrically isolated sub-circuits in the DMX512 path, there is a requirement for four independently operating power supplies. Although this requirement could be fulfilled using a mains supply incorporating special DC/DC converters, the current consumption of the circuit is so low that two 1.5-VA transformers, each having two secondary windings, are probably cheaper and less demanding in respect of board space, albeit less elegant. Each of the four secondaries is followed by a small rectifier bridge feeding a three-pin fixed voltage regulator type 7805.

## Construction

The artwork for the printed circuit board (**Figure 2**) shows a dead conventional project when it comes to building it. Ready-made printed circuit boards for the project may be obtained from **The PCBShop** (i.e., they are not available through Elektor's Readers Services). Do not forget to fit all 4 (four) wire links on the



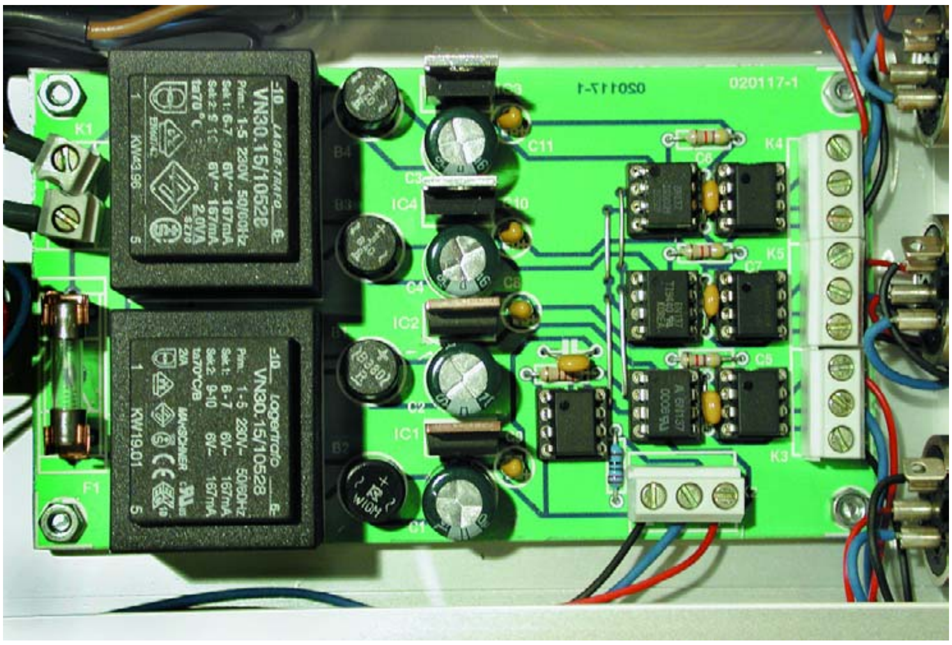



Figure 2. Copper track layout and component mounting plan (board available from The PCBShop).

board and be sure to mount all ICs with the correct orientation. IC sockets are recommended, but not obligatory.

The enclosure that holds the DMX 3-in-1 must be 'stage, bullet and fool-proof' as well as electrically safe, complying with Class-2 electrical isolation requirements. In particular, the distance between mains-carrying parts on the PCB and the metal case should be 6 mm minimum. This rules out the use of 5-mm PCB spacers. The XLR chassis-mount sockets/plugs and the IEC appliance socket are best secured to the case with rivets.

(020117-1)

## COMPONENTS LIST

**Resistors:**

R1 = 120Ω
R2 = 220Ω
R3,R4,R5 = 4kΩ7

**Capacitors:**

C1-C4 = 470µF 16V radial
C5,C6,C7,C12 = 100nF
C8-C11 = 1µF 16V radial

**Semiconductors:**

B1-B4 = B40C1000, round case (40V piv, 1A)
IC1-IC4 = 7805
IC5-IC8 = LTC485CN8 or SN75176BP
IC9,IC19,IC11 = 6N137

**Miscellaneous :**

F1 = fuse holder, PCB mount, with fuse 100mA(T) (slow)
K1 = 2-way PCB terminal block, lead pitch 7.5mm
K2-K5 = 3- way PCB terminal block, lead pitch 5mm
Tr1,Tr2 = mains transformer, 2x6 V secondaries, 1.5VA or 2 VA
1 off IEC mains appliance socket
3 off XLR socket (female), 3-way, chassis mount, 180 degrees
1 off XLR plug (male), 3-way, chassis mount, 180 degrees
PCB, available from **The PCBShop**

# Valve Final Amp (2)

## Part 2: printed circuit boards and construction

Design by Bob Stuurman

**This final amp is easy to build. The stereo version essentially consists of two amplifier boards, a power supply board for the high voltage and negative grid voltage, two output transformers and a power transformer. We have designed two printed circuit boards for building the final amp, but it can also be constructed in the 'old-fashioned' manner using solder turrets.**


The chassis is made of aluminium and consists of two parts: an open-ended U-shaped channel section and a flat plate resting on top of the channel section. The channel section is fitted upside down, with the output transformers on top and the power transformer underneath. The combined weight of the transformers alone is more than eight kilos, and using a channel section gives the chassis adequate stiffness.

The rear wall of the channel section is aligned with the rear edge of the plate. All of the connectors are mounted on the rear wall, along with the master volume control. An IEC appliance socket with integrated filter, switch and fuse holder is used to keep the 230-VAC wiring to a minimum. There is no need for a pilot lamp, since the valves glow nicely when the amplifier is on.

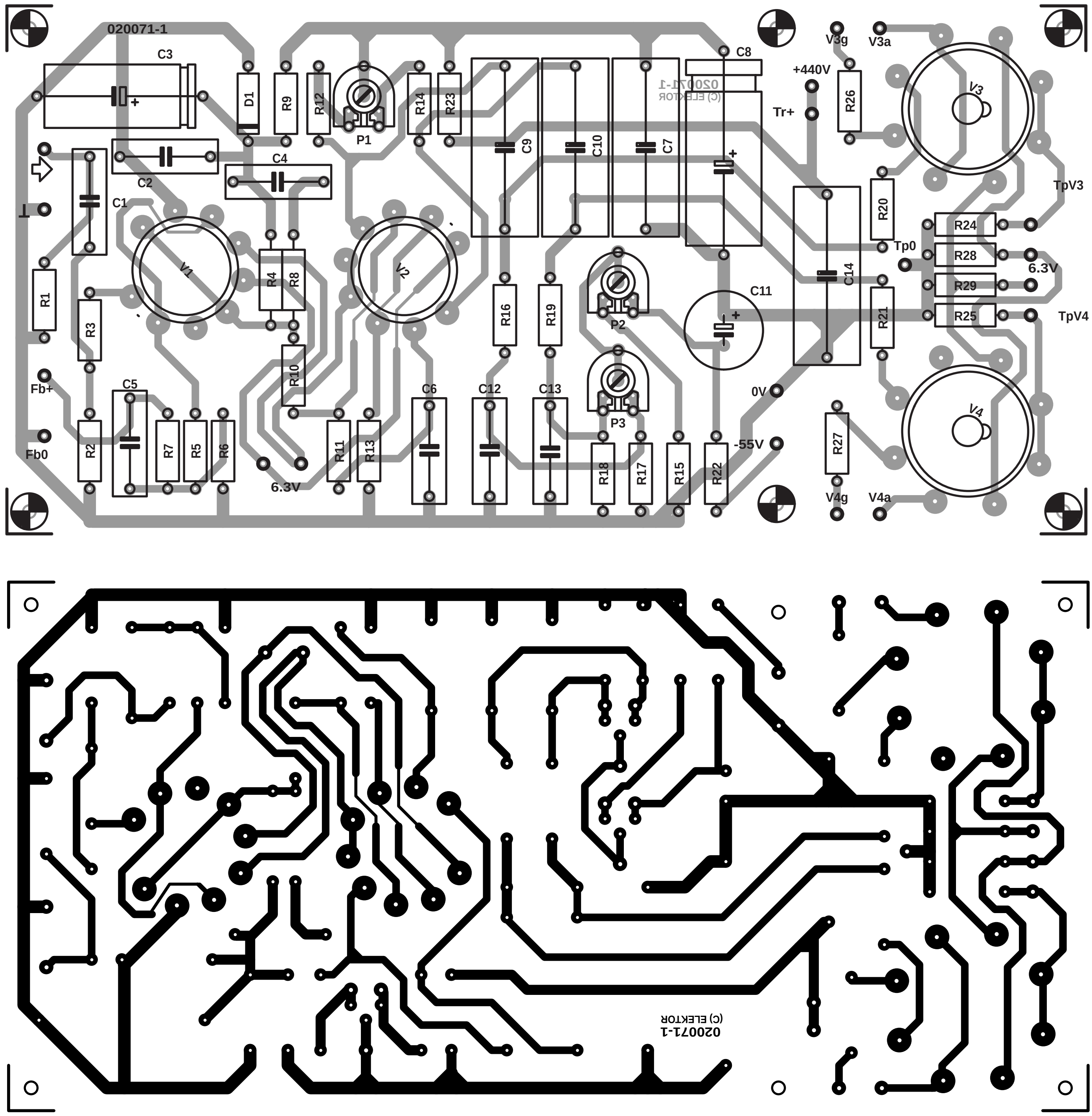



Figure 1. Copper layout and component layout of the printed circuit board for one amplifier channel.

## Safety precautions

Hazardous voltages are present in this amplifier. The electrolytic capacitors in the power supply have a large capacity, so it takes quite a while for the high voltage to drop to a safe level after the amplifier is switched off. For this reason, you should connect two 230-V 15-watt incandescent lamp bulbs in series across the high voltage while the amplifier is being tested. As soon as the mains voltage is switched off, they will discharge the electrolytic capacitors in a few seconds, and they will have practically no effect on the operation of the amplifier.

## Amplifier construction

The copper track and component layouts of the amplifier printed circuit board are shown in **Figure 1**. The only component that is not included on the board is the output transformer. The circuit board is single-sided, and using the artwork shown here (and available from our website) some of you will be able to make it themselves. However, the board it is also available ready-made from Readers Services (order number **020071-1**). Two of these boards will be needed for a stereo version of the amplifier.

All connections to the circuit boards are

## COMPONENTS LIST

### Amplifier (one channel)

**Resistors:**

All fixed resistors: metal film, Beyschlag type MBE0414 or BC Components type PR-02, dim. 4x12 mm.

R1,R2,R11 = 1MΩ
R3 = 4kΩ7
R4,R17,R18 = 47kΩ
R5 = 390Ω
R6,R22,R28,R29 = 100Ω
R7 (LS = 8Ω) = 3kΩ3
R7 (LS = 4Ω) = 2kΩ2
R8 = 27kΩ
R9 = 100kΩ
R10,R26,R27,R30 = 1kΩ
R12,R14 = 150kΩ
R13 = 82k
R15 = 15kΩ
R16,R19 = 390kΩ
R20,R21 = 2kΩ2
R23 = 10kΩ
R24,R25 = 10Ω
P1 = 50kΩ preset
P2 = 10 k preset
P3 = 20 k preset
(All presets: Bourns type 3386P)

**Capacitors:**

All film capacitors: Wima type MKS4, unless indicated otherwise.

C1 = 470nF 100V, lead pitch 15mm
C2 = 100nF 400 V, lead pitch 15mm
C3 = 10µF 350V or 450V, axial, dim. 12x25 mm
C4 = 100pF 630V, polypropylene, dim. 5x11 mm
C5 (for LS 8Ω) = 680pF 630V, polypropylene, dim. 5.5x15 mm
C5 (for LS 4Ω) = 1000pF 630V, polypropylene, dim. 5.5x15 mm
C6,C12,C13 = 220nF 250V, lead pitch 15mm
C7,C14 = 470nF 630V, lead pitch 27.5 mm
C8 = 10µF 450V, axial, dim. 15x30 mm
C9,C10 = 100nF 630V, lead pitch 22.5 mm
C11 = 470µF 63V, radial, dim. 12.5x25 mm

**Semiconductors:**

D1 = 200 V 1.3 W zener diode

**Valves:**

V1 = EF86 (US: 6267)
V2 = ECC83 (US: 12AX7)
V3,V4 = EL34 (US: 6CA7), matched

**Miscellaneous:**

2 noval (9-way) valve sockets, ceramic
2 octal (8-way) valve sockets, ceramic
Tr1 = output transformer, Lundahl type LL1620 P-P
PCB, order code **020071-1** (see Readers Services page)

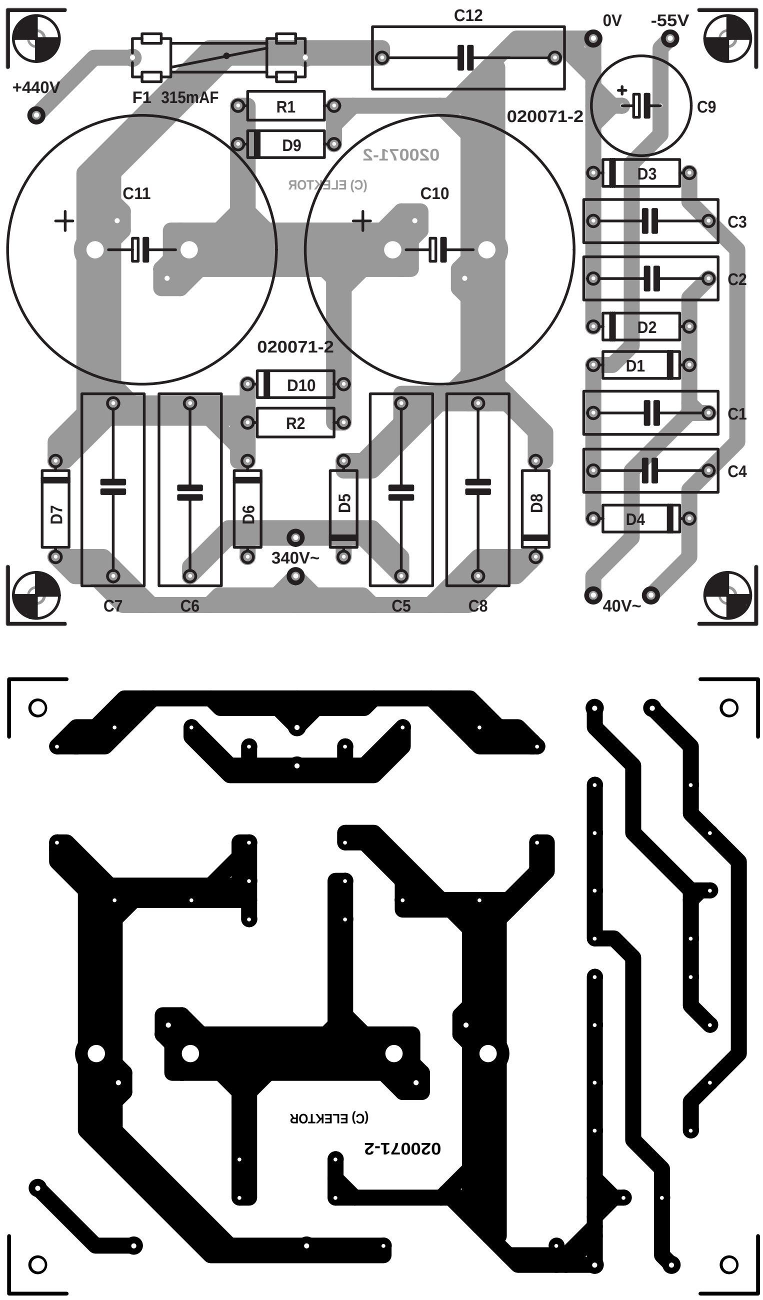


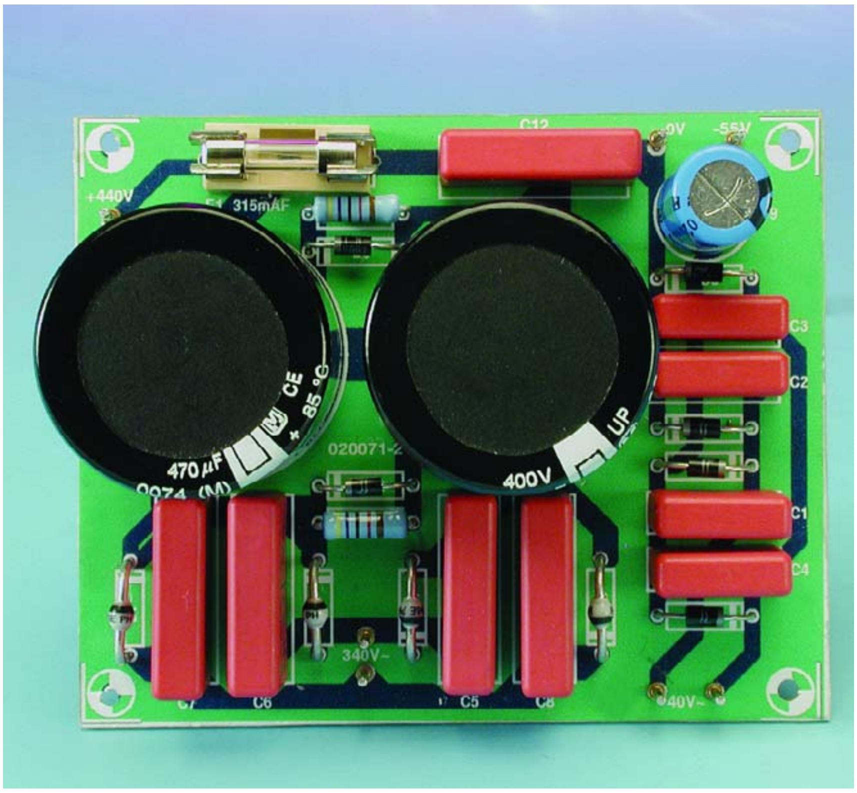

Figure 2. Copper layout and component layout of the printed circuit board for the power supply.

made using solder posts with a diameter of 1.3 mm and matching connectors. Noval valve sockets are used for V1 and V2. These sockets are available in plastic and ceramic versions; the circuit board has been designed for the ceramic version.

Ceramic octal sockets are used for V3 and V4, the EL34s. They have solder tabs with a width of 2 mm and a thickness of 0.5 mm. In order to allow the sockets to be fitted flat against the circuit board, the drilled holes for the solder tabs must be widened somewhat by (mis)using a circuit board drill as a routing bit.

The circuit board has six mounting holes, which allow it to be firmly attached to the base plate. This provides extra support for the portion holding the output valves.

If you stick to the parts shown in the components list, building the printed circuit board is a breeze; everything fits perfectly. The PR-02 resistors from BC Components (formerly Philips) are 1% types and have four colour-coding bands.




Since it can be difficult to read their values from these bands, it's a good idea to always check them with an ohmmeter.

The valve sockets are soldered to the copper side of the circuit board. In order to align the individual contacts properly while soldering them in place, you should insert the valves in the sockets. When fitting the octal sockets, be careful to orient the notches properly. The sockets will 'fit' in all possible orientations, and it's next to impossible to remove a socket once it's been soldered in place.

The single-sided printed circuit board for the power supply (**Figure 2**) is available from Readers Services under order number **020071-2**. Here again, 1.3-mm solder posts with matching connectors are used. Building the power supply board is so simple that we don't need to say anything about it, except to remind you to watch the polarity of the diodes and electrolytic capacitors.

## Building the amplifier

The dimensions of the chassis plate and channel section are shown at the lower left of the wiring diagram (**Figure 3**). The channel section is made from a piece of aluminium sheet 370 mm long and 290 mm wide, with its long edges folded to form a U-shaped channel with 80-mm walls.

The convenient feature of this chassis is that the channel section and plate can be prepared separately. However, some of the holes must be made in both the channel section and the plate, which requires the two parts to be temporarily bolted together. For this purpose, you can drill holes for 2-mm screws inside the outlines of the transformer covers.

For the next stage, you will need paper templates, preferably made from tracing paper. The templates for the amplifier boards and the power supply board can be made by simply copying the component layouts, since they show the dimensions of the circuit boards and the locations of the mounting holes. For the output transformers and their covers, you will have to make a drawing showing the outside dimensions (of the cover) and the locations of the drilled holes. The template for the power transformer consists of a circle and its centre point. Make templates for

## COMPONENTS LIST

### Power supply

**Resistors:**
R1,R2 = 47kΩ (Beyschlag type MBE0414 or BC Components type PR-02, dim. 4x12 mm)

**Capacitors:**
C1-C4 = 100nF 400V, lead pitch 15 mm
C5-C8 = 100 n/1000 V, lead pitch 22.5 mm
C9 = 470µF 63V, radial, lead pitch 5 mm, dim. 12.5x25 mm
C10,C11 = 470µF 400V, radial, lead pitch 10 mm (e.g., Roederstein series EYS)
C12 = 100nF 630V, lead pitch 22.5 mm

**Semiconductors:**
D1-D4,D9,D10 = 1N4007
D5-D8 = BYW96E

**Miscellaneous:**
Fuse, 315 mA (fast) with PCB mount holder
Mains transformer, secondaries 340V at 0.7A, 6.3V at 6.8A and 40V at 0.1A (Amplimo # 7N607)
PCB, order code **020071-2**

## Miscellaneous parts

IEC mains appliance socket with integral filter, switch and fuse holder, fuse 1.5A(T) (time lag)
2 NTC-resistors, 5 Ω 5 W (Amplimo or Conrad Electronics)
Audio potentiometer, 100kΩ stereo, logarithmic law (e.g., Alps type RK-27112) with knob
2 cinch sockets, chassis mount (isolated)
2 binding posts, red (isolated)
2 binding posts, black (isolated)
Terminal block strip
Covers for output transformers

## SUGGESTED SUPPLIERS

**Lundahl transformers**
Lundahl Transformers AB, Tibeliusgatan 7, SE-761 50 Norrtälje, SWEDEN. Tel. +46 176 139 30, Fax +46 176 139 35.
Distributor overview at www.lundahl.se

**Valves and valve sockets**
Chelmer Valve Co. (www.chelmervalve.com),
Conrad Electronics (www.int.conradcom.de),
Amplimo (www.amplimo.nl)

**PR-02 resistors**
Farnell (www.farnell.co.uk),
C-I Electronics (www.dil.nl)

**MKS capacitors**
Farnell (www.farnell.co.uk),
C-I Electronics (www.dil.nl),
Conrad Electronics (www.int.conradcom.de)

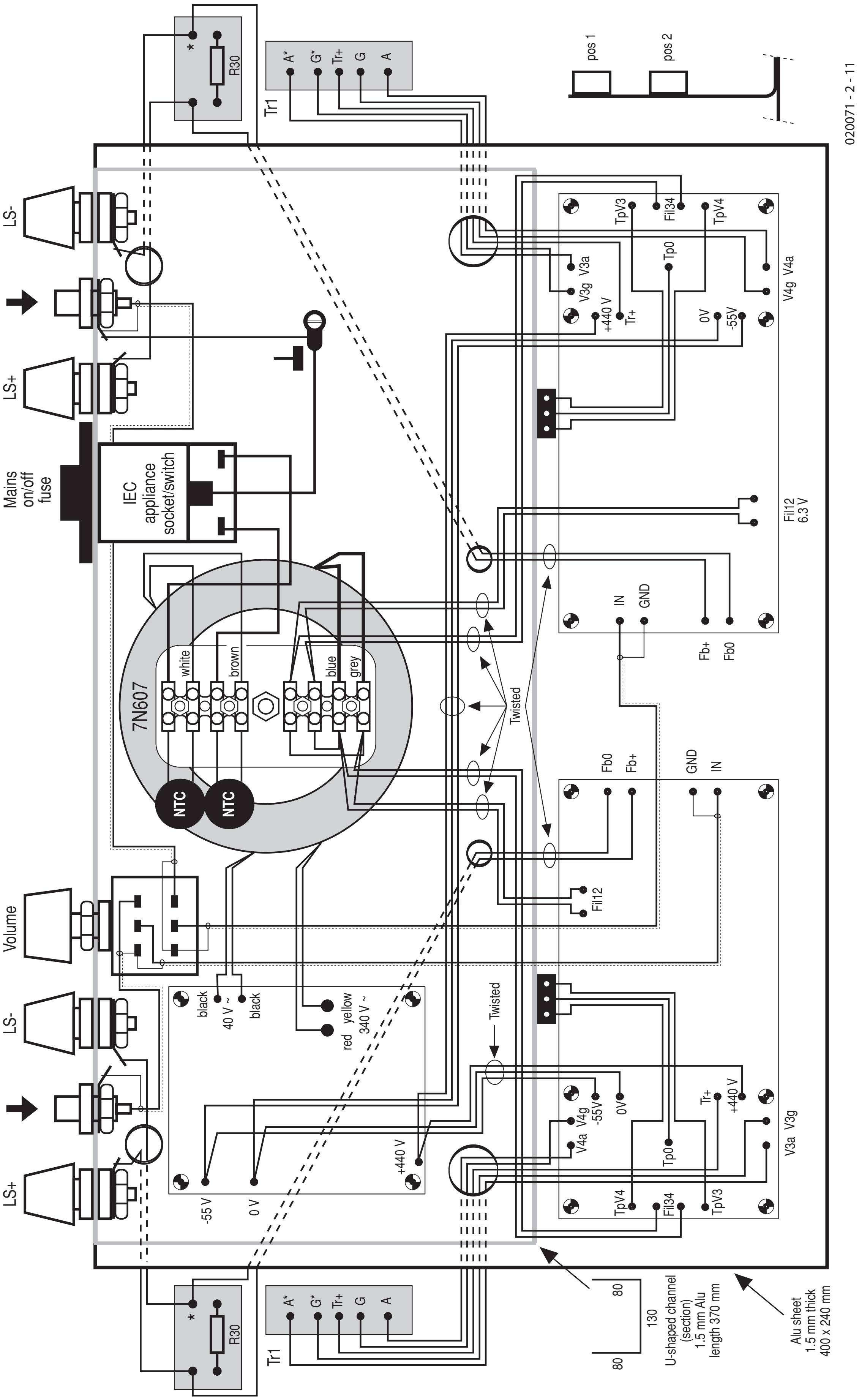
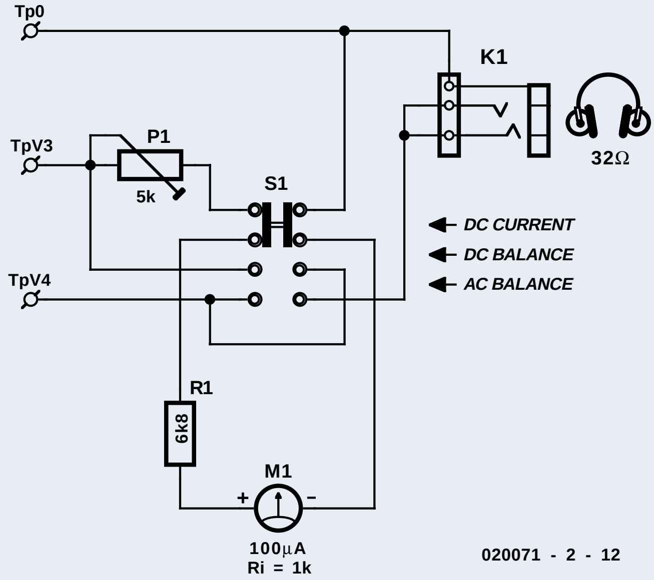



Figure 3. Sample wiring diagram and mechanical layout (bottom view) for a stereo amplifier.

# Alignment




An 8-Ω or 4-Ω load, as appropriate, must be connected to the loudspeaker output of the amplifier during alignment and whenever measurements are being made on the amplifier. Several power resistors attached to a heat sink can be used for this purpose. If the amplifier is not loaded, arcing can occur in the output transformer, possibly resulting in a defective transformer.

The output valves are not self-biasing, since a negative grid voltage is used instead of cathode bias resistors. Consequently, they should preferably be purchased as matched pairs.

The following items must be aligned in the order listed: DC current, DC balance and AC balance. The characteristics of the valves change as they age, so it is advisable to check the settings every two weeks at first, and after that every two months. The current through the output valves fluctuates somewhat, which makes it difficult to use a digital voltmeter to make the adjustments. An analogue moving-coil meter is much easier to use for this purpose. Since the adjustments need to be made repeatedly, an alignment aid is a handy accessory. For this purpose, a pair of three-way female headers (one for each amplifier board) can be fitted in convenient locations using double-sided adhesive strips. The middle contact is connected to Tp0, and the outer contacts are connected to TpV3 and TpV4, respectively. The alignment aid can then be connected using a length of cable with a 3-way circuit-board header.

The current flowing through each EL34 should be 50 mA (combined anode and screen-grid currents). This yields a power dissipation of around 22 W for each valve. At this level of current, the voltage across the cathode resistor of each valve will be 0.5 V.

The circuit diagram of the alignment aid is shown next to this box. It also has to be aligned before it can be used. To do so, connect a DC voltage of 0.5 V to terminals Tp0 and TpV3 of the alignment aid and set S1 to the topmost position ('DC current'). Then adjust P1 until the meter shows a value of 50 (read mA for µA).

When S1 is in the 'DC balance' position, the circuit measures the voltage between TpV3 and TpV4. If the currents through the two valves are equal, the meter reading will be 0. The nice feature of this circuit is that it has higher sensitivity for this adjustment, since the only series resistance is provided by R1.

When S1 is in the 'AC balance' position, TpV3 and TpV4 are tied together and connected to a headphone plugged into K1. The alignment signal can be heard using the headphone.

### Adjusting the DC current and DC balance

On each amplifier board, first set P1 and P3 to their midrange positions and rotate P2 fully counter-clockwise, so that the negative grid voltage has its maximum negative value. Connect the alignment aid with its switch set to 'DC current,' and then switch on the power. Wait a few minutes, and then adjust P2 for a meter reading of 40 mA. Next, change S1 to the middle position ('DC balance') and adjust P3 to obtain a meter reading as close as possible to 0. After the amplifier has warmed up for ten minutes, you can increase the DC current to 50 mA and tweak the DC balance as necessary.

### Adjusting the AC balance

The AC balance of an amplifier is usually adjusted using a distortion meter. Mr Byrith has devised a method to allow this be done using an audible signal. Set the switch to the 'AC balance' position and connect a sine-wave signal to the input of the amplifier (1 kHz / 100 mVrms). While listening to this signal with the headphones, rotate P1 until the 1-kHz tone is as weak as possible. You will also hear mains hum and harmonics of the sine-wave signal, and the loudness of the signal will fluctuate, but it is certainly possible to find a setting where the 1-kHz tone is at a minimum. The signals on the cathodes have opposite phases, and when they are in balance they have equal amplitudes. Clever thinking!

### Square-wave alignment

Capacitor C5 in the feedback loop corrects the phase lag. If its value is a bit too small, the corners of a square-wave signal will be rounded off, and if its value is a bit too large, the corners will have overshoots. You need to have access to a square-wave generator and an oscilloscope if you want to check and/or adjust the square-wave response.

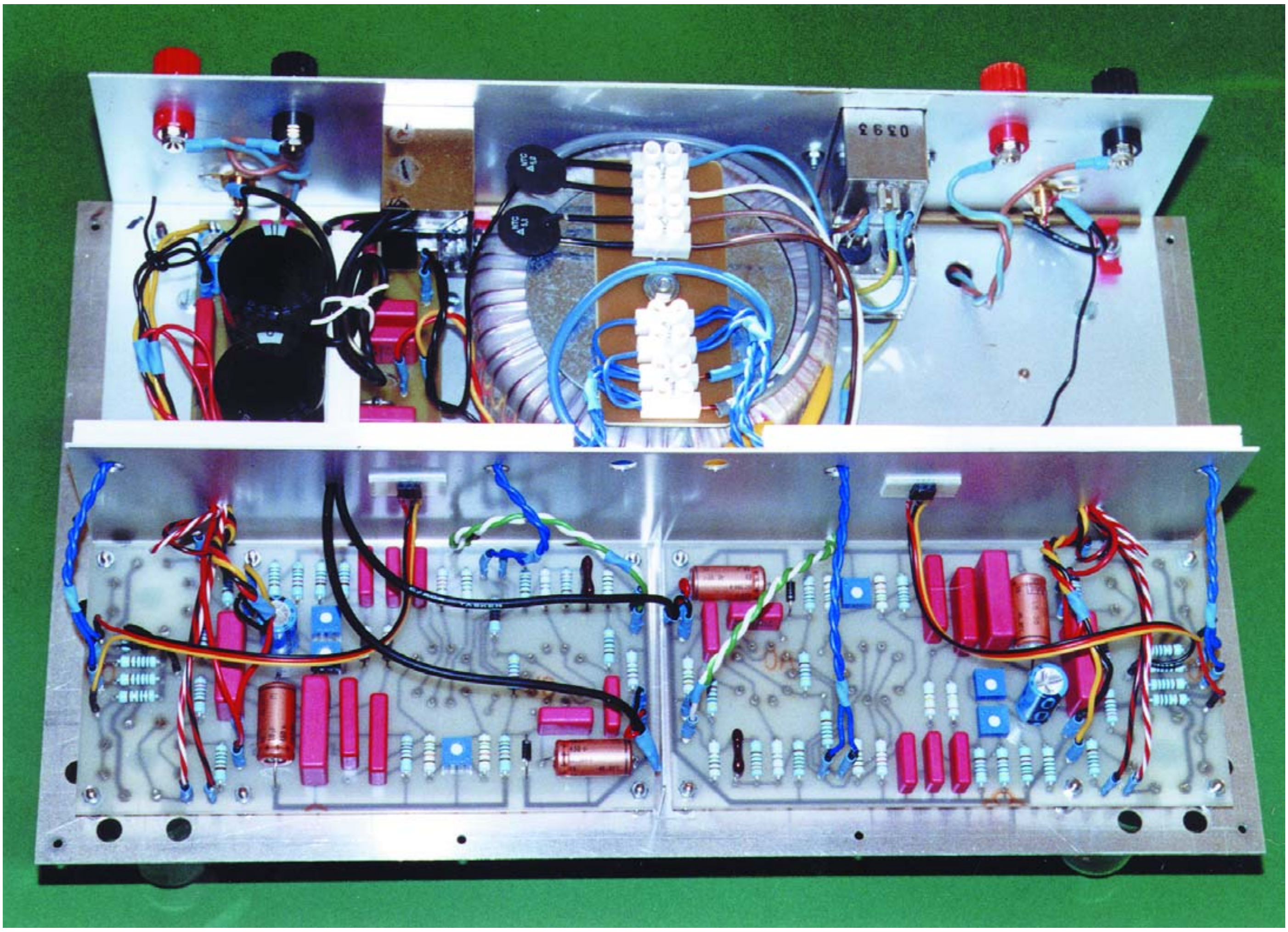


Figure 4. Bottom view of the fully assembled amplifier.

the IEC appliance socket and the Alps volume control as well.

Tape the templates to the chassis plate such that the amplifier boards are spaced 13 mm from the front and side edges (this clearance is required for the supporting strips in the case). The fixing nuts for the transformer covers must fit inside the channel section. Align the C cores with each other, and position the power transformer in the middle of the channel section.

Now you can centre-punch and drill all of the holes. For each output transformer, two holes are needed to allow the wiring to pass through the chassis. If they are drilled within the outline of the cover, they will be hidden when the assembly is finished.

Drill six holes with a diameter of 8 mm around the openings for the output valves to allow cooling air to flow past the EL34s, since they become rather hot.

Run the wiring for the filament supply in a length of small cable duct stuck to the inside of the front wall of the channel section ('pos 1' in the detail at the lower right of the wiring diagram). Make feedthrough openings at the positions of the filament connections on the circuit boards.

Run the wiring for 0 V, –55 V and +440 V in a second small cable duct located at 'pos 2'.

Attach the amplifier boards to the chassis plate using 10-mm standoffs. Adjust the separation between the boards and the plate using shim washers so that the sockets for the output valves are firmly pressed against the top plate. Fit the power supply board using standoffs as well.

Fit an aluminium screening plate between the amplifier boards, and use a sheet-metal enclosure to screen the Alps volume control.

## Testing

As long as the amplifier boards are not yet fitted, everything is easily accessible. In order to test the amplifier boards, it's convenient to first assemble the power supply portion. Fit the power transformer and the power supply circuit board to the channel section, along with the IEC appliance socket. Install a 1.5-A slow-blow fuse. In our amplifier, we fitted two four-way connector strips to a piece of epoxy board using countersunk 3-mm screws, and then secured this board to the fitting screw for the power transformer using an extra nut. The lower set of terminals (as shown in **Figure 3**) is for the filament wiring.

Practically all of the wiring, except the heavy leads for the loudspeaker terminals, consists of 0.5-mm$^2$ flexible hookup wire with various colours of insulation. Three such wires can be easily fitted into a connector strip terminal.

The four upper connector-strip terminals are used to connect the primary leads of the power transformer to the IEC appliance socket. An NTC resistor is placed in series with each lead, in order to reduce the switch-on surge. They are not absolutely necessary, but they are a simple and effective way to achieve a 'soft' switch-on.

Once the wiring interconnecting the IEC appliance socket, mains transformer and power supply board is finished, you can begin testing by checking the power supply by itself. First connect the two 230-V/15-W incandescent lamps in series between the +440 V and 0 V terminals, and then switch on the power. If the lamps light up brightly, you can then (carefully!) check the high voltage and negative grid voltage.

After switching off the power, connect the loose amplifier boards to the power supply and output transformers. Before applying the high voltage, first check that the filaments of the valves light up. With the EF86, you can see this by looking in from the top, although it is a bit tricky. Next, switch off the power and remove the output valves, and then connect the high voltage leads. Switch on the power and allow the EF86s and ECC83s to warm up, and then check the voltages on these valves. Small variations from the nominal values are possible, but a major deviation means that there is probably an incorrect resistor value somewhere.

If everything is OK, switch off the power and plug in the output valves. Now you can perform a preliminary alignment of the amplifier (see the 'Alignment' box). After this, you can fit the amplifier circuit boards in the enclosure and route the rest of the wiring.

## Finishing

Such a nice amplifier naturally deserves an attractive wooden case. We made our case from lengths of 9-mm multiplex board, after finishing off the openings in the chassis plate and channel section. There are two rectangular openings in the back of the case for the connectors and volume control. Our case is finished with veneer, but it would naturally also be possible to build a case using solid wood. Self-adhesive feet are fitted below the channel section.

Even without a case, the amplifier sits quite stably on the walls of the aluminium channel section. If strips of wood are taped onto the transformer covers, the completed amplifier, with the valves installed, can also be placed upside down on top of a table. This makes it easy to access all of the circuitry, and it is also convenient for fitting a wooden case.

The bottom of the case can be closed with an aluminium plate if desired. If you use such a plate, be sure to earth it, and drill openings for cooling airflow.

It's a good idea to switch on all the other equipment in your audio system before switching on the final amp, in order to avoid a switch-on 'thump'.

(020071-2)

Advertisement

# Countdown Timer

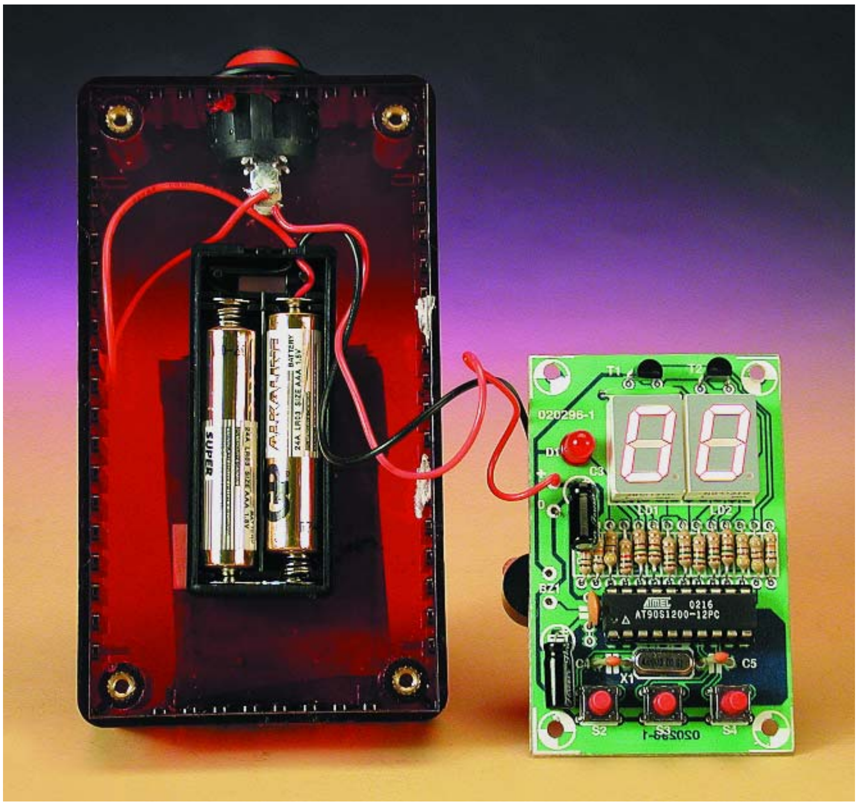
## Accurate timing from an AT90S1200 micro

Design by Andy Morell andy@morell.freeserve.co.uk

This project is simple to build and use yet accurate enough to provide a count down from any time interval anywhere from ninety-nine minutes down to one minute. The bright two digit display shows the current minutes left, and there is a piezo buzzer to indicate when the count reaches zero!

### Main Features

- Countdown period 1 minute to 99 minutes
- 2-digit 7-segment LED display
- 3-button control
- Low power consumption
- 3-volts battery supply
- AT90S1200 microcontroller chip



To use the timer is simplicity itself. First, select the time you require, by pressing the two appropriate push buttons. For example, press the unit button once for 1×1 minute, and the tens of unit button, say, three times for 3×10 minutes. There you are: a 31-minute countdown period. As you press the two switches, the number will show on the double-digit display. Both buttons will allow you to loop continuously from nought to nine, and back to nought .

Having entered the required time, you simply press the Start button. Immediately, a red LED will come on and the down count will commence. The displays start to flash when time-up is reached. A sleep function is automatically entered after 1 minute of buzzer activity. A new countdown cycle requires the microcontroller to be reset which means you have to switch the timer off and on again.

### How the hardware works

The circuit diagram of the Countdown Timer is shown in **Figure 1**. With an AT90S1200 microcontroller running at 4 MHz and using one of its built in timers, (Timer 0), it is possible to get an accurate time period of one second, and so minutes, etc. The microcontroller clock frequency is generated using a dead standard 4-MHz quartz crystal and the usual pair of small capacitors connected to the oscillator pins of the 90S1200.

The AT901200 microcontroller, is cheap, readily available, and easy to use. On top of that, it can work off a 3-volt battery supply, as we demonstrate here.

The reset pulse is generated by

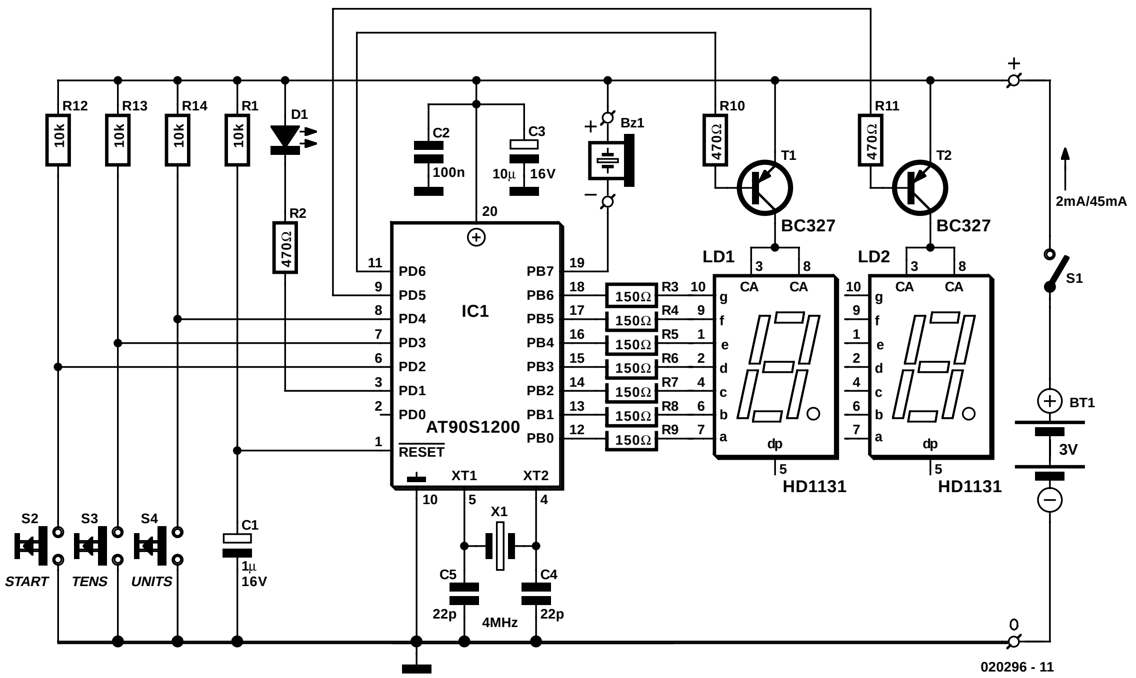



Figure 1. Circuit diagram of the Countdown Timer. A fair degree of component cutting has been achieved through the use of a microcontroller.

R1-C1, while PNP transistors T1 and T2 control the common anode terminals (CA) of the HD1131O displays. There are seven 100-Ω to 150-Ω resistors limiting the current through each of the seven segments of the active display. The actual value of the resistors will depend on the display brightness you require, set off against the current consumption (i.e., battery life), so a compromise will need to be found. Initially, we recommend using 150 Ω resistors.

There are also three 10-kΩ pull-up resistors for the three pushbutton switches. The logic High (+3 V) level created by these resistors helps prevent spurious triggering of the microcontroller input pins PD2, PD3 and PD4.

Finally, a small piezo buzzer of the 'active' (DC) variety is used. This device is switched on directly by the PB7 microcontroller pin and is sure to give an adequate sound level when it is required to.

## How the software works

A flowchart of the main program running inside the AT90S1200 micro is shown in **Figure 2**. The structure of the interrupt routine is illustrated separately in **Figure 3**.

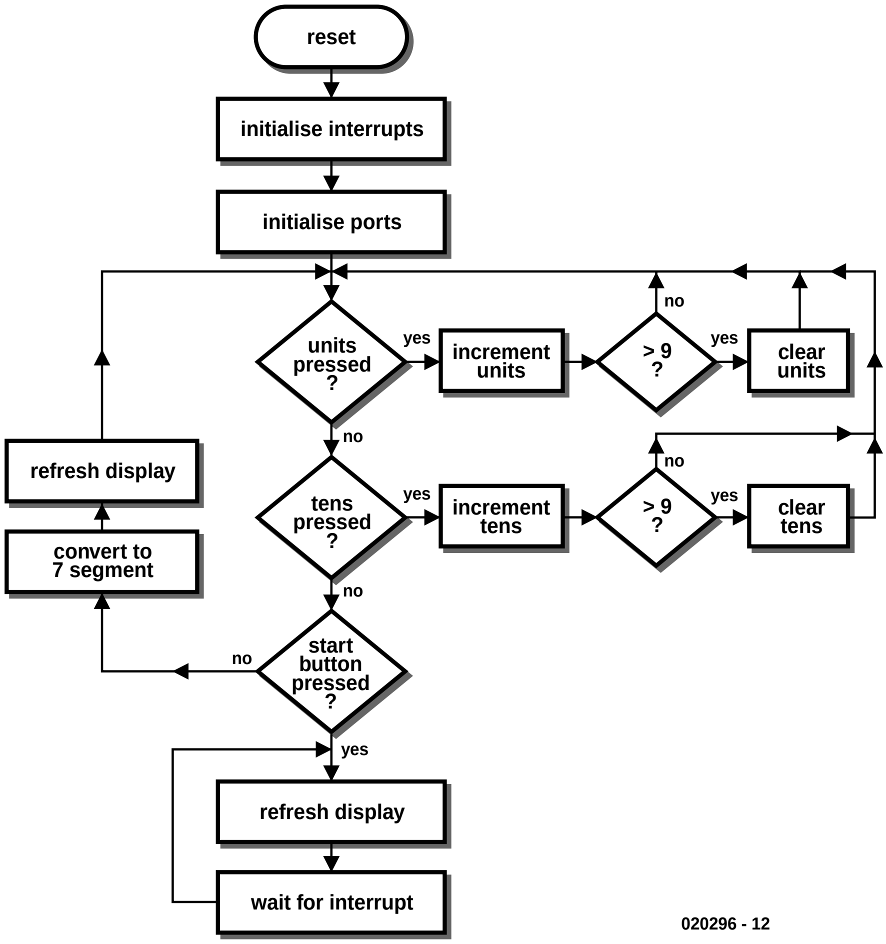



Figure 2. Main program flowchart.

To be able to understand the discussion below, it is useful to have the source code listing of the program available on paper or on your computer screen. The relevant file is available on floppy disk as order code **020296-11**, or as a Free Download from our website under the same number.

Upon Reset, the microcontroller sets up the stack pointer to enable it to store addresses etc., when calling subroutines, especially in the case of interrupts, which are used in this design.

As stated earlier, we use the built in timer Timer 0. This calls for the Timer 0 Control Register, TCCR0, to be set up. Division of the clock frequency by 256 allows us to generate an interrupt every 1 second. This is done by pre-loading the TCCR0 register with 0b00000100, or 04 in (hexa)decimal.

Setting up the Timer 0 interrupt to occur is the next task. This is done by setting bit 1 (T0IE0) in the TIMSK register. Now the Timer 0 interrupt is set up and will occur every 1 second using the 4-MHz crystal.

On interrupt the routine display DISPLAY_DIGITS is called. It is here that 61 interrupts are counted in a register called time_out.

Here is the relevant part of that routine:

```
DISPLAY_DIGITS:

IN      SAVE_STATUS, SREG
CPI     FLAG_1,$FF
BRNE ALARM_NOT_SET

INC     TIME_OUT
CPI     TIME_OUT, 61 ;1 SECOND
BREQ TIMER_OUT
```

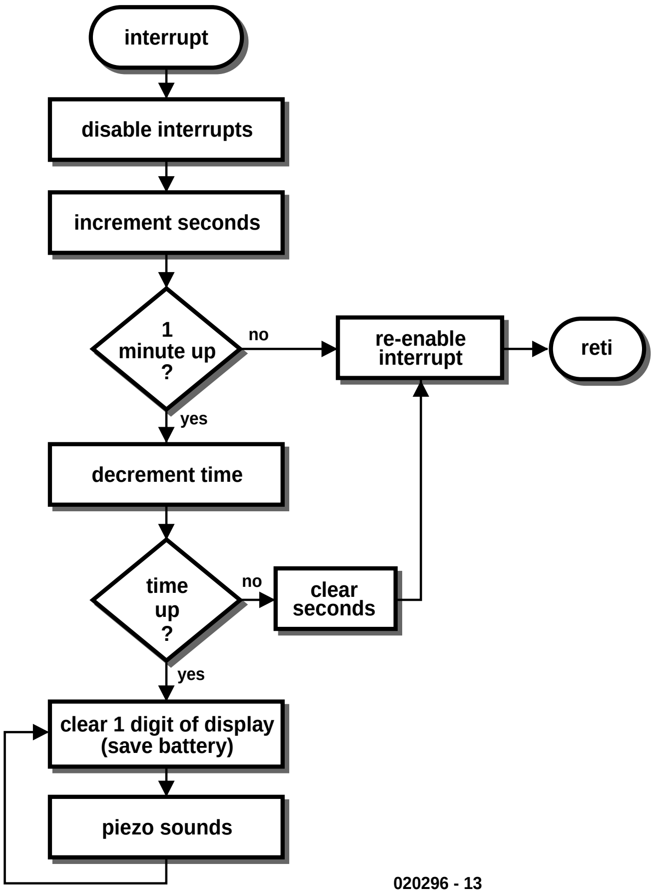
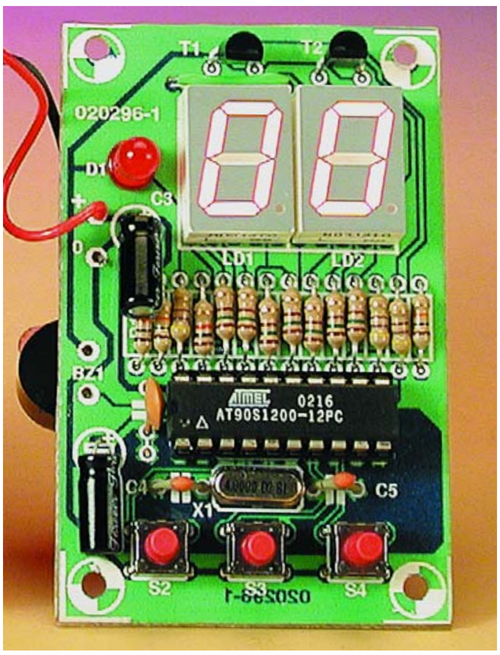
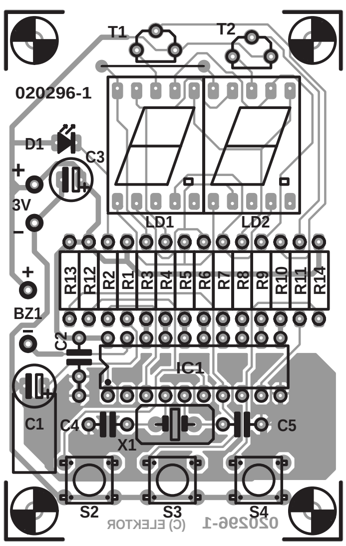



Figure 3. Interrupt routine flowchart.

We don't actually want the interrupt to occur just yet, and so we will now Inhibit it with the instruction 'CLI' (Clear Interrupt). Now the various ports are set up to their respective input and output state.

You can easily see how the software for the pushbuttons works. If you press the 'units' button then the routine for incrementing the units is





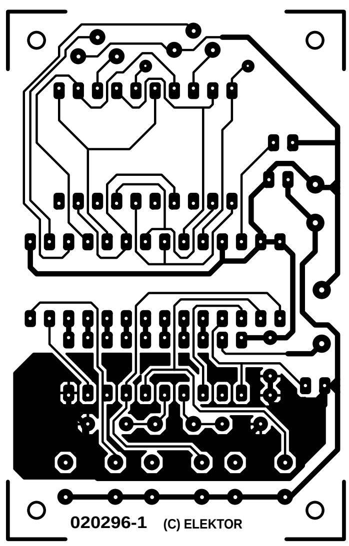




Figure 4. PCB design for the Countdown Timer (board available from The PCBShop).

## COMPONENTS LIST

**Resistors:**
R1,R12,R13,R14 = 10kΩ
R2,R10,R11 = 470Ω
R3-R9 = 100Ω to 150Ω (see text)

**Capacitors:**
C1 = 1µF 16V radial
C2 = 100nF
C3 = 10µF 16V radial
C4,C5 = 22pF

**Semiconductors:**
D1 = LED, red, 3mm dia.
IC1 = AT90S1200-12PC, order code **020296-41**
LD1,LD2 = HD1131O (see text)
T1, T2 = BC327

**Miscellaneous:**
BT1 = two 1.5V AA or AAA batteries with holder
Bz1 = 5V DC buzzer
S1 = on/off switch
S2,S3,S4 = pushbutton with 1 make contact (see text)
X1 = 4MHz quartz crystal (low-profile model)

called. The same applies to the 'tens' button. It is only when you press the 'Start' button that the program branches to the count down section (eventually).

Before pressing the 'Start' button, the program is in a loop, and refresh of the display is achieved within this loop. There is also a 'Get Seven Segment' routine, which puts the appropriate pattern on the displays.

The program is then waiting for you to take your finger off the 'Start' button, so it may proceed to the next bit.

At this point, the interrupt is enabled, and the Interrupt Service Routine is then made accessible every one second. Now the program is in a continual loop, with refresh display, and of course the conversion to 7-segment contained here.

Interrupts will start occurring now, and the Interrupt Service Routine is called (Figure 3).

Not only is this where the down count is done, but it also checks if the count has reached zero, or not (this is done by checking if COUNT1 and COUNT2 are both 0. If this is not the case, then the microcontroller continues to display the current time and also calls a Convert To BCD Routine. This is to stop the display showing Alpha-numeric characters, i.e., characters other than 0-9.

Note that the interrupt is disabled at the start of the ISR and re-enabled at the close of it. Eventually the ISR is left, and the program returns to the main loop described earlier. This will continue until COUNT1 and COUNT2 are both at 00.

There's more to discover and enjoy in the source code listings than can be described here, and the free source code file is recommended not only to those of you who can program their own chips but also to AVR programmers looking for new ideas.

## How to build the timer

The copper track layout and component mounting plan of the small single-sided PCB designed for the Countdown Timer are shown in **Figure 4**. This PCB is not available ready-made through our Readers Services but it may still be ordered from The PCBShop operated by our business partner EuroCircuits.

Start the construction with fitting the single wire link above display LD1. Next, fit the resistors, and then the capacitors (watch the polarity of C1 and C3). After that, insert a 20-pin DIL IC socket, followed by the crystal. The piezo buzzer and the battery are external to the board and connected up by light-duty flexible wire.

The HD1131 common-anode display comes in several versions, the suffix indicating the colour. Here the 'O' super-red device is recommended. The luminosity may be indicated separately in the device type code, for example, by an 'L'. By choosing a really bright version, a good trade-off can be achieved between current consumption and readability. In this context, see also the notes on the values of R3-R9.

Depending on the enclosure you wish to use, the LED and the three pushbuttons may have to be mounted a little off the board surface to make sure they are flush with the enclosure top panel. In the case of the switches, this may require lengthening their pins with pieces of stiff wire.

Then solder in the two BC327 transistors and the displays. As with the LED and the switches, you may wish to raise the displays from the board surface by inserting them in SIL socket strips. There are some vero-pins for the battery, the buzzer and the on/off switch, which are panel mounted. Note that the batteries are 2 x 'AAA' cells, which are perfectly adequate for many hours of practical use.

(020296-1)

## Free Downloads

– source code and HEX files. File number: **020296-11.zip**
– PCB layout in PDF format. File number: **020296-1.zip**

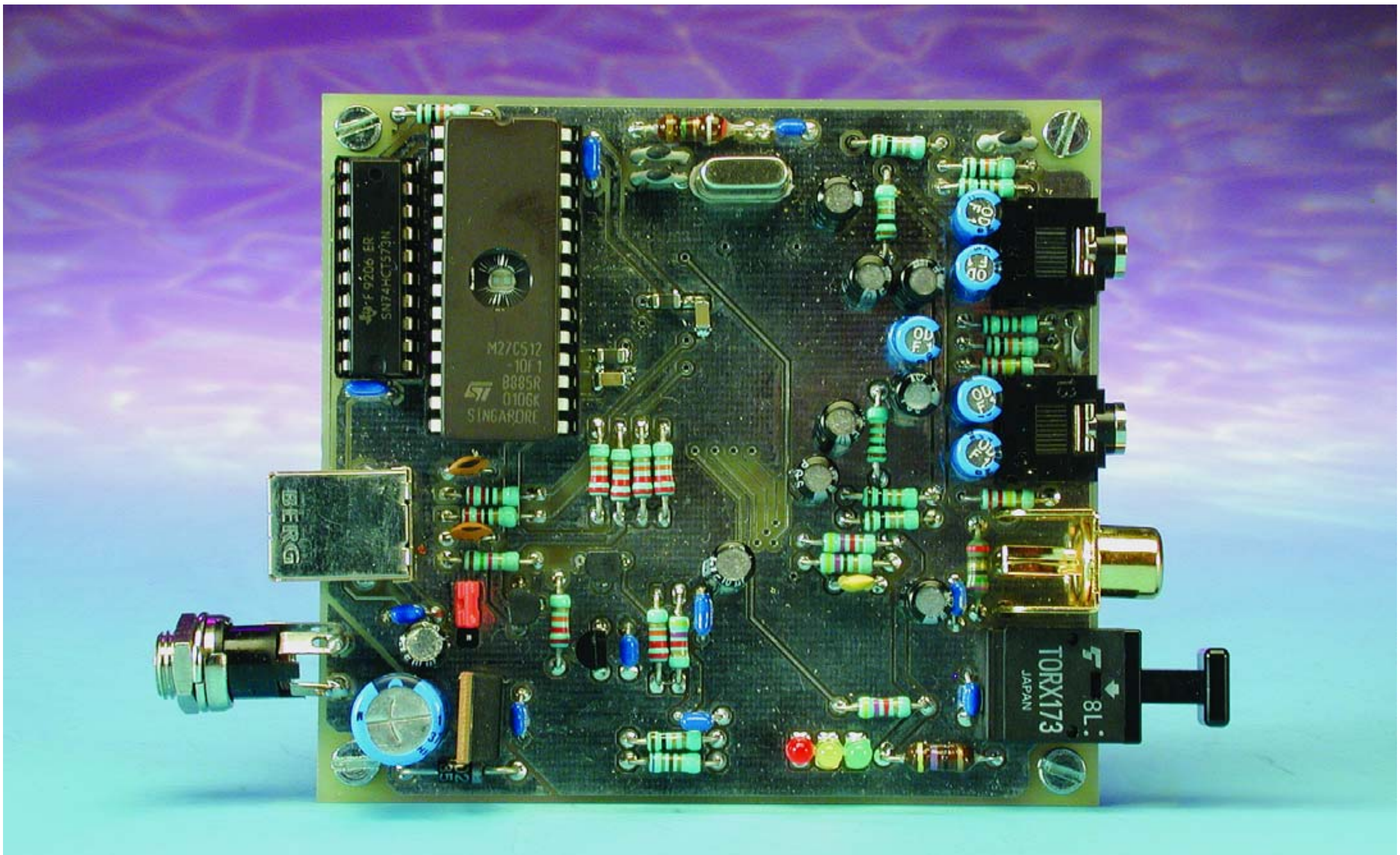
www.elektor-electronics.co.uk/dl/dl.htm, select month of publication.

# USB Audio Recorder

## A fully-featured sound card for the Universal Serial Bus

Design by G. Kronauer and T. Zepf

**Many motherboards these days only offer limited space for expansion cards, and integrated sound chips are generally not so well suited to high-quality recording. If a sound card is fitted in the PC, it is exposed to enormous levels of interference which does nothing for the audio quality. The solution is an external sound card, connected over USB.**




Installing an add-in sound card into a system is generally a lot of fuss. In order to transfer audio data digitally to the PC, either you will need an S/PDIF interface on the sound card or else you will be reduced to 'grabbing' the audio using the CD-ROM drive. If you want to make recordings using a notebook computer, you will have to be content with the microphone input or purchase an expensive PCMCIA sound card.

All more recent PCs offer a powerful interface in the form of the Universal Serial Bus, which can also be used for transferring audio data. The circuit described here is a fully-featured sound card, capable of full-duplex operation, that runs under Windows 98SE, MacOS or Linux. The UDA1335 and UDA1325 are ICs from Philips that include a USB interface, A/D and D/A converters, $I^2C$ interface and an 8052-compatible microcontroller. The chip is available in various versions: the UDA1335 (without ROM), the UDA1325 (with N106 firmware) and the UDA1325 (with N104 firmware). In this application, the microcontroller is always used in the ROM-less (EA=0) mode, and so it does not matter which firmware version is present.

The ROM-less mode of operation has been chose here because the firmware in the ROM version has been found to be so full of bugs that it is not suitable for serious use. The author has had the chip under the microscope since shortly after it came onto the market and, after much detailed work, has analysed the multitude of bugs and developed workarounds for them. The trouble has been worthwhile: this USB sound card is by a clear margin more reliable than generally-available USB sound cards using the UDA1325.

The TDA1315H is used to convert the S/PDIF data into $I^2S$ format: this device offers two S/PDIF inputs. This allows the circuit to switch between S/PDIF phono (or Cinch) and TOSLink (optical) connections directly in the UDA1325 without requiring extra hardware. This means we can connect to older CD players, that only have an phono output, as well as to more recent models that have only an optical output. An EPROM, with the obligatory address latch, and a 3.3 V voltage regulator to supply the UDA1325 complete the design.

## The circuit

The S/PDIF receiver is operated in ‘hardware mode’. It receives the data directly from K1 (phono) or IC1 (TOSLink), connected straight to pins IECIN0 and IECIN1 respectively, as shown in **Figure 1**. If a signal is present at either of these two inputs, the PLL in the receiver will attempt to lock on to it. The level on pin 7 of IC2 (IECSEL) determines which of the two inputs is used. When the PLL is in lock, the sample rate of the received data is indicated by the three LEDs D1 to D3. The $I^2S$ output then carries the digital audio signal which is taken to the UDA1325 on pins DA, DS and BCK. The device also has an analogue input on pins VINR and VINL, which is connected via coupling capacitors C28 and C29. Input selection is controlled in software under Windows: more about this later.

The digital audio data for recording are carried to the PC via a USB ‘In Pipe’ and playback data are carried back from the PC to the sound card over a USB ‘Out Pipe’. The volume can be adjusted using the UDA1325’s built-in DSP. After D/A conversion, the signal is made available at K4 via coupling capacitors C37 and C38 at a level of 0 dBu. This can be connected to a line input or to high-impedance headphones such as those supplied with portable CD players.

The address and data bus of the 80C52 CPU integrated into the UDA1325 is, as usual with such CPUs, connected to EPROM IC5 via latch IC4. PSEN is used as the read strobe signal for the EPROM. ALE selects between data and address information on P0. In order to allow either type 27C256 or type 27C512 EPROMs to be used, A15 is connected permanently to the positive supply: note therefore, when using a 27C512 EPROM, that the code must be stored from location 0x8000 onwards. Since RD and WR are not externally accessible it is not possible to connect other peripherals to the CPU bus. Internally to the chip the RD and WR signals are used in addressing hardware registers. There is little point in going into great detail here about the internals of the chip. Copious information is provided in the data sheets available on the Internet at www.semiconductors.philips.com/acrobat/datasheets/UDA1325_N_1.pdf and /TDA1315H_5.pdf.

A 48 MHz crystal is used as the clock source, which means that the CPU is clocked at 24 MHz. C32 and L1 ensure that the crystal oscillates at 48 MHz and not at 16 MHz. The capacitor at pin 40 (VREFDA) of the UDA1325 looks after not only the provision of a suitable reset pulse for the microcontroller but also the reference voltage for the D/A converter. T2 is used to switch R10 into the D+ circuit of the USB. The host recognises this as a ‘USB connect’ and will respond by beginning the process of enumeration.

The circuit can be powered from the USB itself or powered from its own power supply. Since the configuration descriptor data need to differ between the bus-powered and self-powered cases, JP1 is used to switch address pin A14 (but not while the circuit is powered!).

## Features

- compatible with USB specification 1.0 and above
- compatible with USB Audio specification 1.0
- analogue input (adjustable from –3 dB to +27 dB)
- coaxial S/PDIF phono input
- optical digital input (TOSLink)
- analogue output (adjustable over a 60 dB range)
- analogue sample rates from 8 kHz to 48 kHz
- digital sample rates of 32 kHz, 44.1 kHz and 48 kHz
- full-duplex operation supported
- Plug & Play compatible

## Firmware

The software provided by Philips does not support either switching of the input source or increasing the sensitivity level of the input in analogue operation. This is a further reason to develop our own firmware. Because of a fault in the Windows 98 audio driver, switching between audio inputs only works from Windows 98SE onwards. It is a simple matter to check which version of Windows you have under System Properties. Earlier versions of Windows 98 only allow you to switch off the analogue signal. In this case you will need to update to Windows 98SE, and suitable (even legal!) software is readily available on the Internet for a small fee.

The four main jobs of the firmware are as follows:

- Controlling the USB core;
- Processing USB 1.0 standard requests (for example GetDescriptor);
- Processing USB Audio Class 1.0 requests

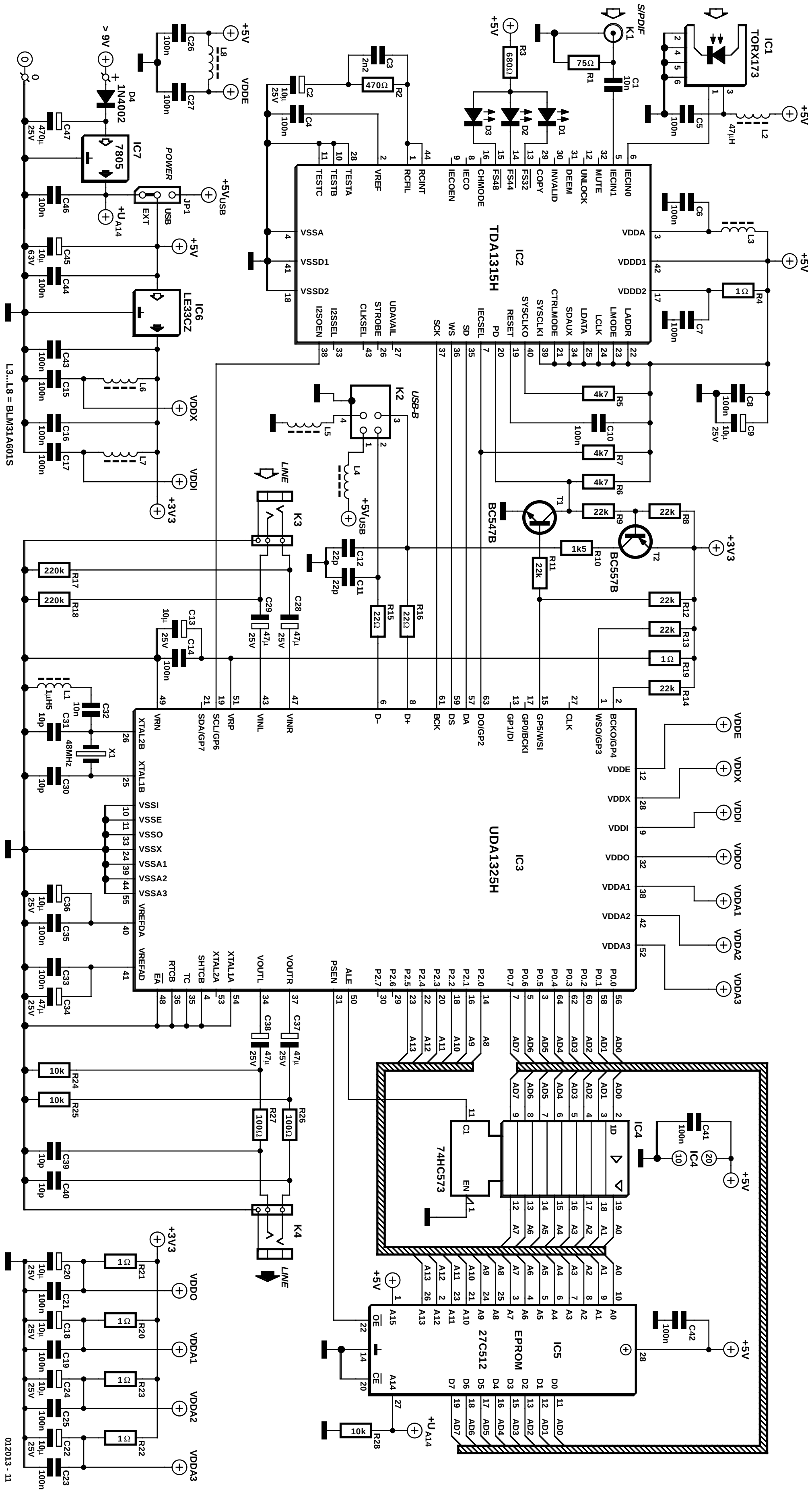



Figure 1. Circuit diagram showing TDA1315 S/PDIF receiver, and the UDA1325, which includes the USB interface, converters and a DSP.

(for example volume control);
– Driving the on-chip hardware, such as the A/D converter, and setting the sample rate.

The digital audio data are autonomously transferred by DMA hardware and so this does not require any software intervention.

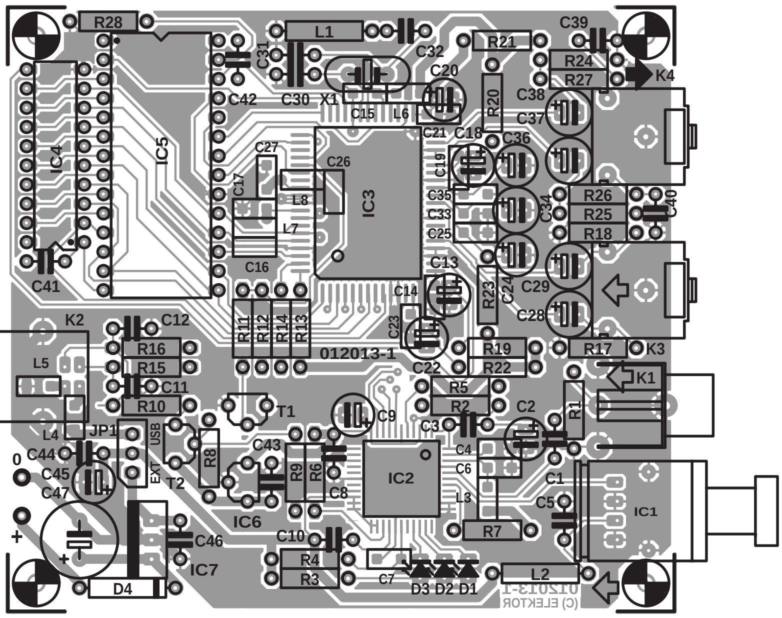
USB audio devices are so-called 'compound devices', which means that they are composed of several interfaces. In this case we have three interfaces: control, wave out, and wave in. Under Windows, each device requires a separate driver to be installed. In this way it is very simple to add extra devices (for example human interface devices) to the list of descriptors.

In order for a USB device to be recognised by the operating system as an audio device, a list of descriptors for audio class (AC) specific interfaces and endpoints must be given in the configuration descriptor. Depending on the scale of the topology and the number of alternate interface settings this can easily add up to a kilobyte of data. The characteristics of the sound card (inputs, outputs, connections, controller) are specified in the AC control descriptor. The Microsoft driver development kit includes a tool called grapher.exe, which allows simple display of the USB topology. Each audio pipe must offer an 'Alternate Setting 0' which releases all possible bandwidth on the USB when the pipe is closed.

**The configuration descriptor for a USB audio device is formed as follows:**

Config

| | |
|---|---|
| Interface0 | pipe (control) |
| AC Control | audio (Topologie) |
| .... | |
| Interface (alt0) | pipe |
| Interface (alt1) | pipe (waveout) |
| ACInterface | audio (stream format) |
| Endpoint | ISO |
| ACEndpoint | audio (sample rates) |
| .... | |
| Interface (alt0) | pipe |
| Interface (alt1) | pipe (wave in) |
| ACInterface | audio (stream format) |
| Endpoint | ISO |
| ACEndpoint | audio (sample rates) |



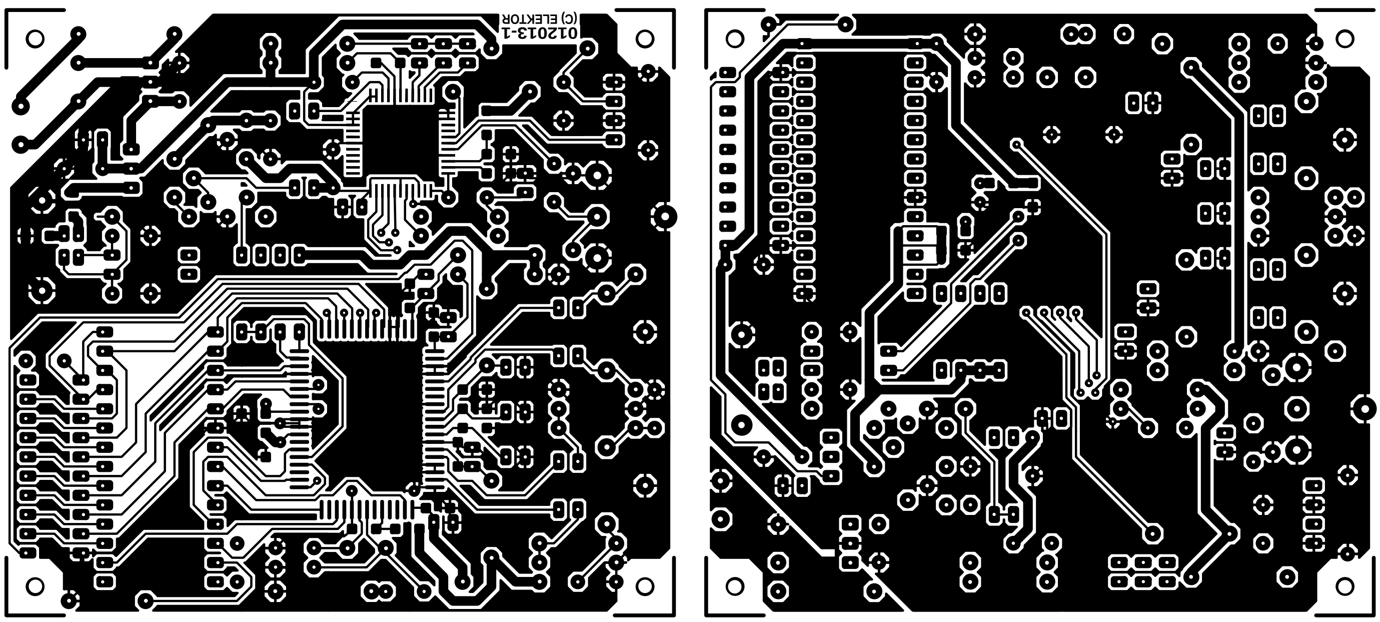


Figure 2. Printed circuit board layout and component mounting plan.

## Construction

A glance at the double-sided layout for the printed circuit board in **Figure 2** will reveal that IC2 and IC3, as well as a few passive components, are SMDs. The two ICs have rather a large number of pins, and so construction cannot exactly be described as a straightforward task. If you have not had previous experience with SMDs, you should get expert advice: it is all to easy to destroy

## COMPONENTS LIST

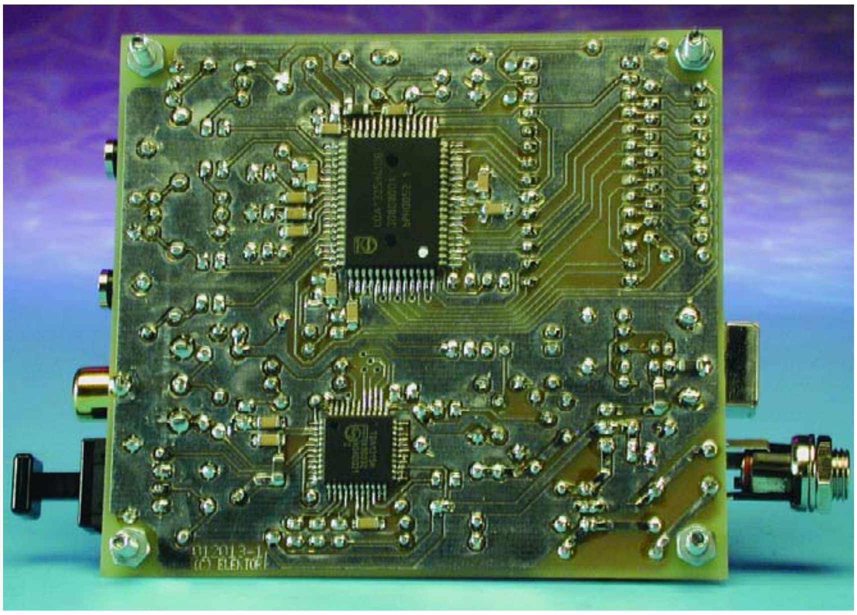
**Resistors:**
R1 = 75Ω
R2 = 470Ω
R3 = 680Ω
R4,R19-R23 = 1Ω
R5,R6,R7 = 4kΩ7
R8,R9,R11-R14 = 22kΩ
R10 = 1kΩ5
R15,R16 = 22Ω
R17,R18 = 220kΩ
R24,R25,R28 = 10kΩ
R26,R27 = 100Ω

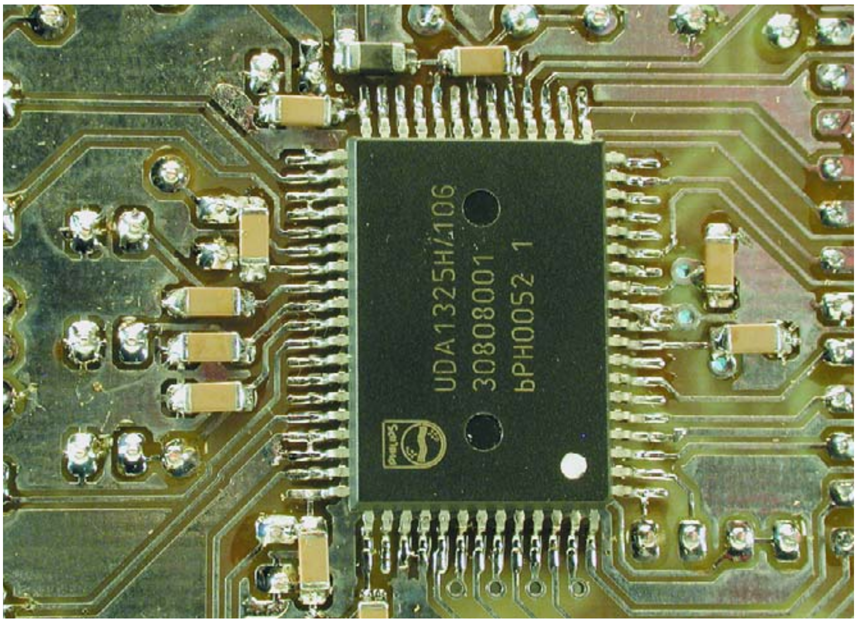
**Capacitors:**
C1,C32 = 10nF ceramic, lead pitch 5mm
C2,C9,C13,C18,C20,C22,C24, C36 = 10μF 25V radial, lead pitch 5mm
C3 = 2nF2 ceramic, lead pitch 5mm
C4,C6,C7,C14-C17,C19,C21,C23,C25, C26,C27, C33,C35 = 100nF (SMD shape 1206)
C5,C8,C10,C41-C44,C46 = 100nF ceramic, lead pitch 5mm
C11,C12 = 22pF
C28,C29,C34,C37,C38 = 47μF 25V radial
C30,C31,C39,C40 = 10pF
C45 = 10μF 63V radial
C47 = 470μF 25V radial

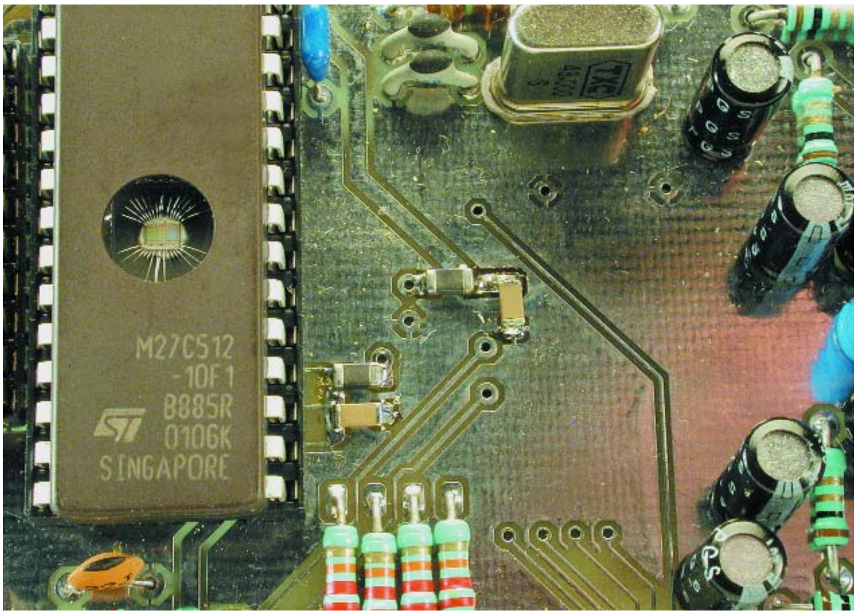
**Inductors:**
L1 = 1μH5 miniature choke
L2 = 47μH miniature choke
L3-L8 = BLM31A601S (Murata, SMD shape 1206, Farnell # 581-094)

**Semiconductors:**
D1,D2,D3 = LED, 3mm dia.
D4 = 1N4002
T1 = BC547B
T2 = BC557B
IC1 = TORX173
IC2 = TDA1315H/N2 (Philips)
IC3 = UDA1325H/N106 (Philips)
IC4 = 74HC573
IC5 = 27C512-10, order code **012013-21**
IC6 = LE33CZ (ST, Farnell) or LP2950CZ-3.3 (National Semiconductor)
IC7 = 7805

**Miscellaneous:**
JP1 = 3-way pinheader with jumper
K1 = Cinch socket, PCB mount (e.g., T-709G from Monacor/Monarch)
K2 = USB-B socket, PCB mount
K3,K4 = 3.5-mm stereo jack socket, PCN mount (Conrad Electronics # 732893)
X1 = 48MHz quartz crystal, 3rd overtone, parallel resonance
PCB, order code **012013-1**
Disk, order code **012013-11**





expensive components, especially when using an ordinary soldering iron. A specially-designed SMD soldering station, as we have recently discussed in *Elektor Electronics* ('SMDs? Don't panic!' in the January 2003 issue) is extremely useful here. Purchasing such apparatus is however rather expensive and only recommended to those who plan to make extensive use of SMDs in their hobby.

As usual, one the two ICs have been neatly soldered and the 21 SMD capacitors and coils fitted, turn the circuit board over and start on the leaded components. After sweating over the SMDs this should present no difficulty, as long as care is taken with the polarity of the electrolytic capacitors. The DIL ICs (the EPROM and the latch) should be fitted in sockets, and the LEDs should only be fitted once you have decided how they should appear in the enclosure. Finally fit the large connectors: two mini-jack sockets, the TOSLink, the phono and the USB sockets. Once the printed circuit board looks like our photos, give it a thorough visual inspection for solder bridges and other problems: and then the job is done.

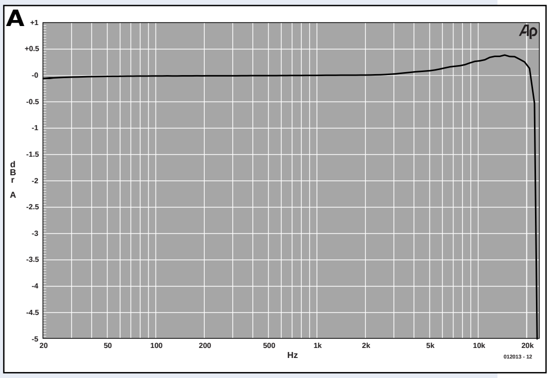
## Operation

Fit jumper JP1 in position 'EXT' and apply power from an external mains power supply. The current consumption should be around 110 mA. In contrast to the USB Audio Codec published in the December 2002 issue of *Elektor Electronics* this circuit can also be supplied from a USB hub, if JP1 is fitted in the appropriate position. A signal at 4 MHz should be found at pin 11 of IC4; if not, check the oscillator circuit around X1 for soldering or other assembly errors.

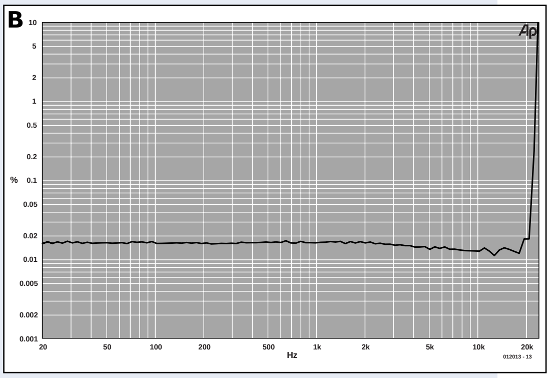
Users of Windows 98SE or Windows ME should install the relevant patches from Microsoft before proceeding. The reasons for doing this and how to go about it are explained in the text box. Installing the audio recorder is considerably simplified by Plug and Play. Once the device is connected, Windows recognises it automatically and installs the USB audio recorder drivers: SNDREC10.INF for the self-powered

## Performance curves

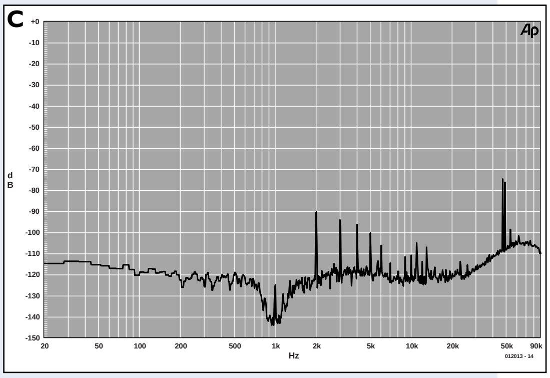
The first performance curve, **curve A**, shows the amplitude response of the system when the A/D and D/A converters are connected in series. To carry out this measurement a special ASIO multimedia driver for Cubase VST/32 was used (see www.usb-audio.com), which allows full-duplex operation, to briefly delay the output from a signal generator. The peak at the end of the curve is probably caused by the digital output filter: the data sheet is silent on this point. If the D/A converter is measured on its own, the peak is at most 0.5 dB high.




**Curve B** shows distortion plus noise as a function of frequency. Again the converters are connected in series, and the distortion is chiefly caused by the A/D converter. The values (digitally measured) are about a factor of two higher than the maximum figures given in the data sheet.




**Curve C** shows the frequency response of the D/A converter when driven at full amplitude (0 dB), measured with a 997 Hz signal sampled at 48 kHz. The output signal level is approximately 0.66 V. Over a bandwidth of 22 kHz the total distortion plus noise is approximately 0.008 %. There are no harmonics within the audio band above –90 dB. Despite the rise in the noise level above 20 kHz (typical of delta-sigma converters with noise shaping) the distortion figure is still only 0.03 % when the measurement bandwidth is increased to 80 kHz.




**Measured values** (bus-powered unless otherwise noted)

| | |
|---|---|
| Current consumption (bus-powered) | 96 mA |
| Current consumption (self-powered) | 109 mA |
| **DAC** | |
| Nominal output amplitude (0 dB) | 0.66 V |
| Bandwidth (-3 dB) | 23.9 kHz ($f_s$ = 48 kHz) |
| Amplitude (20 kHz) | +0.38 dB ($f_s$ = 48 kHz) |
| Output impedance | 110 Ω |
| Signal-to-noise ratio | >96 dBA |
| THD+N (1 kHz) | 0.008 % (B = 22 kHz) |
| | 0.03 % (B = 80 kHz) |
| Channel separation | >100 dB (1 kHz) |
| | >87 dB (20 kHz) |
| **ADC** | |
| Maximum input sensitivity | 58 mV |
| Minimum input sensitivity | 1.42 V |
| Input impedance | 12 kΩ |
| THD+N (1 kHz, –0.5 dBFs) | 0.02 % (B = 24 kHz) |
| Channel separation | >69 dB (1 kHz) |
| | >47 dB (20 kHz) |

## Free Downloads

– PCB layout. File number: **012013-1**.
– EPROM-Hex code (no source!).
File number: **012013-11**.
www.elektor-electronics.co.uk/dl/dl.htm,
select month of publication.

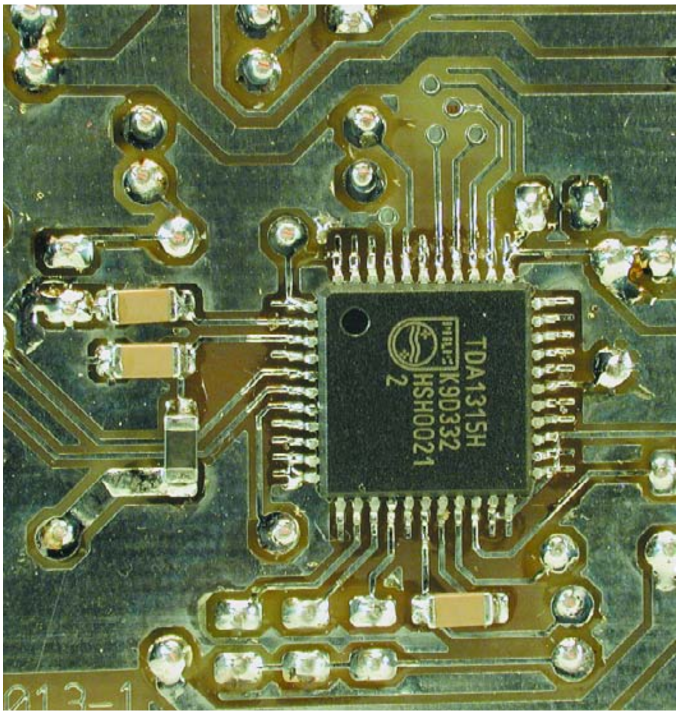
## Windows bug fixes

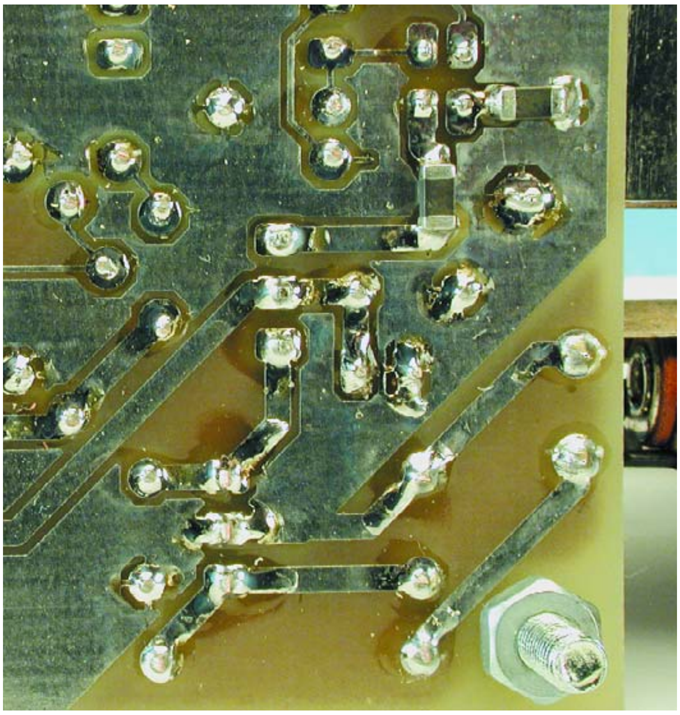
USB audio playback and recording devices are not correctly dealt with in the driver (SNDVOL32.EXE), since it cannot distinguish between two devices with the same name. This can have various consequences. Once the USB device is installed and the system rebooted, the CD volume control no longer works. The volume slider is not displayed with the correct name, there is more than one input selector, both are labelled 'USB Audio Device', and they cannot be individually selected. Hardware such as this sound card, which supports full-duplex recording and playback, is also not correctly displayed. The Line In and Mic options are displayed as a USB Audio Device, but neither of the input selectors can be chosen. The master playback volume control is displayed as a USB Audio Device.
Bug fixes for these problems are available as files Q269601.EXE and Q280127.EXE at the sites
www.media-assistance.com/English/index_english.html
and
www.pcsound.philips.com/driverfiles/patch269601usa8/269601usa8.exe (only for Windows 98SE).
On Microsoft's website at http://support.microsoft.com/?scid=KB;en-us:280127 you can also find the patch for Windows ME.
If these bug fixes are applied after the USB Audio Recorder has been installed, or if you experience other problems with the display of audio devices, it can help to remove the USB Audio Device from the registry using RegEdit. This requires a little experience, since an erroneous addition to or deletion from the registry can have fatal consequences for the system.
First search for the key HKEY_LOCAL_MACHINE\Enum\USB. If multiple USB devices have been or are connected, you will find them categorised according to their Vendor ID and Product ID. The entry for the USB Audio Recorder starts 'VID_0C7D...'. Now disconnect the USB cable and delete all the keys relating to the device from the registry. Close RegEdit and reconnect the USB Audio Recorder. All the drivers will be installed afresh. In contrast, installation under Windows XP, MacOS and, of course, Linux, should present no problems.

mode and SNDREC11.INF for the bus-powered mode (these files are stored in the INF directory). The operating system CD is required for this operation. After installation of these files and restarting the machine, the audio device will be recognised and the hardware manager should be opened again. This time the USB audio drivers are installed. If the patches have been installed, the PnP manager will point out that there are newer versions of WDM Audio in the driver list. These newer versions should not be overwritten. After a successful installation the USB audio device will be found under 'Audio/Video/ Game controller' in the Control Panel. Now any sound software that supports the Windows Multimedia Driver can record and play back audio data.

An advantage of the USB audio recorder over the USB Audio Codec is that the input selection is under menu control. The recorder Wave In menu allows the user to select between three inputs: S/PDIF digital audio over the coaxial input, S/PDIF over the optical input, and the analogue line input. If a digital input is selected the LEDs indicate the sample rate, and the input level cannot of course be adjusted. The level control for the line input covers the gain range of the internal preamplifier from –3 dB to +27 dB; on the USB Audio Codec the choice was fixed in hardware between two levels.

(012013-1)



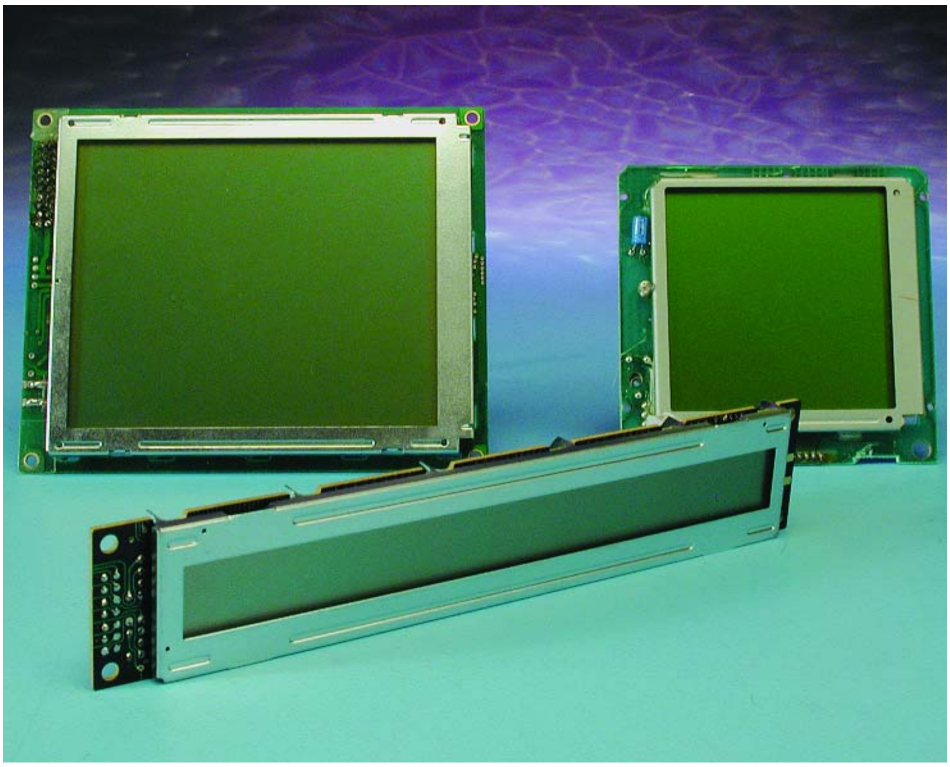



# Low-Cost LCD Controller (1)

## Part 1: Background and operation

By Wim Huiskamp huiskamp@fel.tno.nl

Surplus stores often sell cheap matrix LCD modules, some brand new (NOS), others salvaged from broken or old equipment. Most of these displays have segment and common drivers on board but lack a controller and memory. This is different from the well-known LCD character modules with built-in HD44780 controllers and memory. Part 2 of this article will show the design and implementation of a low-cost 80C51-based controller for bare LCDs. First, however, this month's instalment provides some background information on the operation of matrix LCDs.



LCDs from surplus stores are cheap but problematic when it comes to actually connecting them. This is in sharp contrast (pun intended) with the displays usually applied in Elektor circuits because the latter are usually based on the industry standard HD44780 controller chip. Obviously, matrix LCD controllers (e.g., the Seiko SED1335) are available that will generate the required drive signals and also interface to the memory of large, not so standard displays that may be picked up at bargain prices. The downside of these devices, from the perspective of the electronics hobbyist, is that they are SMDs (surface mount devices) and either hard to obtain or very expensive as one-offs.

Next month's instalment will discuss a DIY multi-purpose controller for an LCD matrix display with up to 200 × 200 pixels.

## Matrix LCD Module Operation

Liquid Crystal Displays or LCDs are different from Cathode Ray Tubes (CRTs) in that they do not emit light. LCDs either block or pass the ambient light or light produced by an (integrated) active source like LEDs or a fluorescent lamp. The liquid crystal material is locked in between two glass plates that are placed a few micrometres apart. The crystals change their optical characteristics (i.e., modify the polarisation direction of the light travelling through them) when exposed to an electric field. The electric field is provided by very thin (i.e., virtually transparent) electrodes deposited onto each of the two glass plates. Two types of LCDs are used: Reflective and Transmissive. The reflective type has a polarisation filter in front of the display that allows ambient light polarised to travel through the glass and the LCD material, but in one direction only. The polarised light is reflected by a mirror at the back of the display and passes the LCD material once again. Depending on the additional change in polarisation by the LCD material, the front polarisation filter will now either block the light (producing a black image) or allow it to pass through (for a grey image). The transmissive LCDs require a light source on one side of the display and this light is either blocked or passed depending on the polarisation of the LCD material.

Almost any symbol can be shown on LCDs by shaping the outline of the electrodes. For example, individual segments of a '7-segment' digit or battery full/empty symbols.

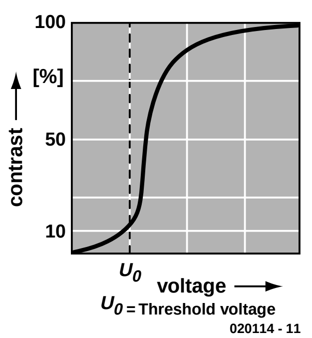



Figure 1. Typical LCD contrast control graph.

## Control

LCDs with small numbers of segments (e.g., four digits of a digital clock) may have a separate electrode for each segment — these displays are driven in the so-called 'static' mode. This driver method is not suitable for a display with lots of segments because the number of connections to the glass becomes impractical. The so-called 'multiplex' driver mode solves the connection problem and reduces the cost of the driver hardware. Obviously, graphics LCDs are driven in multiplex mode. Graphics or matrix LCD displays are organised as a two-dimensional grid of picture elements (pixels). Each pixel can be either 'on' or 'off'. One of the glass plates has its electrodes arranged in horizontal rows ('common' electrodes); the other glass plate has its electrodes arranged in vertical columns ('segment' electrodes). Each crossing point of common electrodes and segment electrodes forms a display pixel. An electrical field is created across a pixel location by creating a voltage difference between one row and one column electrode. The pixel decays within a few milliseconds after the field has been removed. LCDs therefore need to constantly refresh image display data (just like CRTs do) and when the display refresh rate is too low, the image starts to flicker.

While the intensity (i.e., perceived brightness) of the CRT image depends on the current of the electron beam hitting the screen phosphor, LCD pixels are turned on rather abruptly when the electric field between the electrodes of the pixel is above a certain threshold (see **Figure 1**). The voltage between the electrodes does have some influence on the contrast between pixels that are 'on' or 'off' but intensity variation (i.e., level of pixel 'blackness') needs to be achieved in another way. The most common method to create multiple grey levels is by changing the ratio between the 'on' and 'off' time of the pixel.

Colour LCDs are basically created by tripling the number of pixels and by adding colour filters (Red, Green and Blue) to form triplets of colour pixels. Colour and grey level displays are not discussed any further in this article.

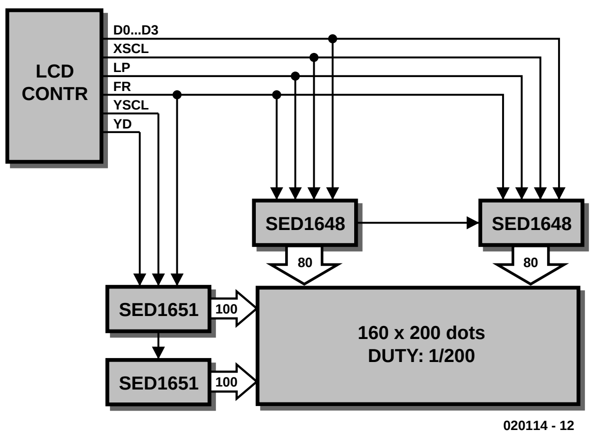



Figure 2. Block diagram of a 'single panel' type LCD module (source: Seiko/Epson datasheet, modified).

## Columns and Rows

Matrix LCD driver hardware looks like a two-dimensional multiplexer. The LCD image is refreshed continuously and updates one complete row at a time from the 'top' to the 'bottom' of the display. The drawing in **Figure 2** shows the general block diagram of a matrix driver circuit. One or more 'segment driver' devices (e.g., SED1648) are responsible for driving all column electrodes and one or more row- or common drivers control the common electrodes. At each moment, only one of the horizontal pixel rows (i.e., common electrodes) is activated by the 'common drivers' (e.g., SED1651) — all the other rows are disabled to avoid shadow images of the active pixels appearing on these rows. The diagram shows two cascaded segment drivers (SED1648), each one controlling 80 segments and two common drivers with 100 outputs each. The controller reads the image data continuously from RAM and sends it to the segment drivers in the correct order and in sync with the currently active output of the common driver. In this way, the segment driver outputs are multiplexed onto the image rows by sequentially activating the outputs of the common drivers. The segment drivers cycle through all the display rows at a frame rate of 60-70 Hz.

## Segment drivers

LCD Segment drivers are constructed as a shift register. Segment data (D0-D3) is shifted-in serially under control of ‘XSCL’. The shift register allows a reduction of pins on the segment driver (4 Data inputs versus 80). Note that unlike ‘normal’ 1-bit shift registers (e.g., a 74164) most LCD segment drivers use a 4-bit wide shift register (D0-D3). Entering four bits in parallel reduces the maximum clock frequency required on XSCL. The ‘LP’ or ‘Latch Pulse’ signal transfers the contents of the shift register to a latch when the shift register has been completely filled with new data. The latch register is internally connected to the LCD output drivers providing the necessary voltages (i.e., V0, V2, V3 or V5) to the segments on the panel. The output drivers also deal with the ‘FR’ (or ‘M’) signal that prevents harmful DC build up over the LCD electrodes (more details on this to follow).

The timing diagram in **Figure 3** shows the basic controls required for the segment drivers: pixel data is presented on D0-D3 and XSCL clocks in this data. This means that an 80-bit segment shift register is filled after 20 XSCL cycles (4 bits x 20). Assuming an LCD module with three cascaded segment drivers (i.e., 240 pixels horizontally), we would see a latch pulse after 60 XSCL cycles (three devices x 20 XSCLs). Note that LP is similar to Hsync in video signals.

Typically the first set of D0-D3 shifted into the segment drivers appears at the leftmost edge of the LCD, the next set appears to the right of these, and so on (See **Figure 4**).

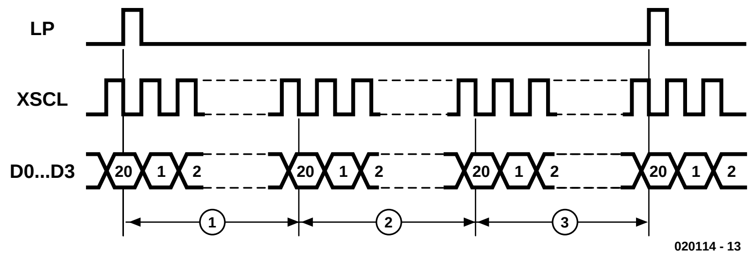



Figure 3. Segment driver timing (source: Seiko/Epson datasheet).

| Set 1 D3...D0 | Set 2 D3...D0 | Set 3 D3...D0 | ..... | ..... | ..... | Set 59 D3...D0 | Set 60 D3...D0 |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| | | | | | | | |

020114 - 14

Figure 4. Mapping of databits onto LCD panel.

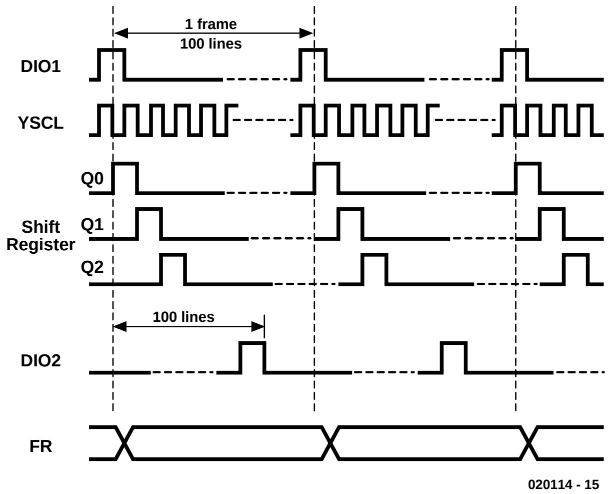



Figure 5. Common Drivers timing (source: Seiko/Epson datasheet, modified).

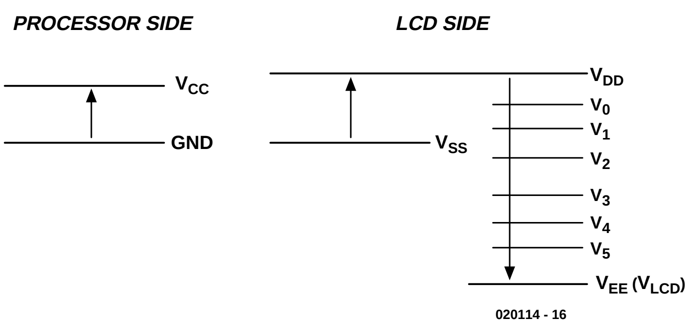



Figure 6. LCD bias voltages: Vdd >= V0 > V1 > V2 > V3 > V4 > V5 >= Vee.

## Common Drivers

The LCD ‘common’ drivers are similar in design to the segment drivers, but use only one data input bit. No more than one ‘common’ output should be active at any time and each output is activated sequentially (first C0, then C1, etc.). This is accomplished by shifting in a ‘1’ bit followed by a number of ‘0’s. The Common input data is in fact a vertical sync signal (a.k.a. ‘FLM’ or First Line Marker). One complete cycle of common electrode activation is known as a ‘frame’, which is a term used for regular video signals also. The common driver does not need a latch as a result of this structured input pattern since there is no risk of shadow rows. The shift register is internally connected to the LCD common output drivers providing the necessary voltages (i.e., V0, V1, V4, V5) to the common electrodes. The output drivers also deal with the ‘FR’ signal. **Figure 5** shows the timing diagram for the ‘common’ control signals. The

**Table 1.** Segment Output voltages.

| /Dsp_off | Shift Register | FR | Output |
|---|---|---|---|
| L | – | – | V0 |
| H | L (pixel off) | L | V2 |
| H | L (pixel off) | H | V3 |
| H | H (pixel on) | L | V0 |
| H | H (pixel on) | H | V5 |

**Table 2.** Common Output voltages.

| /Dsp_off | Shift Register | FR | Output |
|---|---|---|---|
| L | – | – | V0 |
| H | L (inactive row) | L | V1 |
| H | L (inactive row) | H | V4 |
| H | H (active row) | L | V5 |
| H | H (active row) | H | V0 |

shift register input data is presented on 'DIO1' and YSCL clocks in this data. Assuming an LCD module with only one common driver for 100 pixels vertically, we would see each of the outputs (Q0-Q99) activated in turn. The DIO1 input of the first common driver is connected to the FLM control signal of the LCD module. FLM will carry logic '1' once for every 100 YSCL cycles. Note that the FLM bit will shift out of the first common driver device at 'DIO2' after 100 YSCLs and could be used for cascading drivers. On most display modules 'YSCL' and 'LP' are actually connected together: the next row is activated when new segment data is latched to the segment driver outputs. The FR signal is usually synchronised to the falling edge of FLM. The 'duty cycle' of the LCD is related to the number of common outputs: 200 common outputs result in each row being 'actively' driven for 1/200th part of the time, which is still relatively low. One improvement is to divide the screen into an 'upper half' and a 'lower half'. This mode is called 'Dual Panel'. The duty cycle is now doubled in comparison to the 'Single Panel' mode. Most panels of 200+ lines are dual-scan. Active Matrix or TFT (Thin Film Transistor) displays have an effective duty cycle of 1:1 since every pixel is driven constantly by a small driver circuit. The driver circuit can be seen in the corner of each pixel if you examine the LCD closely. This added complexity explains the higher cost and better image quality of a TFT display in comparison to passive matrix displays discussed here.

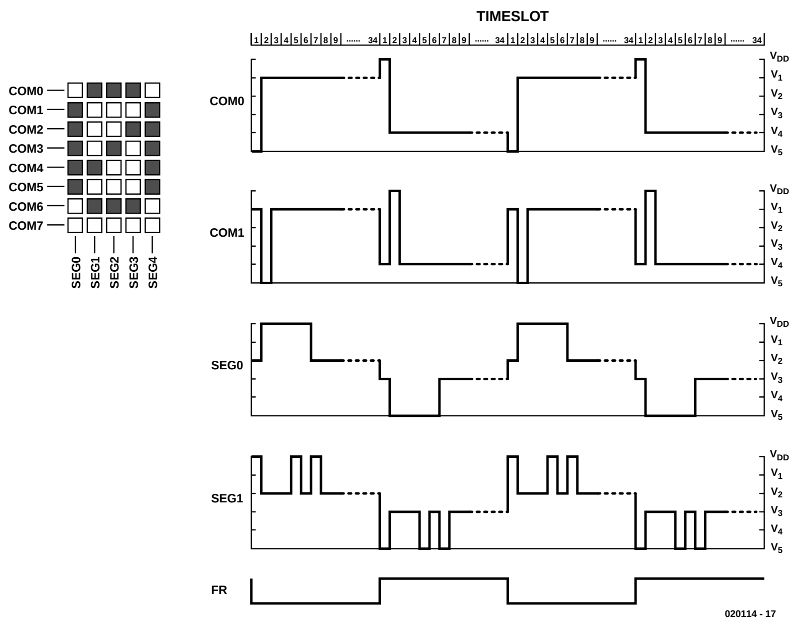



Figure 7. LCD segment and common waveforms (source: Seiko/Epson datasheet).

## Burn-in and LCD driver voltage levels

LCDs require regular polarity switching on the outputs for common and segments to avoid the build up of an average DC voltage that would damage the electrodes of the display and polarise the liquid crystal material. This problem shows up on pixels that are 'on' for more than a few seconds. Damage to the LCD can become permanent, like a CRT screen 'burn in'. The FR signal is needed to avoid DC problems and guarantees a 'pseudo' AC signal on the electrodes by reversing the voltages used for switching pixels either 'on' or 'off'. Usually FR is toggled at every frame. This ensures that the crystals change twist direction every other frame and the average DC field is zero. The driver voltage levels V0-V5 are known as the LCD 'bias voltages' (see **Figure 6**). Note that most LCD modules are designed in such a way that Vcc=Vdd=V0=5 V.

**Table 1** and **Table 2** show the relationship between shift register logic levels and driver output voltage levels. Note that the '/Display Off' control signal results in a 'V0' level on all outputs of both the Segment and Common drivers. This means that all pixels are 'off' and there is no DC build up because there is no voltage difference across the electrodes.

## Example of LCD electrode waveforms

**Figure 7** shows the voltage levels on the outputs of the segment- and common drivers for a simple image. The waveform diagram represents the case of a display with 34 commons and at least 5 segments of which only two common outputs and two segment outputs are shown. The bottom trace represents the 'FR' signal which is inverted after each complete frame of 34 rows. As expected, Common 0 through Common 33 are activated sequentially, resulting in an output level of V1 (inactive common) or V5 (active common). Voltage levels on the segments are either V0

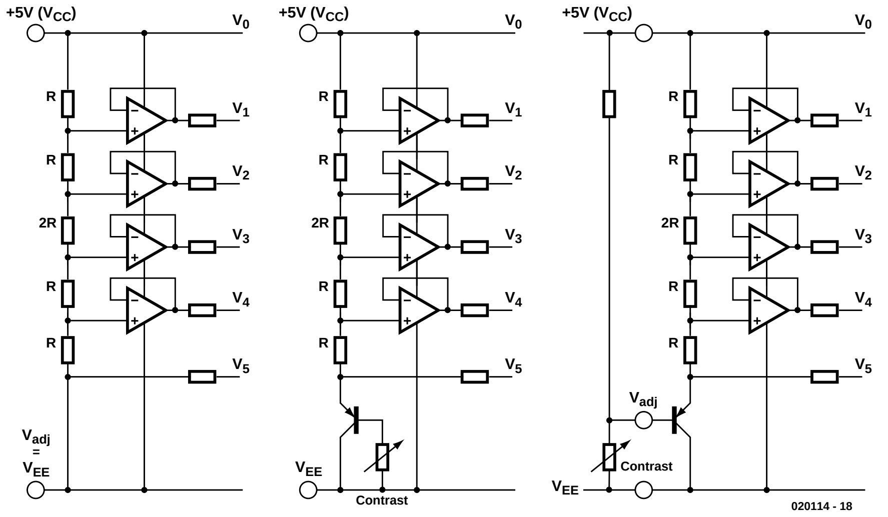



Figure 8. Three examples of typical bias generators used on LCD modules.

(active or black pixel) or V2 (inactive pixel). Note how the waveform of Segment 0 corresponds to the pixels in the first column of the example image (Segment 0 is 'off' for Common 0, 'on' for Common 1 through Common 5 and 'off' for the remaining Common outputs).

Pixels are only visible if the voltage difference between common and segment drivers is sufficiently large. Observe the waveforms for the first row of the first frame ('FR' is '0') : Common 0 is active and the voltage difference between Segment 0 and Common 0 is only V2-V5 (hence the top-left pixel is 'off'), at the same time the voltage difference between Segment 1 and Common 0 is V0-V5

## Table 3. Overview of typical pin names used by LCD manufacturers.

**D0-D3 (Data0-Data3), Display data signals for single panel LCD. Alternative names: XD0-XD3**
Some displays use only one data signal input (D or XD). This display type has 1-bit wide segment shift registers. This solution is only practical for low-resolution displays because the XSCL rate is quadrupled.
The so-called 'dual panel' displays use UD0-UD3 for 'upper half' data and LD0-LD3 for 'lower half' data of the screen. Dual panel displays have two sets of segment shift registers (one for the upper half and one for the lower half of the display). The segment drivers share XSCL. All common electrodes are connected to both the upper half and the lower half of the display. The 'Low-Cost LCD controller' does not support dual-panel displays.

**XSCL (X shift clock). Alternative names: CL2, CP2, CP, Dot Clock, DCLK**
Display data shift register clock. D0-D3 is clocked in on the falling edge of XSCL. The XSLC frequency depends on display frame update rate and number of pixels per line; typical value is 5 MHz.

**LP (Latch Pulse). Alternative names: CP1 (clock pulse), CL1, HS, HSYNC, LOAD**
Display data latch signal. The falling edge of LP transfers pixel data from segment shift register to the segment driver latch and thus outputs a complete row of pixels on the segment drivers.
The LP frequency is equal to the line update rate and is given by : HSync = (Frame Rate x Number of Lines). LP is usually connected to the YSCL input of the common shift register. This results in clocking the common shift register and thus activating the next row, which will then show the newly loaded pixel data.

**FLM (First Line Marker). Alternative names: DI, DIN, YD (Y data), S (Scan start-up), VSYNC**
Data input of the common shift register. FLM receives a synchronising signal at the start of each frame (equivalent to VSync in video signals). The FLM frequency is equal to the display frame rate, typical values are 60 Hz or higher.

**FR (Frame signal). Alternative names: M, DF, WF**
Frame AC alternate signal. Usually FR is toggled for each frame and the frequency is therefore typically half the display frame rate (30 Hz or higher).
Some display modules derive the FR signal internally from the FLM and LP. These modules obviously do not need an external FR. Some datasheets describe this signal as 'AC drive waveform' or something similar. Do not confuse this with the AC power supply for a CCFL backlight!

**/DISP_OFF (Display Off). Alternative names: /DISP, /D_OFF, ENA, INH**
Display on/off signal (H=on, L=off). The signal disables the common and segment drivers. Used to blank the display (e.g. to save power or avoid visible display updates). The disable signal is also required to avoid damage to the display by DC voltages when the LCD controller is not yet active (e.g. after reset).

**Vdd, Vss (Logic power supply). Alternative names: Vcc, Gnd**
Power supply voltage for logic (5 V). Current is low, typically below 100 mA (unless Vdd is also used to power the CCFL converter).

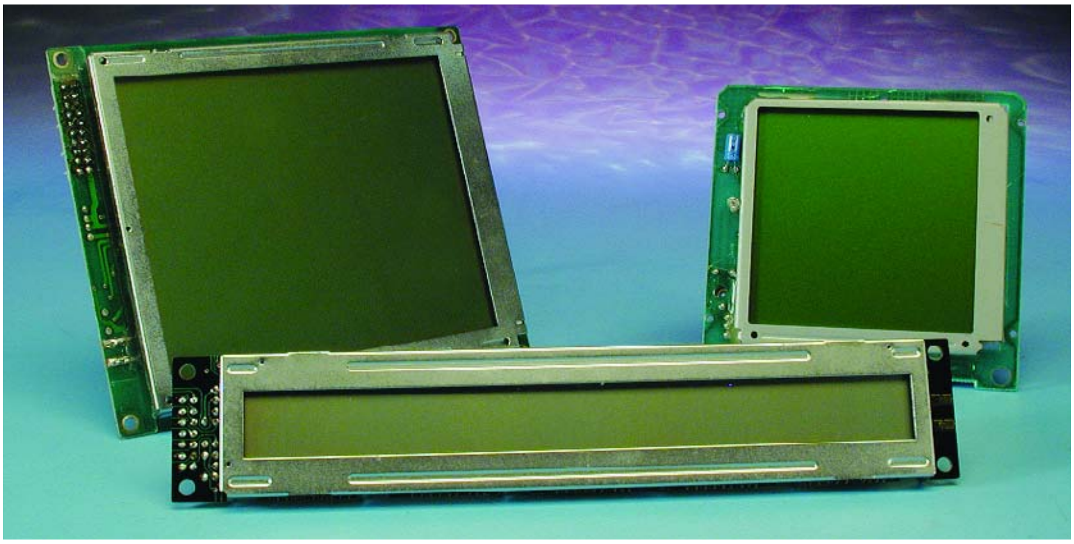
**Vee (Power supply for LCD). Alternative names: Vlcd, Vssh**
Power supply for LCD drivers. –5V to –25V depending on model. LCD datasheets usually specify this voltage relative to Vdd as opposed to relative to Gnd! Required current is very low, typically only a few mA. Vee should not be applied when Vdd is not present. Failing to do so may damage the LCD drivers !
Some display modules have onboard DC/DC converters to generate Vee from Vdd. Obviously these modules do not need an external Vee. There may be an output pin on the module for Vee and/or some circuitry to allow external setting of the contrast voltage Vadj.

**Vadj (Contrast Adjust). Alternative names: V0, Vcon**
LCD contrast adjustment voltage. The contrast voltage usually has some effect on the optimal viewing angle for the display also. Some LCD modules use an adjustable Vee as a means to change the contrast. Some displays may also have the contrast preset on the PCB. Do not confuse the alternative signal name '$V_0$' with 'V0' used to identify the highest bias voltage!

**A, C (Anode, Cathode)**
Power supply for LED backlight illumination of display. One or more LEDs may be switched in series. Check specifications before applying power to avoid damage!

**AC, AC (Power supply for CCFL backlight), Alternative names: CCFL, HV, HOT & GND**
Power supply for Cold Cathode Fluorescent Lamp (CCFL) backlight illumination. CCFLs require approximately 200 VAC at approx. 100 mA. Check specifications before applying power to avoid damage! For safety reasons the high voltage power supply for the CCFL backlight is always provided on a separate connector from the control signals.
Some display modules have onboard DC/AC converters to generate CCFL power from Vcc or from a separate supply. These DC/AC converter modules might also have additional pins to adjust the brightness of the CCFL.
Backlight is optional on most LCD modules and may not be present. However, displays designed for CCFL (Transmissive type) usually need an activated backlight. The image will be very dark and almost invisible when the backlight is off. CCFL displays create a deep blue/black background and brilliant white pixels.

**FG (Frame Ground)**
Ground for metal frame of LCD module; connect FG to GND externally.



which is above the threshold and will result in an active pixel. The voltage between any Segment output (either 'on' or 'off') and any of the inactive Common driver outputs is always below the minimum threshold needed to switch on a pixel. The figure shows that for two sequential frames the driver waveforms are shaped equally but opposite as a result of the toggled 'FR' signal. The flipped waveform results in the same absolute voltage differences between the electrodes and thus the same visible image on the display, while cancelling out the DC component across common and segment electrodes.

Note that LCDs are in general tolerant with respect to control signal timing (i.e., frequencies, pulse widths or jitter). The control signals are basically asynchronous and the display should operate fine as long as the correct relationship between the controls is maintained.

## LCD Bias voltage generator

The driver voltages are generated by a 'bias' circuit similar to the examples shown in Figure 8. Vcc and Vee are applied to a resistor voltage divider and the resulting voltages are buffered by op-amps and buffer electrolytics that stabilise V0-V5 under the dynamic load of the multiplexed image content. The transistor and the variable resistor provide both temperature compensation of the bias voltage and allow contrast setting. The bias circuit on the right side of **Figure 8** shows the basic configuration for external contrast adjustment. In this case an external voltage is provided on the Vadj pin to set the LCD contrast level.

The bias circuit is normally implemented on the LCD module using discrete (SMD) components (e.g., LM324 quad op-amp). Generating V0-V5 on the LCD module reduces external wiring and, more importantly, avoids long feeds that could pick up noise on the sensitive driver voltage levels (e.g. from XSCL or from 50-Hz AC hum) which would be visible on the display as modulation in the pixel contrast. Note that Vee should not be applied when Vdd is not present. Failing to do so may damage the LCD drivers or cause them to latch-up!

## LCD Module Interface Signals

To close off this month's instalment, **Table 3** lists the interface signals typically found on matrix LCDs. Unfortunately there is neither an established nomenclature nor a standard pin-out for matrix LCDs. The solution for contrast setting and the related Vee/Vadj interfacing tends to vary across manufacturers and even between different modules of the same brand.

(020114-1)

**Web References**

– www.seiko-instruments.de/
– www.optrex.co.jp/us/product/catalog/index.html
– www.eio.com/public/lcd

# Programmable Switching Clock with Sensor Inputs

## Another winning project from our Flash Micro Competition

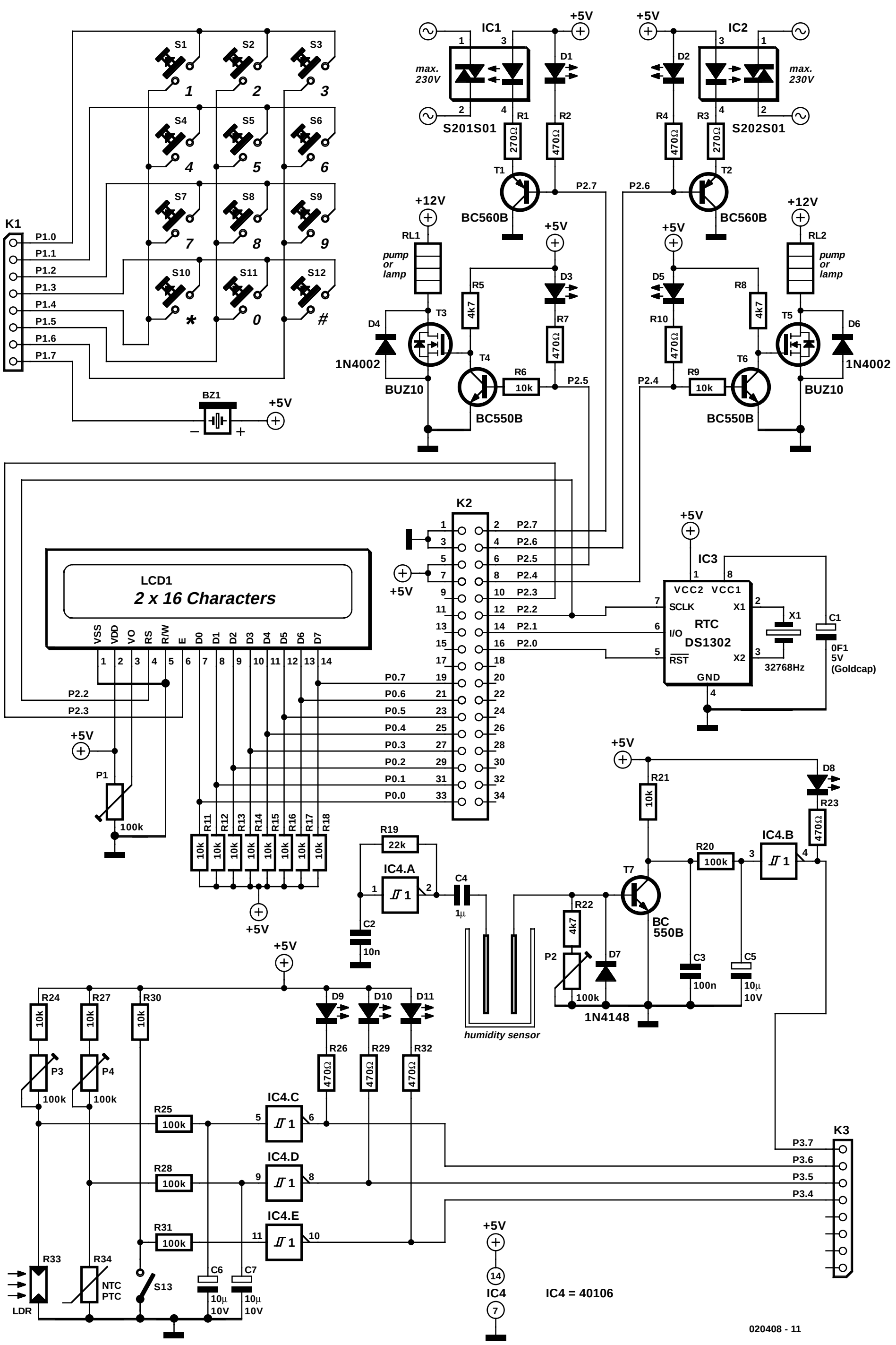



Figure 1. Peripherals strewn around the Flash Micro board: an LCD, an RTC and a keyboard matrix.

Design by W. Wätzig

Using a handful of external components our AT89S8252 Flash Micro board can be turned into a switching clock with sensor inputs and four switched outputs. The author demonstrates the programming of various inputs and outputs, aiming to control, for example, a small irrigation system.

The peripheral circuitry to add to the Flash Micro board consists of a (telephone) keypad connected as a matrix across port lines P1.0 through P1.6, a real-time clock type DS1302 (from Dallas Semiconductor) whose 3-wire bus is hooked up to port lines P2.0, P2.1 and P2.2, and, finally, a standard LC display with 2 rows of 16 characters. Thanks to the abundance of port lines on the 89S8252 controller, the LCD can be operated in 8-bit mode when connected to P0.0 through P0.7. The real-time clock chip supply is backed up using a 0.1-microfarad Goldcap to make sure the date & time information remains valid even when a (not too long) power cut occurs.

Two port lines have double occupation: the keyboard connection P1.6 doubles as an output to a buzzer, while the serial clock input of the RTC, P2.2 (CLK, pin 7 of the DS1302), is also taken to the data/control input (pin 4) of the LC display. Thanks to clever software programming, such double use of port lines does not present hardware conflicts or contention.

As an example, the Switching Clock may be used for an automated irrigation system in a conservatory, greenhouse or similar. External signals arriving by way of three Schmitt triggers may be processed, for example, supplied by the following sensors:

– a humidity sensor;
– an LDR as a twilight detector;
– an NTC/PTC for temperature measurement;
– a switch.

The outputs may be applied for:

– a 12-V switch using a power MOSFET energizing a lamp or a pump;
– a 230-V solid-state relay for heating control.

The inputs are connected to port lines P3.3 through P3.6, the outputs, to P2.2 through P2.7.

## Process control statements

The switching clock may be used to execute up to nine process control 'scripts', stored in an EEPROM. The individual statements consists of logic combinations with this syntax:

```
<Output> =
(<Input1| Output1>
<and|or>
<Input2|Output2)
and
(Start-Time .. End-Time)
```

The microcontroller executes the process statements once a second and sets its outputs accordingly. For an irrigation system, process statements of the kind shown below may be used, provided, of course, the inputs and outputs are connected to the appropriate hardware.

```
when (DRY and WARM) and
TIME(16:00..20:00) then switch on PUMP
or
when (DARK and not PUMP) and
TIME(19:00..22:00) then switch on LAMP
```

In normal mode, the output ports are immediately controlled. In trigger mode, however, the system waits for the first time the condition (EAI */+ EA2) * TIME is satisfied. The time check may be disabled or switched to hourly.

All ports are internally interrogated and set as active-Low. The input ports are driven active-High externally because an inverting Schmitt trigger is inserted in the line. Although only available in German, The assembly code listing should prove a useful source of information when it comes to learning programming techniques for the 89C8252 micro. The relevant file may be downloaded free of charge from our website under number **020408-11.zip** (see month of publication).

(020408-1)

## USB UART

**December 2001, p. 12-18 (010207-1)**

New USBUART.SYS and USBUART.INF files have been written that enable the software to be installed under Windows XP. The new files are available from the Free Downloads section on our website.

## Atmel Micro Programmer

**September 2001, p. 52-57 (010005-1)**

The author of the project has kindly supplied updated versions of the PC control software (010005-11) and the firmware (010005-41). The relevant files are available from our Free Downloads page. Apparently the last few bytes could not be programmed when the 4k version of the Atmel chip was used. The problem was caused by timing errors and is solved by the new software.

## Autoranging Capacitance Meter

**February 2003, p. 60-64 (020144-1)**

The parts list and the circuit diagram indicate different transistors in position T1. Both the BC559B and the BC557B will work in this circuit, although the BC559B is the preferred device. The Farnell order code for the Pactec case used in this project should be: 736-442.

## 20/40-MHz Logic Analyser

**February 2003, p. 12-21 (020032-1)**

The parts list and the circuit diagram should be amended to read:

IC8,IC9 = 74F393 or 74LS393

IC13 = 74LS573 or 74F573

IC11 = 74LS688

Kits and components for this project are available from C-I Electronics, PO Box 55544, NL-3008-AM, Rotterdam, The Netherlands, www.dil.nl

## Temperature Indicator for the PC

**April 2003, p. 70-71 (020380-1)**

In case relatively long wires are used between the LM75 and the microcontroller, it is recommended to fit pull-up resistors on the SDA and SCL lines. These resistors can take values between 3.3 kΩ and 10 kΩ.

# MOSFET Testing using a Multimeter

## Yes it can be done

By Carlo Cianferotti

carloc@infol.it

This short article will deal with testing power MOSFETs using an ordinary digital multimeter. Anybody who have ever tried this must have marvelled at strange readings and short circuits found even on surely working devices. Of course, there is no magic involved!

Testing bipolar transistors, whether low or high power, is all plain sailing if you have just an ordinary ohmmeter available and know (1) the connections on the device and (2) the six steps involved. Such a simple test should spot 8 out of 10 defective bipolar transistors, although your editor had to admit defeat with some high-power UHF RF transistors he tried to examine the other day. For example, an MRF646 50-watts 'beast' checked out just fine electrically but its in-circuit power gain was nothing to write home about! Returning to the subject of our article, MOSFET testing is a different (but not necessarily nasty smelling) kettle of fish.

### Whither charge and voltage?

It can be argued that MOSFETs are charge controlled devices, because the gate, which is the control electrode, represents a virtually ideal capacitor exhibiting an extremely low leakage current. However, the same charge produces a voltage, which in turn determines the degree of conduction so these wonderful devices may equally be called 'voltage-controlled'.

No matter if voltage or charge-controlled, when a MOSFET is out of its circuit, any charge stored in it will stay there, keeping the device on if it is positive, or keeping it off if it is negative (we are talking of most common N-channel devices, for P-channel MOSFETs you just have to reverse any polarity involved). For an N-channel device, 'negative' also means below the threshold necessary for switching the MOSFET on.

In fact when you handle a MOSFET for testing — pulling it from the circuit or from its protective packaging — your fingers, the soldering iron etc., will typically cause a random charge to be stored in the gate-source equivalent capacitor.

The first thing to be done is give this charge a known value because only then does it become possible to check the drain-source path (junction) for correct on/off operation. Let's see how this is done in practice.

### Preparing for testing

First of all you have to switch your digital multimeter to the Diode Check range. This way your multimeter will supply the junction under test with a voltage that's usually of the order of a couple of volts (open-circuit) and a current limited to a few milliamps. This is just what we need — don't try to use an ohm range since the voltage supplied is then much lower (approx. 0.2 volts) and certainly not enough to switch MOSFETs on and off.

Now its time to lay your MOSFET on the table surface. No matter if the surface of the table is conductive or not, the most important point to observe is that the MOSFETs leads do not touch anything. Also be sure not to touch the leads or probe tips with your fingers so as not to lose any stored charge. In the case of power MOSFETs, the Drain tab can be freely touched and laid on the table surface, but a safer method is to pick up and hold the power MOSFET by the tab, then touch the table surface with your other hand, and only then lay the MOSFET on the table.

### Let's test again

At his point you are ready for the actual testing, which involves a number of steps described below.

**1.** In the first test we switch the MOSFET off and test its gate-source (GS) junction.

| MOSFET | Gate | Source | Expected reading |
|---|---|---|---|
| Meter | – (black lead) | + (red lead) | Open circuit |

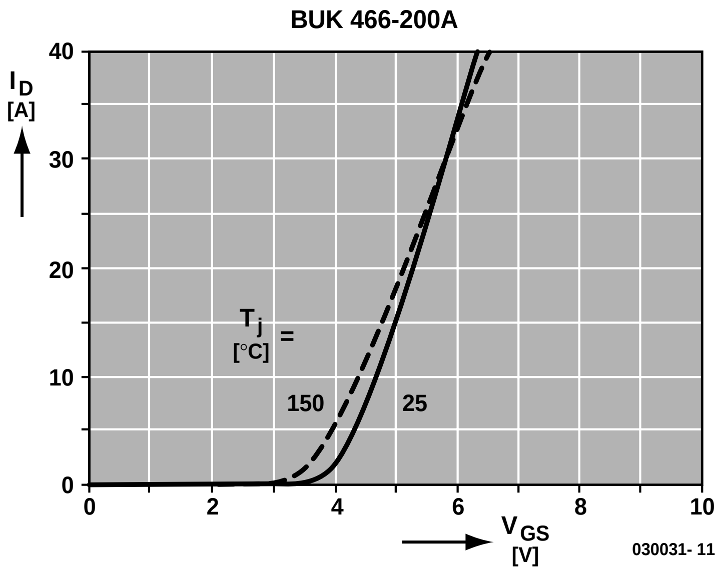



Figure 1. Typical transfer characteristics of an ordinary power MOSFET, in this case, a BUK446-200 from Philips Semiconductors. Graph shows $I_D = f(V_{GS})$ at $V_{DS}$ = 25 V, with two values of $T_j$ as parameters.

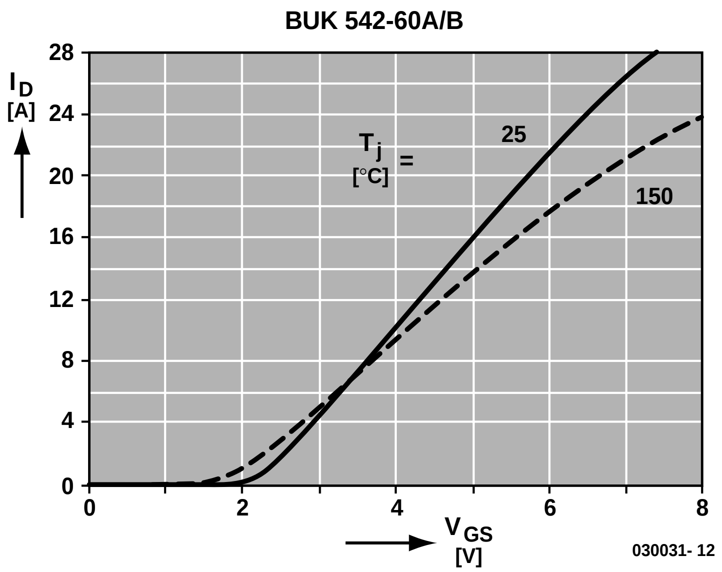



Figure 2. As Figure 1, but for a Logic FET, in this case the BUK542-60A/B.

Any reading other than 'open circuit' means that gate is short-circuited to the source and that the MOSFET may be discarded without the need for any further testing.

**2.** Now that we have the 'off' charge in the gate we can test if the drain-source (DS) junction is opened. Most MOSFETs have an integral reverse protection diode across the drain and source and this may be tested simply by reversing the test voltage polarity.

| MOSFET | Drain | Source | Expected reading |
|---|---|---|---|
| Meter | + (red lead) | – (black lead) | Open circuit |

Any reading other than open-circuit means that MOSFET has a short circuit and should be discarded. Meters usually give readings in mV in this range, so you can expect something between 250 mV and 500 mV for the forward diode drop.

| MOSFET | Drain | Source | Expected reading |
|---|---|---|---|
| Meter | – (black lead) | + (red lead) | Open circuit / Forward diode drop |

**3.** Now it's time to switch our MOSFET on.

| MOSFET | Gate | Source | Expected reading |
|---|---|---|---|
| Meter | + (red lead) | – (black lead) | Open circuit |

This way you double check that the gate is not short-circuited. A wrong reading at this point is very rare, but nonetheless you should discard your device if you don't see an open circuit.

**4.** Now that we have the 'on' charge in the gate, all we have to do is check the drain-source junction for proper conduction. This should be done using both polarities because when a MOSFETs is on, it acts as a small value resistor, irrespective of the direction of the current flow.

| MOSFET | Drain | Source | Expected reading |
|---|---|---|---|
| Meter | + (red lead) | – (black lead) | Short circuit |

| MOSFET | Drain | Source | Expected reading |
|---|---|---|---|
| Meter | – (black lead) | + (red lead) | Short circuit |

If you don't get these results, the MOSFET has a permanently open-circuit Drain-Source junction and should be discarded.

If the suspect MOSFET has passed steps 1 through 4 successfully it can be relied upon to work properly. As a matter of fact, voltages and currents delivered by multimeters are usually much lower than those required for effective testing of power MOSFETs (IRF, BUZ, etc.) but nonetheless this simple test procedure has given very good results over years of field tests.

If you look at the $I_D = f(V_{GS})$ graph in **Figure 1**, you'll notice that conduction starts at a gate-source voltage of 3.5 to 4 volts, while at 5 volts (i.e., TTL High level) some 15 A is allowed to flow through the drain-source junction. The graph is for a Philips BUK466-200A which can be describes as a 'typical' example of a power MOSFET.

Other devices called 'logic FETs' start to conduct at slightly lower levels of $V_{GS}$, for example, at about 2 V already in the case of the BUK542-60 (see **Figure 2**). This makes these devices the perfect choice for direct insertion between a logic output such as a microcontroller port line set up as an output and a (very heavy) load like a power relay or a motor. Again, this is an example only and the graph should not be taken to apply to 'any old logic FET' — see the notes below on finding exact datasheets of the device(s) you're working on.

## P-channel devices and the ohmmeter

If you wish to test a P-channel device you should obviously swap the polarity of the test probes in the above tables.

Using an analogue moving coil multimeter is also possible but these instruments usually do not sport a Diode Check range. This requires some investigation about the open circuit voltage and short circuit current supplied by the meter. You should aim at one ohm range to get 2-3 V and 5-20 mA respectively which is ideal for safe testing. You can find out by connecting a second multimeter to the one used for testing MOSFETs. First select a voltage range and the check open circuit voltage on the probes, then switch to current range and note the short circuit current. Usually the OHM×1 or OHM×10 range will do the job. Finally, as we are sure you will note at some point when attempting to use an old fashioned ohmmeter, the – (black) lead is usually positive (+) and vice versa!

## Whence the pinout?

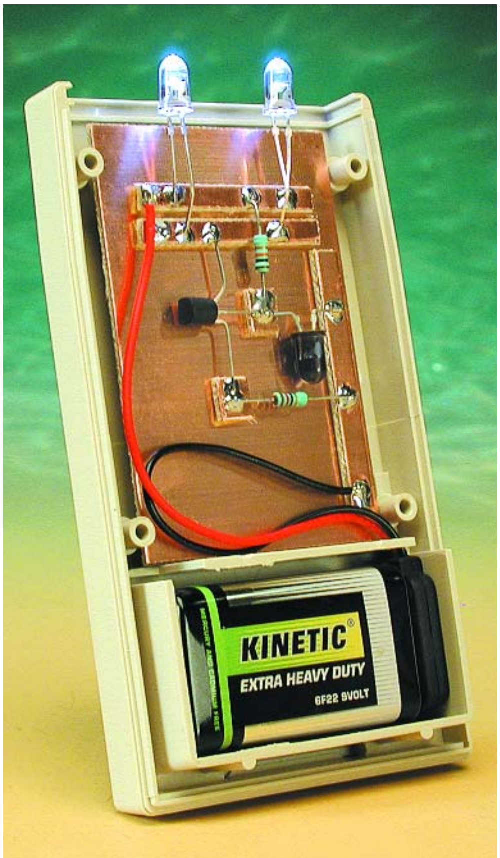
At the risk of stating the obvious, you should always know **exactly** where the gate, source and drain pins are on the device you wish to test using the method described in this article. Educated guesses, 'a friend told me' and 'seem to remember' are worthless in this respect and may lead to costly mistakes and hours of fruitless efforts at repairing equipment. The information you want should be obtained from manufacturers' data books or from original datasheets downloaded from the manufacturers' website.

(030031-1)

# Two-Eyed LED Lamp

## One lamp, three types of LED

Design by W. Zeiller




**We have featured numerous torch circuits using white LEDs. This simple design adds a couple of other types of LED to extend light emissions either side of the visible spectrum.**

LEDs are usually driven from a constant current source so that the LED brightness is not dependent on a varying supply voltage. The circuit here in **Figure 1** shows a constant-current source produced by transistor T1.

To explore its workings in a bit more detail assume that at switch on the voltage at the base of T1 ($V_B$) is greater than the base conduction voltage ($V_{BE}$ = 0.65 V) of T1. Current flows into the base and T1 starts to conduct. This will make current flow through its collector-emitter junction and resistor R2, generating a voltage across R2. This raises the emitter of T1 above ground potential. The current through R2 can only reach a level where the voltage drop across R2 plus $V_{BE}$ of the transistor is equal to the base voltage $V_B$. Current cannot get any larger because that would generate a bigger voltage across R2 which would turn off the transistor. The collector current therefore settles at a value so that the voltage drop across R2 plus $V_{BE}$ equals the base voltage $V_B$.

We have produced a constant current through R2 defined by the constant voltage $V_B$.

In the circuit, R1 and D1 form the constant-voltage source. D1 is an infrared LED and has a forward conduction voltage of around 1.0 V. In contrast, red LEDs conduct at about 1.8 V while white ones conduct at about 3.5 V. An IR LED was chosen just for its forward conduction voltage and the actual type is not important, you could salvage one from a scrapped TV remote or you may have one in your junkbox. Alternatively if such an LED is not available it is possible to replace D1 with two diodes connected in series: a standard silicon diode (0.65 V) e.g., 1N4148 with a Schottky diode (0.35 V) e.g., BAT 85 or BAT 43.

Using a supply voltage of 9 V and a 10 kΩ resistor for R1 the current through D1 will be:

$$I_{LED} = (V_B - V_{D1}) / R1 = 800\ \mu A$$

ignoring current flowing into the base of T1. This level of current is too small to produce any IR output from D1 but prolongs battery life. To produce IR light (you still won't be able to see it though) the current through D1 needs to be about 20 mA, this gives a value for R1 of 510 Ω. D1 is not a perfect voltage source and the increased current will raise the voltage drop across the diode.

The constant current through the collector–emitter path of T1 is divided between two white LEDs, each requiring 20 mA so the value of R2 can be found:

$$R2 = (V_{D1} - V_{BE}) / I_{D2,D3} = (1.0\ V - 0.65\ V) / 40\ mA = 8.75\ \Omega$$

To be on the safe side R2 is increased to 12 Ω, the difference in brightness is hardly noticeable.

### Series or parallel?

At first glance it seems pointless to connect both white LEDs in parallel. It would be better to connect them in series and they would then consume only half of the current. The problem is their relatively high forward conduction voltage. Each white LED conducts when the voltage across it exceeds 3.3 V so if we add the 0.35 V drop across R2 together with the collector-emitter voltage drop across T1 we need at least 4 V supply voltage before current will flow through the LED and T1. Connecting a second LED in series with the first will drop another 3.3 V

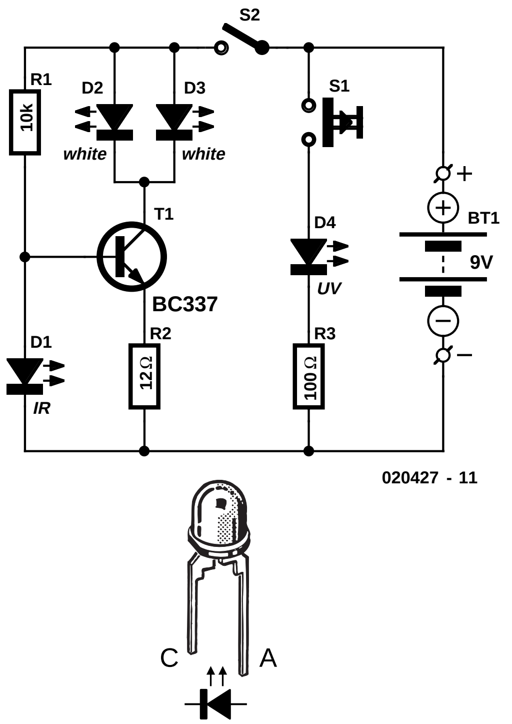



Figure 1. The circuit uses two super-bright white LEDs and a constant current source.

so the supply must now be 7.3 V.

This voltage level should not be a problem if we use a 9 V battery but a closer look at the discharge characteristics for a typical alkaline PP3 shown in **Figure 2** (developments in modern cell design show improvements to these figures but they remain broadly similar) reveals that with a load of 220 Ω (a combined LED current of 40 mA) the cell voltage falls below the conduction threshold (4 V) after 13 hrs use while the 470 Ω load (equivalent to the two LEDs connected in series) will cross the conduction threshold (this time 7.3 V) after only 11 hrs approximately. Some white LEDs have an even higher forward conduction voltage of 3.6 V. Connecting two in a series configuration would raise the conduction threshold for the circuit to 7.9 V and looking at the 470 Ω load curve we can see that the LEDs would extinguish after only 4 hrs use!

During tests the author found that with the LEDs connected in parallel the lamp could burn continuously for 17 hrs powered by a PP3 type battery and even when the voltage had fallen to 3 V (!) there was still enough light to help you pinpoint those elusive keyholes.

LEDs are usually not connected in parallel because the device with the lowest conduction voltage will always conduct before the other LED and take the majority of the current. The reason that this configuration works is that white LEDs have a typical series resistance of about 18 Ω. This produces an additional voltage drop when conduction begins and helps to share the current more evenly between the two.

A problem can occur if the characteristics are too dissimilar because one LED will take most of the current, increasing its dissipation and eventually leading to an early demise, at which point the second LED will become overloaded and also fail when it must take the full 40 mA.

## Ultraviolet

A useful addition to the circuit is the UV LED (D4) connected via a 100 Ω series resistor and pushbutton S1. This light source is handy for detecting counterfeit banknotes.

D4 emits light with a wavelength of 405 nm so it is not a true UV source but rather ‘UV near’ (UV light has a wavelength less than 400 nm), UV light from D4 is relatively harmless to the eyes but it is not advisable to stare directly into the beam. This light source is only ever needed momentarily so a non-latching push button is used to switch it on.

The UV light intensity is very high and series resistor R3 limits current through D4.

## Case notes

The finished circuit is fitted into a plastic enclosure with a separate battery compartment. The circuit is so simple that a purpose-made PCB is not necessary; a ‘custom’ board can easily be made up from off cuts of PCB sheet instead. First remove the dull oxide coating from the copper surface with very fine emery or sandpaper. Cut a rectangular piece of the PCB to fit into the enclosure and act as a base plate, next cut a few strips of the remaining PCB sheet to form the connecting tracks and pads for the circuit. Fix these tracks in the correct position on the base plate using either double-sided tape or super glue. The components can now be soldered onto these tracks to form the circuit. The title picture shows how this has been achieved.

The switch and push button are secured to the enclosure lid with hot-melt glue and then wired to the circuit board.

(020427-1)

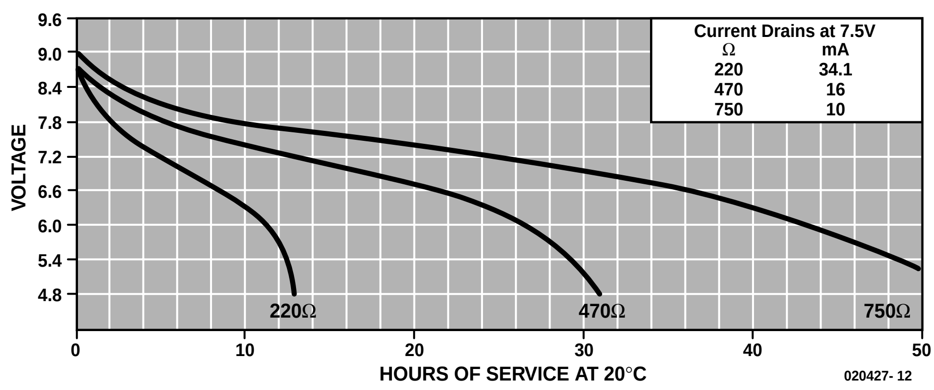



Figure 2. Typical alkaline battery discharge characteristics.

# Printing PCB Artwork

## Make clever use of the Clipboard

By Harry Baggen

Practically all PCB layouts that have been printed in this magazine are currently also available on the *Elektor Electronics* website in pdf-format. Using these, it is easy to make a printout on film in order to expose and etch a PCB yourself. Unfortunately, printing the image at the correct size is often problematic to say the least. That's why we give a number of useful tips and hints in this article so that you can print the PCB layout at the correct size and in the right place on paper or film.

Practically all PCB layouts that are printed in *Elektor Electronics* are currently also available on our website in the form of pdf (Acrobat Reader) documents. With these it is easy to use your printer to make a printout on film and subsequently use this to expose and then etch a PCB.

For some time now, we have been providng both the normal and reflected (mirror-reversed) images of the layout in the pdf file. This enables you to choose which one is best for your particular application. It is usually best if the printed side faces the PCB to obtain the sharpest image when making the exposure. You will need to use the reflected image for this. In addition, the paper size in the pdf-file is matched to the PCB size. This means that PCBs larger than A4-size will not be split across multiple pages.

A significant benefit of using the pdf-format is the fact that the PCB layout is stored in a (Acrobat proprietary) vector format. The result is a file of minimal size but with optimum sharpness of the PCB traces. Unfortunately there are very few programs that can handle this format, so that exporting the layout for manipulation using another program is not all that straightforward.

Many readers appear to have difficulties when printing these files: the PCB is not printed at the correct size or turns up in the wrong place. Especially with expensive film, it is wise to make the best possible use of any all available material. So don't print both (normal and mirrored) layouts and not in the middle of the film (the default settings in Acrobat Reader). That is why we have collected a number of tips that should enable you to produce a good printout with a minimum number of additional resources (that is, without additional programs).

### Selection

Everything normally starts with a version of Acrobat Reader, because that is what you will need to use to open the downloaded file. We will assume that most readers will be using version 5 of this program by now (available on the *Elektor Electronics* annual CD-ROM or from the Adobe website, www.adobe.com). After opening the file, the two layouts will appear on the screen. Typically you will need only one of these layouts. First adjust the zoom size in Acrobat Reader so that the desired layout can be seen in its entirety in the window. Activate the *Graphics Select Tool* in the centre of the toolbar by clicking on it with the mouse. The cursor changes into a cross-hair pointer. We can now select one of the two layouts by positioning the mouse just outside a corner of the desired layout, pressing the left mouse button and selecting the entire layout while keeping the mouse button pressed. When you release the mouse button, a dashed outline becomes visible around the layout (**Figure 1**). You can now print the selected layout using Acrobat Reader or copy the layout to another program using the clipboard for further manipulation or printing.

## Printing

We will first describe the process of printing from within Acrobat Reader. The details of the print options are strongly dependent on the actual printer driver that you are using, so we will only describe the options available from within Acrobat Reader. Click on *File/Print*. The printer independent print menu of Acrobat Reader will pop up (**Figure 2**). In the preview window you will see how the printer will print the selected layout. Make sure that the options *Shrink oversized pages to paper size* and *Expand small pages to paper size* are turned off, otherwise the layout will not be printed at the correct size. It is better if *Auto-rotate and center pages* are turned off as well.

When a selection has been made (like the one we just did) Acrobat Reader will always print it at the centre of the paper or film. If you would prefer to have the printout appear off to one side on the page then you can experiment by selecting a different paper size in the printer-driver. Select, for example, A6 while you have A4 film loaded in the printer (this trick may not work with all printers).

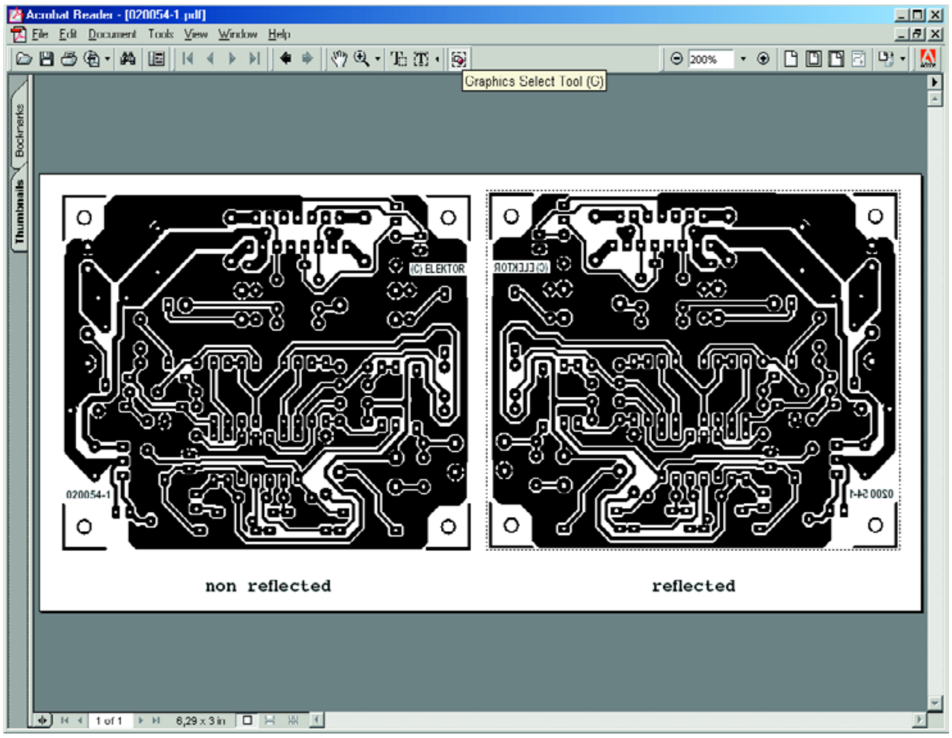
It is a good idea to make a few test printouts on cheap paper before loading the printer with expensive film. Also check that the size of the printed layout is correct (compare with the layout in the printed magazine).

## Via the clipboard

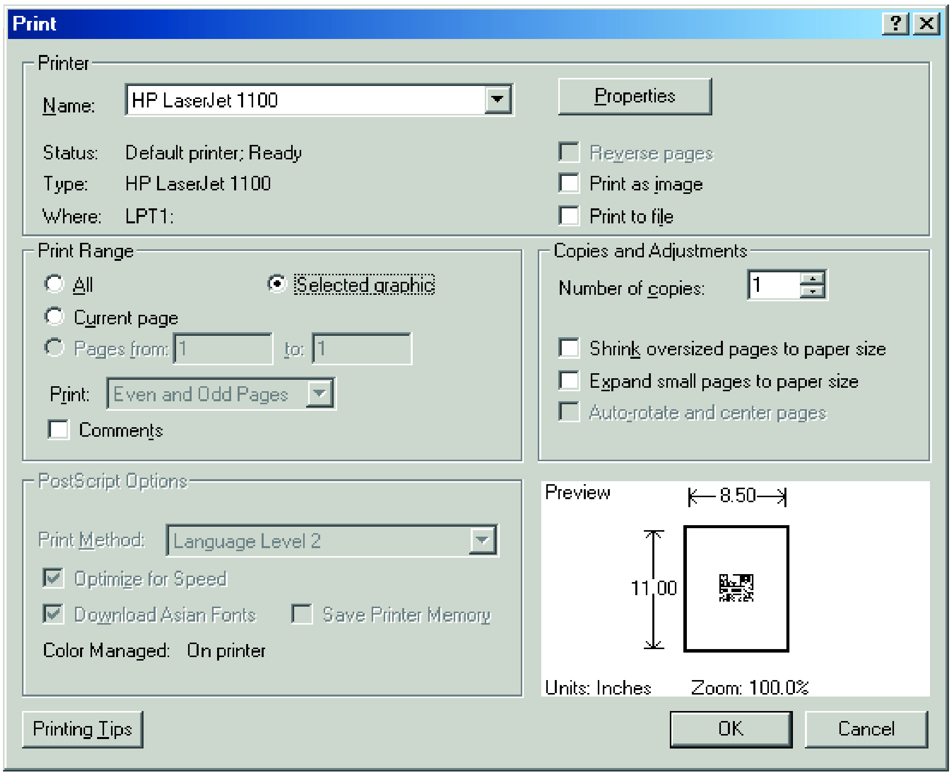
A second method offers somewhat more possibilities. Here we make use of picture editing software that may already be present on your computer and allows the positioning of an image on the page to be adjusted manually. Examples are Paintshop Pro, Picture Publisher or the well-known word-processor Word. We copy the section via the clipboard to Word or the image editor.

When a copy is made using the clipboard, the selected (vector) layout will automatically be converted to a pixel format and this results in a resolution that is too low to produce a good (without ragged edges) printout when using another program. Normally only the selection that is visible on the screen can be copied to the clipboard, but then you are restricted to the pixel size of the monitor. This does, however, not apply to Acrobat Reader! It is possible to copy selections that are only partially visible on the screen. After we have selected the desired layout as previously described, we zoom the image size in Acrobat Reader to 400 or 500% (or even bigger). Only a small part of the layout is visible at this point, but that doesn't matter. Now make a copy to the clipboard by using *Edit/Copy* (or Ctrl-C on

1




2



the keyboard). You can then paste the layout into Paintshop Pro, for example, using *Edit/Paste as new image* or Word (*Edit/Paste*). You can then print the image using that program.

When using Paintshop Pro, the *File/Page Setup* (**Figure 3**) options first have to be adjusted to where exactly on the page you want the layout to appear. In addition, you have to make sure that the PCB will be printed at the correct scale. If you used a magnification of 400% then you will have to reduce the printout to 1/4th of true size, and at 500% to 20%. Once everything has been correctly adjusted you can make the printout. The extra step of the image editor has the advantage that you can make small changes to the PCB, such as deleting a connection or moving a trace. It is true that this has to be done at the pixel level, but as a result of the large magnification the final printout will still be nice and sharp.

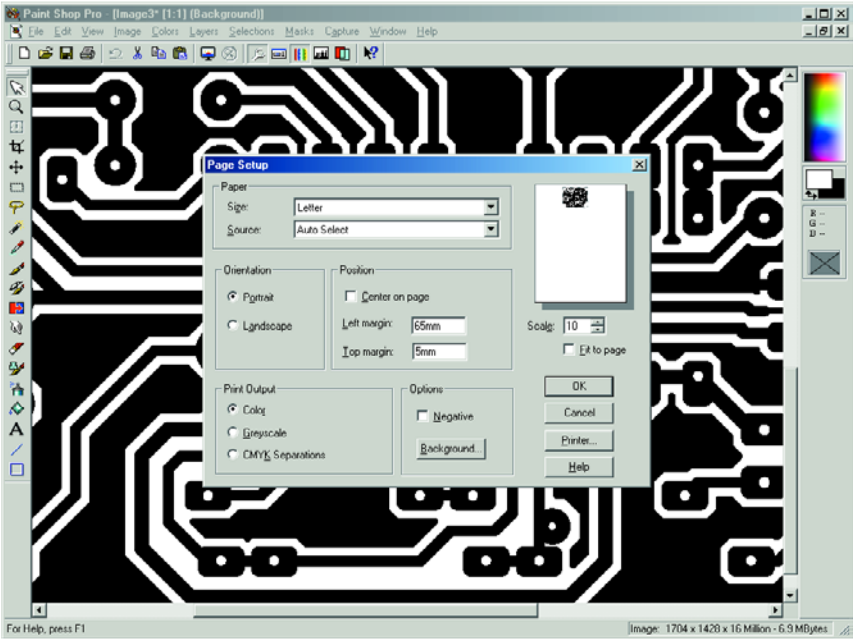
In Word we need to take a slightly different approach. Once the layout is placed on a blank page via the clipboard, we first click to select the illustration. Then click with the right mouse button and select *Format Picture*. Under *Size*, (see **Figure 4**) you can now select the desired scale (depending on the magnification selected in Acrobat Reader, 25 or 20% respectively, and make sure that *Lock aspect ratio* is selected). Also ensure that *In front of text* under the *Layout* tab is selected. The latter enables you the place the illustration in any arbitrary place on the page, using the mouse. You can now print the page. Note that Word does not accept fractional percentages, but you are able to adjust the dimensions exactly.

With one of the methods described above, it should be possible for you to print a PCB layout in exactly the right place on a piece of paper or a sheet of film.

Two more tips before we conclude. Firstly, large PCB layouts that do not fit on one A4 sheet can often be printed by selecting Letter-size paper (slightly wider than A4) or by creating a user-defined (custom) paper size. For this purpose, cut an A3 sheet by hand to the desired dimensions and place it in the paper tray of the printer. Secondly, PCB layouts of older articles are only shown in their normal orientation. If the printer driver does not offer the option of mirror printing of the image, then this function is almost certainly available in the image editing software (Paintshop Pro: *Image/Mirror*).

(030065-1)

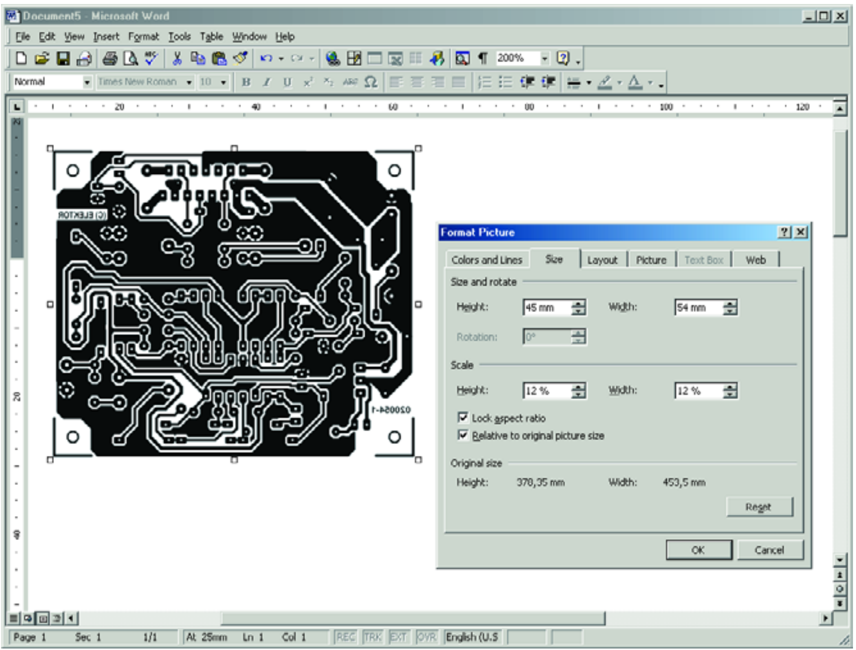
3




## Screen resolution

In some of our experiments it appeared that some programs take the screen resolution into account and others do not. It can happen that a layout that has been enlarged in Acrobat Reader by 400% and subsequently is reduced to 25% in Word does not have the correct dimensions. In that case you will have to do a little experimentation in order to determine the correct reduction factor. You only need to do this once, because once you know this, you can simply use it again on future occasions. The error is often the difference between the screen resolution of 96 and 72 dpi, that is, a factor of 1.33. Fortunately, when using Word, you can also enter the exact dimensions directly, but you will have to take into account any extra white space that you may have selected in Acrobat Reader.

4

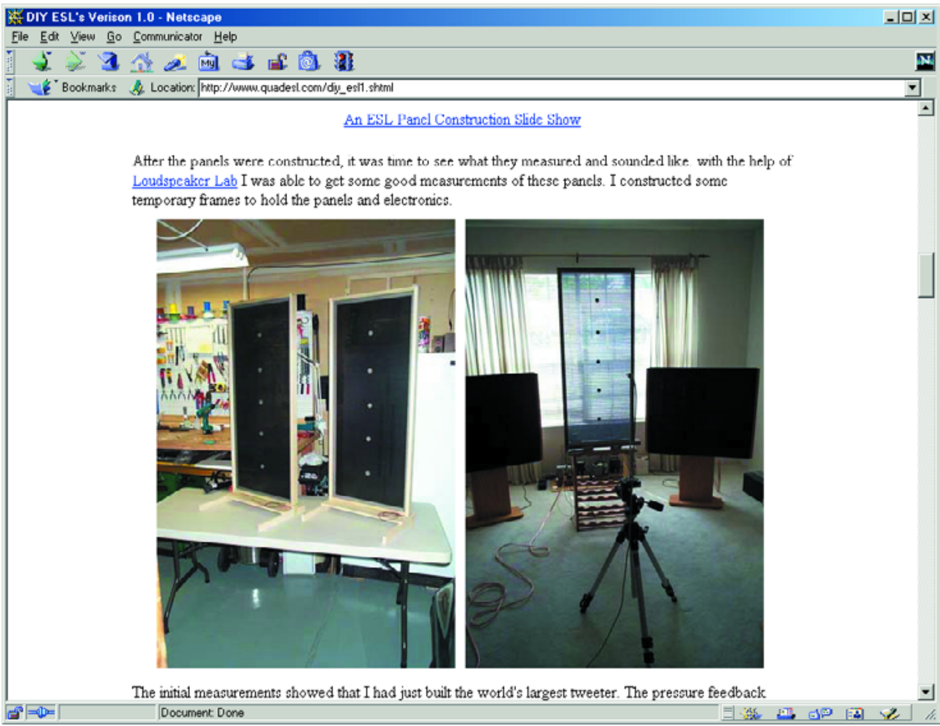


# DIY Electrostatic Loudspeakers

## High tension for sound reproduction

By Harry Baggen

**Building your own loudspeaker boxes using traditional drive units was once a popular pastime but now seems to have lost much of its appeal. Those of you interested in a different approach to sound reproduction may want to have a go at building electrostatic loudspeakers. Amazing results can be obtained from basic materials that cost next to nothing. We trawled the net and found loads of information on these oddball boxes.**




The operating principle of the electrostatic loudspeaker (or ESL) is based on the phenomenon of two electrical current conducting plates with opposite polarities developing a mutual attracting force, and two plates with the same polarity, a mutual repelling force. Most electrostatic loudspeaker units employ two air permeable panels (stators) separated by a taut conductive membrane. When a high alternating voltage is applied across the membrane and the stators, the panel will start to move between the stators in the rhythm of the ac voltage, causing a certain air volume to be displaced. The sound reproduction obtained in this way is (at least potentially) extremely faithful thanks to the small moving mass, the large surface area and the evenly distributed force.

By virtue of its relatively simple overall structure, the ESL is easy to experiment with. Lots of technically inclined music lovers have reached different results from perforated steel sheet, plastic foil and a number of variations on these materials. Some of the results have culminated in a fair number of Internet websites describing such experiments with electrostatic loudspeakers.

A foundation article telling you how to get started with ESLs is Mark Rehorst's story called **How to Make Electrostatic Loudspeakers** [1]. It has been around on the net for many (web-)years and may be


found at various sites. The article covers all basic elements that make up an ESL, from the building materials for the stators right up to the design of a high-tension supply. Anyone with some knowledge of electronics should be able to use this information and be off for a really quick start.

A more extensive and technically 'deeper' article is Neil McKean's **Electrostatic Loudspeaker Design** [2], covering both the design and the construction of electrostatic loudspeaker units.

Lots of ESL builders describe their designs on dedicated websites. There are far too many to list in this short article — only a small selection will be mentioned. The **Mini Mite** [3] by Sheldon Stokes is a small electrostatic panel combined with a Jordan driver in a transmission line enclosure. Another extremely well documented DIY project by the same author goes by the name of **DIY ESL 1.0** [4].

Matthew Anker's **Electrostatic Loudspeaker Page** [5] is also devoted to activities involving electrostatic speakers. In particular, the pages showing lots of constructional pictures are well worth having a look at.

An unusual ESL project covers the DIY construction of **electrostatic headphones** [6]. On his website dubbed 'Headwize' Andrew Radford (maybe a distant relative of the legendary Radford from the valve era?) explains how he changed an old pair of dynamic headphones into electrostatic ones! The story is supported by a number of clear drawings and photographs which go to show the amount of work that went into the design.

Not surprisingly, there are also suppliers of commercially made ESLs offering parts of their products for the benefit of home constructors. Completely assembled panels are particularly interesting as they allow less experienced constructors to build excellent sound reproduction systems. The Dutch manufacturers **AudioStatic** [7], for example, offer their DCI-LT as a construction kit consisting of two ready to use panels and built up high-tension units. All you need to do yourself is build a frame to hold the panels.

The Australian firm **Metaxas** [8] also offers an ESL set, called MET 1. Like the DCI-LT, MET 1 comprises a complete set of parts including a high-voltage section and audio transformers. The remarkable thing about this system is that several panels may be purchased for connecting in parallel.

If you just need an address to order suitable foil from (foil often being the only elusive material) you'll be pleased to know that there are several sources that can also supply other useful parts like graphite spray. **Twinstatic Audio** is a Dutch firm with a fair assortment of special components for ESL builders. Just a few clicks away but actually at the other side of the globe, in Australia, we found **The Electrostatic Loudspeaker Information Exchange** [10] and **ER Audio** [11].

Finally, the **Electrostatic Loudspeaker Circuit** [12] is a great source of inspiration for ESL lovers. The website has been on line since 1998 (!) and contains a large number of descriptions of commercial and home made ESLs. Here, too, we found extremely useful information on related matters like amplifiers for ESLs. Furthermore, there is a forum which allows hobbyists and fans of the electrostatic loudspeaker to exchange ideas and information.

(035033-1)

## Internet addresses


[1] How to Make Electrostatic Loudspeakers: www.amasci.com/esloud/eslhwto.html

[2] Electrostatic Loudspeaker Design: www.audiodesignguide.com/esl/esl.html

[3] Mini Mite ESL: www.audiocircuit.com/9041-esl-circuit/Diy/Projects/SheldonStokes-SS/9041DESS-MM.htm

[4] DIY ESL 1.0: www.quadesl.com/diy_esl1.shtml

[5] Electrostatic Loudspeaker Page: http://home1.gte.net/res0f2t3/index.htm

[6] Electrostatic headphones: http://headwize2.powerpill.org/projects/showproj.php?file=radford1_prj.htm

[7] AudioStatic-DIY kit: www.audiostatic.com/page9.htm

[8] Metaxas MET 1: www.metaxas.com/pages/masnewfiles/index.html

[9] Twinstatic Audio: www.twinstaticaudio.nl/onderdelen/

[10] The Electrostatic Loudspeaker Information Exchange: www.hitechnetworks.net/bwaldron/main.htm

[11] ER Audio: www.eraudio.com.au/

[12] The Electrostatic Loudspeaker Circuit: www.audiocircuit.com/9041-esl-circuit/9041IMAI.htm